hp 48gII grafikfähiger Taschenrechner

Benutzeranleitung

4. Ausgabe
HP Artikel-Nr. F2226-90022

Hinweis

REGISTRIEREN Sie IHRES PRODUKT AN : www.register.hp.com

Hewlett-Packard Company
4995 Murphy Canyon Rd,
Suite 301
San Diego,CA 92123

Druckgeschichte

4. Ausgabe April 2004

Vorwort

Sie halten einen kompakten Taschenrechner für symbolische und numerische Anwendungen in Händen, der Ihnen die Berechnung und mathematische Analyse einer Vielzahl von Aufgaben in den verschiedensten Fachbereichen erleichtern wird, von elementarer Mathematik über Berechnungen im Ingenieurwesen bis hin zu anspruchsvollen wissenschaftlichen Aufgabenstellungen. Obgleich das Gerät hier aufgrund seiner kompakten Abmessungen als Taschenrechner bezeichnet wird, handelt es sich bei dem hp 48gII um einen vollwertigen grafischen, programmierbaren Handheldcomputer.

Der hp 48gII verfügt über zwei verschiedene Betriebsmodi, nämlich über den RPN-Modus (RPN=*Reverse Polish Notation – Umgekehrte Polnische Notation*) und über den ALG-Modus (ALG=*Algebra*, weitere Details siehe die Seiten 1-11 in der Bedienungsanleitung. Der RPN-Modus wurde bei Taschenrechnern eingeführt, um die Effizienz von Berechnungen zu verbessern. In dieser Betriebsart werden zunächst die Operanden einer Berechnung (z. B. die '2' und die '3' in der Berechnung des Ergebnisses von '2+3') eingegeben und erst anschließend der Operator (z. B. '+' bei der Berechnung von '*2+3*'). Im ALG-Modus hingegen geben Sie arithmetische Ausdrücke genauso ein, wie Sie sie auf Papier niederschreiben würden. Um die Berechnung durchzuführen benutzen wir die ENTER Taste. Diese Bedienungsanleitung erläutert Beispiele für die Anwendung der verschiedenen Funktionen und Berechnungsmöglichkeiten dieses Taschenrechners in beiden Modi.

Die Kapitel dieser Bedienungsanleitung sind nach Themen mit ansteigendem Schwierigkeitsgrad angeordnet. Sie beginnen mit der Einstellung der Betriebsarten des Taschenrechners und der Anzeigeoptionen und fahren fort mit Rechenoperationen mit realen und komplexen Zahlen, Operationen mit Listen, Vektoren und Matrixen, detaillierten Beispielen grafischer Anwendungen, der Benutzung von Strings, Grundlagen der Programmierung, Grafikprogrammierung, Bearbeitung von programmierten Schleifen, Kalkulus, Kalkulus-Anwendungen mit multiplen Variablen, Anwendungen mit Differentialgleichungen (einschließlich der Laplace-Transformation, Fourier-

Reihen und Fourier-Transformationen), Wahrscheinlichkeitsberechnungen und statistischen Anwendungen.

Das Herz des Taschenrechners ist ein erweiterbares Betriebssystem, dass Sie durch den Download neuer Versionen von der Website des Taschenrechners aktualisieren können. Für Operationen mit Symbolen verfügt der Taschenrechner über eine mächtiges Computer Algebra System (CAS), mit dem Sie verschiedene Betriebsarten wählen können, z. B. für komplexe Zahlen und reale Zahlen oder exakte und angenäherte Zahlendarstellungen. Das Display kann so eingestellt werden, dass es Ausdrücke ähnlich wie in einem Lehrbuch anzeigt, was bei der Arbeit mit Matrixen, Vektoren, Brüchen, Summen, Ableitungen und Integralen nützlich sein kann. Mit der Hochgeschwindigkeits-Grafikfähigkeit des Taschenrechners erzeugen Sie grafische Darstellungen bequem und in kürzester Zeit.

Dank der Infrarot-Schnittstelle und des RS 232-Kabels, das für Ihren Taschenrechner erhältlich ist, können Sie Ihren Taschenrechner mit anderen Taschenrechnern oder Computern verbinden. Durch die Infrarot- oder die RS 232-Hochgeschwindigkeitsverbindung können Sie schnell und effizient Daten und Programme mit anderen Taschenrechnern oder Computern austauschen. Der Taschenrechner verfügt über Anschlüsse für Flash-Memory-Card, um die Speicherung und den Austausch von Daten mit anderen Anwendern zu ermöglichen.

Die Programmierfähigkeiten des Taschenrechners ermöglichen Ihnen die Entwicklung von effizienten Anwendungen für spezielle Zwecke. Ob fortgeschrittene mathematische Anwendungen, spezifische Problemlösungen oder Datenaufzeichnung – die über Ihren Taschenrechner zur Verfügung gestellten Programmiersprachen machen ihn zu einem äußerst vielseitigen Gerät.

Wir hoffen, dass Ihnen ihr Taschenrechner ein treuer, zuverlässiger Begleiter sein wird, sowohl für Anwendungen in der Ausbildung als auch im Beruf.

Inhaltsverzeichnis

Anhänge

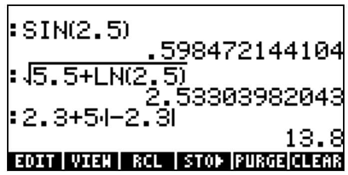
Hinweis zu Screenshots in dieser Anleitung

Ein Screenshot ist eine Dartsellung des Taschenrechner-Displays. Wenn der Taschenrechner beispielsweise das erste Mal eingeschaltet wird, wird das folgende Display angezeigt (in diesem Abschnitt werden die Displays des Taschenrechners mit einem dicken Rand dargestellt):

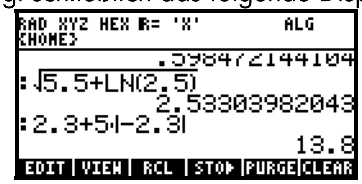
Die oberen beiden Zeilen stellen die Kopfzeile des Displays dar und der verbleibende Bereich wird für die Taschenrechnerausgabe verwendet.

Die meisten Screenshots in diesem Handbuch wurden mit Hilfe eines Computer-basierten Emulators generiert (ein Programm, das die Operation des Taschenrechnrs in einem Computer simuliert). In diesen Screenshots werden die Kopfzeilen des Displays nicht angezeigt. Stattdessen wird ein zusätzlicher Display-Ausgabebereich wie in der folgenden Abbildung angezeigt:

Dieser zusätzliche Display-Ausgabebereich wird in vielen Scrrenshots dieses Handbuchs nicht angezeigt, wenn Sie versuchen, die angeführten Beispiele auf Ihren Taschenrechner auszuführen. Daher wird Ihnen möglicherweise ein Screenshot wie der Folgende angezeigt:

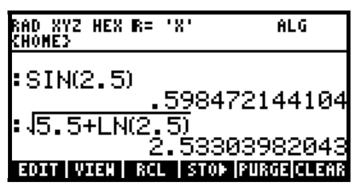


der Taschenrechner zeigt schließlich das folgende Display an:



Beachten Sie, dass die Kopfzeilen die obere Hälfte und die ersten Zeilen der Ausgabe des Taschenrechner-Displays abdecken. Sie können die nicht sichtbaren Zeilen der Ausgabe dennoch verwenden. Sie können auf diese Zeilen zugreifen, indem Sie die Aufwärtstaste (▲) betätigen, mit der Sie durch den Display-Inhalt blättern können.

Wenn Sie zudem die drei im Scrennshot aufgeführten Operationen in der angezeigten Reihenfolge ausführen, werden diese im Display wie folgt angezeigt:



Die zum Ausführen dieser Aufgaben erforderliche Tastatureingabe lautet wie folgt:

SIN 2 . 5 ENTER
√x ← () 5 . 5 + → LN 2 . 5 ENTER

Die nächste Operation,

2 . 3 + 5 × ← ABS +/- 2 . 3 ENTER

erzwingt, dass die der Operation SIN(2.5) entsprechenden Zeilen nach oben verschoben und von dern Kopfzeilen verdeckt werden.

Viele Screenshots in diesem Handbuch wurden zudem so geändert, dass sie nur die jeweils betreffende Operation anzeigen. Der Screenshot für die Operation SIN(2.5), oben dargestellt, kann in diesem Handbuch beispielsweise wie folgt vereinfacht dargestellt werden:

Diese Vereinfachungen der Screenshots dienen dazu, den Platz in diesem Handbuch besser zu nutzen.

Wenn Sie die Unterschiede der Screenshots in diesem Handbuch und der tatsächlichen Display-Anzeige berücksichtigen, sollten Sie die Übungen in diesem Handbuch problemlos ausführen können.

Kapitel 1
Einführung

Dieses Kapitel ist dazu gedacht, Ihnen Grundkenntnisse zur Bedienung Ihres Taschenrechners zu vermitteln. Die Beispiele dienen dazu, Sie mit den Grundoperationen und Einstellungen des Taschenrechners vertraut zu machen, bevor Sie mit den eigentlichen Berechnungen beginnen.

Grundoperationen

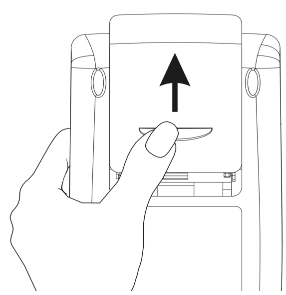
Nachfolgende Beispiele sollen Sie mit der Hardware des Taschenrechners vertraut machen.

Batterien

Für den Taschenrechner werden 3 AAA(LR03)-Batterien zur Hauptstromversorgung und eine CR2032 Lithiumbatterie für das Sichern des Speichers benötigt.
Bevor Sie den Taschenrechner in Betrieb nehmen, setzen Sie die Batterien wie folgt ein:

So installieren Sie die Hauptbatterien

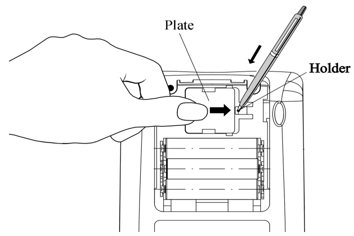
a. **Stellen Sie sicher, daß der Rechner ausgeschaltet ist.** Schieben Sie die Abdeckung des Batteriefachs wie abgebildet nach oben.



b. Legen Sie 3 neue AAA(LR03)-Batterien in das Hauptfach. Stellen Sie sicher, dass jede Batterie in der angegebenen Richtung eingelegt wird.

So installieren Sie die Batterie für das Backup des Speichers

a. **Stellen Sie sicher, daß der Rechner ausgeschaltet ist.** Drücken Sie die Halterung nach unten. Schieben Sie den Deckel in die angegebene Richtung, und heben Sie ihn an.



b. Setzen Sie eine neue CR2032-Lithium-Batterie ein. Stellen Sie sicher, dass der mit (+) gekennzeichnete Pol nach oben zeigt.
c. Setzen Sie den Deckel wieder auf, und schieben Sie ihn an die ursprüngliche Position zurück.

Nachdem Sie die Batterien installiert haben, drücken Sie [ON], um den Taschenrechner einzuschalten.

Warnung: Sobald das Symbol für niedrigen Batteriefüllstand angezeigt wird, sollten die Batterien so schnell wie möglich ausgetauscht werden. Um einen Datenverlust zu vermeiden, wechseln Sie niemals die Batterien für den Hauptstromanschluss und die Batterie für die Unterstützung des Datenspeichers gleichzeitig aus.

Ein- und Ausschalten des Taschenrechners

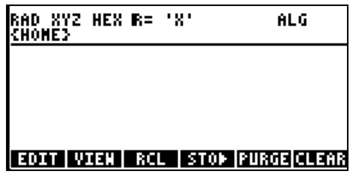
Die [ON]-Taste befindet sich links unten auf der Tastatur. Drücken Sie diese Taste einmal, um den Taschenrechner einzuschalten. Zum Ausschalten des Taschenrechners drücken Sie die rote rechte Umschalttaste [↱] (von unten die erste Taste in der zweiten Reihe) und anschließend die Taste [ON]. Beachten Sie, dass sich als Erinnerung an die Ausschaltfunktion des Taschenrechners auf der Taste [ON] eine rote Markierung OFF in der rechten oberen Ecke befindet.

Einstellen des Kontrasts für das Display

Der Kontrast des Displays kann mit den Tasten [+] und [−] bei gleichzeitig gedrückter [ON] Taste eingestellt werden. Die Tastenkombination [ON] (halten) [+] stellt das Display dunkler. Durch Drücken der Tastenkombination [ON] (halten) [−] wird das Display heller eingestellt.

Anzeigen im Display des Taschenrechners

Schalten Sie Ihren Taschenrechner erneut ein. Das Display sollte wie folgt aussehen:



Im oberen Teil des Displays erscheinen zwei Zeilen mit den Einstellungen des Taschenrechners. In der ersten Zeile erscheinen folgende Zeichen:

RAD XYZ HEX R= 'X'

Details über die Bedeutung dieser Angaben finden Sie in Kapitel 2.
In der zweiten Zeile erscheinen nachfolgende Zeichen: { HOME }, welche das HOME-Verzeichnis als aktuelles Verzeichnis im Speicher des Taschenrechners ausweisen. In Kapitel 2 erfahren Sie dann, dass Sie in Ihrem Taschenrechner Daten entweder in Dateien oder Variablen speichern können. Variablen können in Verzeichnissen und Unterverzeichnissen organisiert werden. Schließlich können Sie auch eine Verzeichnisstruktur erstellen, ähnlich der wie diese bei Computern verwendet wird. Sie können dann durch die Verzeichnisstruktur navigieren, um das gewünschte Verzeichnis auszuwählen. Während Sie sich durch die Verzeichnisstruktur bewegen, wird in der zweiten Zeile des Displays das entsprechende Verzeichnis zusammen mit dem Unterverzeichnis angezeigt.

Am unteren Teil des Displays befinden sich die beiden folgenden Beschriftungen:

EDIT VIEW RCL STO PURGE CLEAR

welche den *Funktionstasten* F1 bis F6 zugeordnet sind:

F1 F2 F3 F4 F5 F6

Die sechs Beschriftungen am unteren Rand des Taschenrechners befindlichen Beschriftungen wechseln abhängig vom aktuell angezeigten Menü. Die Funktionstaste F1 ist jedoch immer der ersten angezeigten Beschriftung zugeordnet, F2 der zweiten Beschriftung und so weiter.

Menüs

Die sechs den Funktionstasten F1 bis F6 zugeordneten Beschriftungen sind Befehle eines Funktionsmenüs. Von den sechs einer Taste zugeordneten Funktionen werden nur die ersten vier auf der Tastatur selbst angezeigt. Ein Menü kann aber auch mehr als nur sechs Einträge besitzen. Eine Gruppe von 6 Einträgen wird als Menüseite bezeichnet. Das aktuelle Menü (das TOOL-Menü, siehe unten), verfügt über acht Einträge, die auf zwei Seiten angeordnet sind. Die zweite Seite mit den letzten beiden Einträge kann durch Drücken der Taste NXT (Menü NeXT – nächstes Menü) angezeigt werden. Auf der Tastatur ist dies die dritte Taste von links in der dritten Reihe von unten. Ein weiteres Drücken der Taste NXT oder TOOL (dritte Taste in der zweiten Reihe von oben) bringt Sie zurück ins Hauptmenü TOOL.

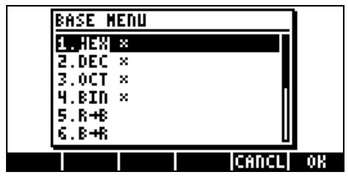
Das TOOL-Menü wird im nächsten Abschnitt behandelt. An dieser Stelle möchten wir einige Menüeigenschaften zeigen, welche Ihnen bei der Benutzung Ihres Taschenrechners hilfreich sind.

SOFT-Menüs vs. CHOOSE boxes

Menüs (auch als SOFT-Menüs bezeichnet) werden zusammen mit den sechs am unteren Bildschirmrand befindlichen Funktionstasten bedient (F1 bis F6). Durch Drücken der entsprechenden Funktionstaste wird die Funktion mit der entsprechenden Beschriftung aktiviert. So kann z. B. durch Drücken der Taste CLEAR (F6) bei aktiviertem TOOL-Menü die Funktion CLEAR ausgeführt werden, die den Inhalt der Anzeige löscht. Um diese Funktion auszuprobieren, geben Sie eine Zahl ein, beispielsweise 1 2 3 ENTER, und drücken Sie anschließend die Taste F6 .

Die SOFT-Menüs werden normalerweise zur Auswahl aus einer bestimmten Menge zusammengehöriger Funktionen eingesetzt. SOFT-Menüs sind aber nicht der einzige Zugriff auf Gruppen von verwandten Funktionen. Die Alternative dazu sind die sogenannten CHOOSE boxes. Um ein Beispiel eines

CHOOSE boxes zu sehen, aktivieren Sie das TOOL-Menü (durch Drücken der Taste TOOL), und anschließend verwenden Sie die Tastenkombination ⮣ BASE (der Taste 3 zugeordnet). Das folgende CHOOSE box wird angezeigt:



Dieses CHOOSE box ist als BASE-Menü (Basismenü) beschriftet und stellte eine durchnummerierte Liste von Funktionen zur Verfügung, von 1. HEX x bis 6. B→R. Diese Anzeige stellt die erste Seite dieses CHOOSE box Menüs dar und zeigt 6 Menüfunktionen. Mit den Pfeiltasten ▲▼, welche sich im oberen Teil der Tastatur unmittelbar unter den Funktionstasten F5 und F6 befinden, können Sie durch das Menü navigieren. Um eine der Funktionen zu starten, markieren Sie diese Funktion mithilfe der Pfeiltasten ▲▼ oder einfach durch Drücken der dem CHOOSE box zugeordneten Zahl. Nachdem Sie die Funktion ausgewählt haben, drücken Sie die Taste OK, Funktionstaste (F6). Möchten Sie z. B. die Funktion R→B (reell nach binär) auswählen, könnten Sie die Tastenfolge 6 F6 drücken.

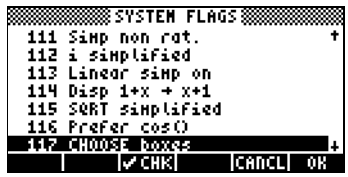
Wenn Sie direkt an den Anfang der Menüseite in ein CHOOSE box springen möchten, geschieht dies mit der Tastenfolge ⮢▲. Um ans Ende der aktuellen Seite zu gelangen, verwenden Sie die Tastenfolge ⮢▼. Um an den absoluten Anfang des Menüs zu gelangen, drücken Sie die Tastenfolge ⮣▲. Um ans absolute Ende des Menüs zu gelangen, drücken Sie die Tastenfolge ⮣▼.

Auswahl von SOFT-Menüs oder CHOOSE boxes

Sie können das Format, in dem Menüs angezeigt werden sollen, durch ändern eines Systemflags im Taschenrechner selbst bestimmen (Ein Systemflag ist eine Variable des Taschenrechners, welche einen bestimmten Operationsmodus des Taschenrechners überwacht. Weitere Informationen dazu erhalten Sie in Kapitel 24.) Durch Umstellen des Systemflags 117 können Sie die Anzeige von SOFT-Menü auf CHOOSE boxes ändern. Um zu diesem Flag zu gelangen, verwenden Sie die Tastefolge:

MODE FLAGS ▲ ← ▲▼

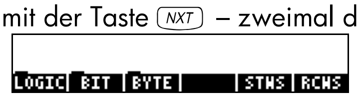
Auf Ihrem Display erscheint die folgende Anzeige, wobei die Zeile, beginnend mit der Nummer 117, hervorgehoben ist:



Standardmäßig wird die Zeile wie oben aussehen. Die hervorgehobene Zeile (117 CHOOSE boxes) zeigt an, dass Ihre Anzeige im Display im Moment auf CHOOSE boxes steht. Bevorzugen Sie aber die Verwendung von Funktionstasten, drücken Sie ✓CHK, Funktionstaste (F3) gefolgt von OK (F6). Drücken Sie OK (F6) ein weiteres Mal, um zur Normalanzeige des Taschenrechners zurückzukehren.

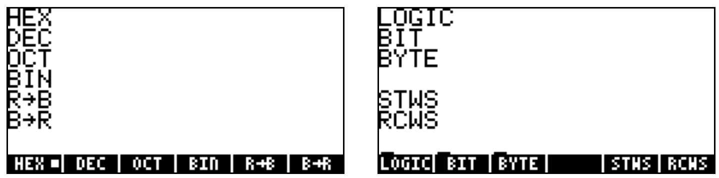
Wenn Sie die Tastenfolge → BASE, anstelle der CHOOSE boxes, die Sie vorhin gesehen haben, drücken, weist Ihre Anzeige sechs Einträge als erste Seite des STACK-Menüs auf:

Um zwischen den einzelnen Funktionen des Menüs zu navigieren, drücken Sie die Taste NXT, um auf Seite zwei des Menüs zu gelangen oder ← PREV (der Taste NXT zugeordnet), um zur vorherigen Seite zu gelangen. Die nachfolgenden Zahlen zeigen verschiedene Seiten des BASE-Menüs, auf welche mit der Taste NXT – zweimal drücken – zugegriffen werden kann.



Ein weiteres Drücken der Taste NXT, bringt uns auf die erste Menüseite zurück.

Anmerkung: Sobald das Systemflag 117 auf SOFT-Menü gesetzt ist, erhalten Sie über die Tastenkombination ⮣(halten) ▼, eine Liste der Funktionen aus dem aktuellen Menü. So z. B. erhalten Sie für die ersten beiden Seiten im BASE-Menü folgendes:



Um die Einstellung auf CHOOSE boxes zurückzustellen, verwenden Sie die Tastefolge:

MODE FLAGS ▲ ⮢ ▲ ▼ ✓CHK OK OK.

Anmerkungen:
1. Das TOOL-Menü, welches Sie durch Drücken der Taste TOOL erhalten, wird immer ein SOFT-Menü erzeugen.
2. Die meisten Beispiele dieser Anleitung zeigen beide Varianten SOFT-Menüs und CHOOSE boxes. Für Programmieranwendungen (Kapitel 21 und 22) verwenden Sie ausschließlich SOFT-Menüs.
3. Zusätzliche Informationen zu SOFT-Menüs vs. CHOOSE boxes erhalten Sie in Kapitel 2 dieser Anleitung.

Das Menü TOOL

Die Funktionstasten für das gegenwärtig angezeigte Menü, bekannt als das TOOL-Menü, sind Operationen zur Manipulation von Variablen zugeordnet (siehe zusätzliche Seiten über Variablen):

EDIT F1 EDIT – bearbeitet den Inhalt einer Variablen (siehe Kapitel 2 und Anhang L für weitere Informationen zu diesem Thema)
VIEW F2 VIEW – zeigt den Inhalt einer Variablen
RCL F3 ReCaLl – holt den Inhalt der Variablen zurück
STO F4 STOre – speichert den Inhalt einer Variablen
PURGE F5 PURGE – löscht (bereinigt) eine Variable

CLEAR F6 CLEAR – löscht das Display oder den Stack

Der Taschenrechner hat nur insgesamt sechs Funktionstasten, deshalb können jeweils lediglich 6 Beschriftungen gleichzeitig angezeigt werden. Ein Menü kann aber auch mehr als nur sechs Einträge besitzen. Eine Gruppe von 6 Einträgen wird als Menüseite bezeichnet. Eigentlich hat das TOOL-Menü acht Einträge, aufgeteilt auf zwei Seiten. Die zweite Seite, in welchem die letzten beiden Einträge enthalten sind, kann durch Drücken der Taste NXT (NeXT-Menü – nächstes Menü) angezeigt werden. Auf der Tastatur ist dies die dritte Taste von links in der dritten Reihe von unten.
In diesem Fall sind nur den ersten beiden Funktionstasten Befehle zugeordnet. Diese Befehle lauten:

CASCM F1 CASCMD: Der Befehl CAS CoMmanD, der dazu verwendet wird, einen Befehl aus dem CAS aus einer Liste zu starten

HELP F2 Die Hilfefunktion zur Beschreibung der zur Verfügung stehenden Befehle, kann über die Funktionstaste NXT erreicht werden und wird das Original TOOL-Menü anzeigen. Sie können auch zum TOOL-Menü zurückkehren, indem Sie die Taste TOOL (zweite Reihe von oben, dritte Taste von links) drücken.

Datum und Uhrzeit einstellen

Der Taschenrechner besitzt eine interne Echtzeituhr. Diese Uhr kann im Display ständig angezeigt werden und kann sowohl für Alarme oder für geplante laufende Aufgaben eingesetzt werden. Dieser Abschnitt zeigt nicht nur wie Sie Uhrzeit und Datum einstellen können, sondern auch die Grundlagen für die Anwendung von CHOOSE boxes und Dateneingabe in ein Dialogfeld. Die Dialogfelder Ihres Taschenrechners entsprechen denen eines Computers.

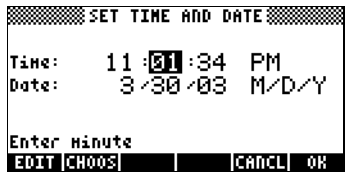
Um Datum und Uhrzeit einzustellen, verwenden Sie das CHOOSE box TIME, alternativ zur Funktion der Taste 9. Kombinieren Sie die rote rechte Shift-Taste ↱ mit der Taste 9, wird das CHOOSE box TIME aktiviert. Dieser Vorgang kann auch als ↱ TIME dargestellt werden. Die Abbildung unten zeigt das CHOOSE box für TIME:

Wie oben bereits erwähnt stellt das TIME-Menü vier verschiedene Optionen, durchnummeriert von 1 bis 4, zur Verfügung. An dieser Stelle ist für uns nur Option *3. Set time, date...* (Datum und Uhrzeit einstellen) von Interesse. Heben Sie mithilfe der Pfeiltaste ▼ heben diese Option hervor, und drücken Sie anschließend die Funktionstaten OK F6. Im Anschluss daran erhalten Sie die Eingabemaske (siehe Anhang 1-A) zur Einstellung von Uhrzeit und Datum:

Einstellen der Uhrzeit

Mit den Zahlentasten (1 2 3 4 5 6 7 8 9 0) können Sie mit das Einstellen der Stunden durchführen. Wenn Sie die Uhrzeit auf 11 ändern, indem Sie 1 1 drücken, sobald das Feld SET TIME AND DATE hervorgehoben ist, wird die Zahl 11 wird in der unteren Zeile der Eingabemaske angezeigt.

Um die Änderung vorzunehmen, drücken Sie die Funktionstaste OK F6. Im Stundenfeld erscheint nun die *11*, gleichzeitig wird das Minutenfeld hervorgehoben:



Ändern Sie nun das Minutenfeld auf 25 durch Drücken der Tasten 2 5 OK . Nun wird das Sekundenfeld hervorgehoben. Um dieses Feld auf 45 zu ändern, geben Sie 4 5 OK ein.

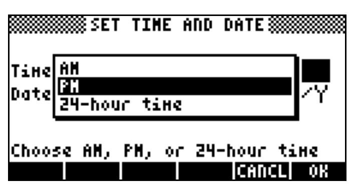
Nun wird Feld für das Zeitformat hervorgehoben. Um die aktuellen Einstellungen des Feldes zu ändern, können Sie entweder die Taste +/- (zweite Taste von links, fünfte von unten) oder die Funktionstaste CHOOS (F2) drücken.

- Benutzen Sie die Taste +/-, wird sich das Feld für das Zeitformat in eine der nachfolgenden Optionen ändern.

    - AM : zeigt an, dass es sich um eine Uhrzeit vor Mittag handelt, d. h. AM
    - PM : zeigt an, dass es sich um eine Uhrzeit nach Mittag handelt, d. h. PM
    - 24-hr: zeigt an, dass die Uhrzeit im 24 Stundenformat angezeigt wird wobei z. B. 18:00 h, 6pm darstellt

  Die zuletzt ausgewählte Option wird, sobald Sie nachfolgende Prozedur anwenden, eingestellt.

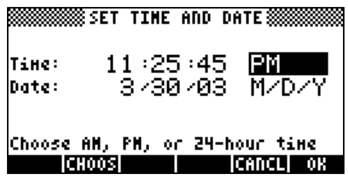
- Wenn Sie dabei die Funktionstaste CHOOS benutzen, stehen Ihnen folgende Optionen zur Auswahl:



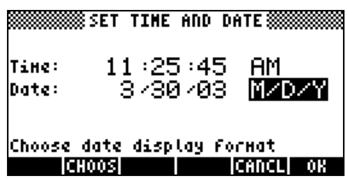
Benutzen Sie die Pfeiltasten, ▲ ▼, um zwischen diesen drei Optionen (AM, PM und 24-h) auszuwählen. Um Ihre Auswahl zu bestätigen, drücken Sie die Funktionstaste OK F6 .

Einstellen des Datums

Nachdem Sie das Format der Uhrzeit ausgewählt haben, wird die Eingabemaske SET TIME AND DATE wie folgt aussehen:



Um das Datum einzustellen, müssen Sie zunächst das Datumsformat auswählen. Das standardmäßig eingestellte Format lautet M/D/Y (Monat/Tag/Jahr). Um dieses Format zu ändern, drücken Sie die Pfeiltaste nach unten. Das Datumsformat wird wie unten angezeigt hervorgehoben:

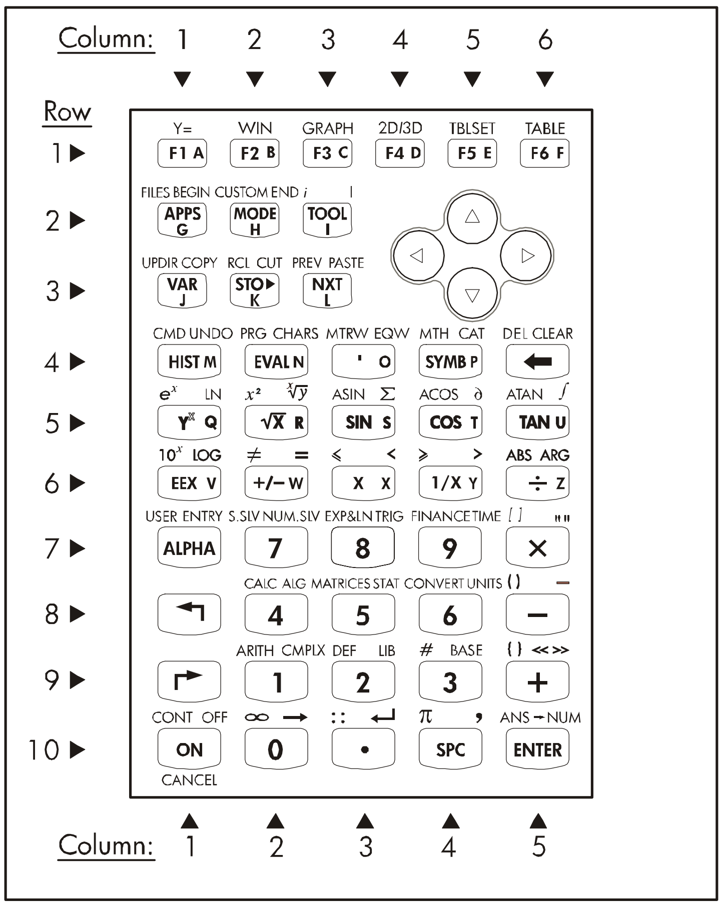


Benutzen Sie die Funktionstaste CHOOS (B), um die verschiedenen Optionen für das Datumsformat anzeigen zu lassen:

Mit den Pfeiltasten heben Sie die gewünschte Auswahl hervor und drücken anschließend die Funktionstaste OK F6, um diese zu übernehmen.

Einführung in die Tastatur des Taschenrechners

Die nachfolgende Abbildung zeigt ein Diagramm der Tastatur Ihres Taschenrechners mit nummerierten Zeilen und Spalten.

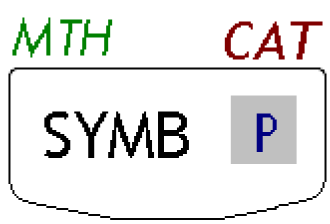


Die Abbildung zeigt 10 Reihen von Tasten mit 3, 5 oder 6 Spalten. Reihe 1 hat 6 Tasten, Reihe 2 und 3, jeweils 3 Tasten, während Reihe 4 bis 10 jeweils 5 Tasten aufweisen. In der rechten oberen Ecke, in Höhe der Reihen 2 und 3, befinden sich 4 Pfeiltasten.

Jeder einzelne Taste hat drei, vier oder fünf verschiedene Funktionen. Die Hauptfunktion der Taste entspricht der auf der Tastatur hervorgehobenen (größten) Beschriftung. Auch die grüne linke *Shift-Taste (8,1),* die rote rechte *Shift-Taste (9,1)* und die blaue ALPHA-*Taste (7,1)*, können mit anderen Tasten kombiniert werden, um alternative Funktionen, die auf der Tastatur angezeigt werden, zu starten. So hat z. B. die SYMB Taste *(4,4),* die folgenden sechs Funktionen, die dieser wie folgt zugeordnet sind:

SYMB	Hauptfunktion, Starten des SYMB-Menüs (SYMBolic – symbolisch)
← MTH	Linke-Shift Funktion zum Starten des MTH (Math)-Menüs
→ CAT	Rechte-Shift Funktion zum Starten der Funktion CATalog (Katalog)
ALPHA P	ALPHA Funktion, um den Großbuchstaben P einzufügen
ALPHA ← P	ALPHA linke-Shift Funktion, um den Kleinbuchstaben P einzufügen
ALPHA → P	ALPHA rechte-Shift Funktion, um das Symbol P einzufügen

Von den sechs dieser Taste zugeordneten Funktionen werden nur vier auf der Taste selbst angezeigt. So sieht Ihre Tastenanzeige auf der Tastatur aus:



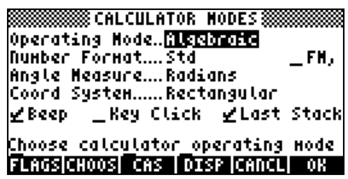
Beachten Sie, dass Farbe und Position der Beschriftung auf der Taste (**SYMB**, *MTH*, *CAT* und **P**) anzeigen, welches die Hauptfunktion (**SYMB**) ist, und welche der drei weiteren Funktionen der jeweiligen Tastenkombination linke-Shift Taste ← (*MTH*), rechte-Shift Taste → (*CAT*), und ALPHA (**P**) zugeordnet ist.

Weitere Informationen zur Tastatur des Taschenrechners finden Sie in Anhang B.

Auswahl der Taschenrechnermodi

In diesem Abschnitt wird vorausgesetzt, dass Sie mindestens teilweise im Umgang mit Auswahl- und Dialogboxen vertraut sind (sollten Sie es nicht sein, lesen Sie erst Kapitel 2 in dieser Anleitung nach).

Drücken Sie die Schaltfläche MODE (zweite Taste von links, zweite Reihe von oben), um die folgende CALCULATOR MODES- (Taschenrechnermodi) Eingabemaske zu erhalten:



Drücken Sie die Funktionstaste OK F6, um zur Normalanzeige zurückzukehren. Nachfolgend einige Beispiele, wie verschiedene Taschenrechnermodi ausgewählt werden können.

Operationsmodus

Der Taschenrechner bietet zwei verschieden Operationsmodi: den *algebraischen* (ALG)-Modus und den *Reverse Polish Notation* (RPN)-Modus. Standardmäßig ist der algebraische Modus (wie in der obigen Abbildung gezeigt) eingestellt, aber Anwender früherer HP-Taschenrechner sind möglicherweise mit dem RPN-Modus besser vertraut.

Um einen Operationsmodus auszuwählen, öffnen Sie zuerst die CALCULATOR MODES-Eingabemaske durch Drücken der Schaltfläche MODE. Das Feld *Operating Mode* (Operationsmodus) wird hervorgehoben. Wählen Sie nun den *ALG*- oder *RPN*-Modus durch Drücken der Taste +/- (zweite von Links, Reihe fünf von unten) oder durch Drücken der Funktionstaste CHOOS (F2) aus. Sollten Sie letztere zur Auswahl drücken, benutzen Sie die Pfeiltasten (▲

▼) zur Auswahl des entsprechenden Modus und drücken anschließend die Funktionstaste OK, um den Vorgang abzuschließen.

Um den Unterschied zwischen diesen beiden Operationsmodi zu veranschaulichen, berechnen wir nachfolgenden Ausdruck in beiden Modi:

$$\sqrt{\frac{3 \cdot \left(5 - \frac{1}{3 \cdot 3}\right)}{23^3} + e^{2.5}}$$

Um diesen Ausdruck in den Taschenrechner einzugeben, verwenden wir zuerst den *Equation Writer (Gleichungsschreiber)*, ↱ EQW . Sie benötigen die folgende Tasten neben den numerischen Tasten des Tastenfeldes:

↰ ↱ • SPC × + − ÷ √x
y^x e^x () ↱ EQW ◄ ► ▼ ▲ ENTER

Der EquationWriter ist ein Anzeigemodus, in welchem Sie mathematische Ausdrücke über explizite mathematische Bezeichnung, einschließlich Brüche, Ableitungsfunktionen, Integrale, Wurzeln usw. bilden können. Um mit dem EquationWriter den oben angegebenen Ausdruck einzugeben, benutzen Sie die Tastenfolge:

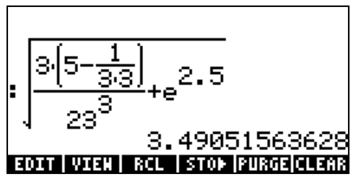
↱ EQW √x 3 × ↰ () 5 −
1 ÷ 3 × 3
▲ ▲ ▲ ▲ ▲ ▲ ▲
÷ 2 3 y^x 3 ► ► + ↰ e^x 2 • 5 ENTER

Nachdem Sie die Taste ENTER drücken, wird im Display des Taschenrechners, nachfolgender Ausdruck angezeigt:

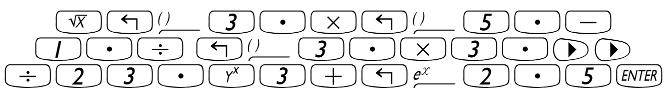
```
√(3*(5-1/(3*3))/23^3+EXP(2.5))
```

Durch erneutes Drücken der Taste ENTER, wird folgender Wert ausgegeben. Wenn Sie gefragt werden, akzeptieren Sie Approx mode (ungefährer Modus), indem Sie OK drücken. [**Anmerkung**: Die oben verwendeten Integerwerte,

z. B. 3, 5, 1 stellen exakte Werte dar. EXP(2,5) hingegen kann nicht als exakter Wert dargestellt werden, daher ist ein Umschalten in den Approx (Näherungs)-Modus erforderlich]:



Sie könnten den Ausdruck jedoch auch direkt, ohne Verwendung des EquationWriter wie folgt eingeben:



Sie erhalten dasselbe Ergebnis.

Ändern Sie nun den Modus auf RPN , indem Sie zuerst die Schaltfläche MODE drücken. Wählen Sie den *RPN*-Modus entweder über die Taste +/- oder durch Drücken der Funktionstaste CHOOS. Drücken Sie die Funktionstaste OK F6 , um den Vorgang abzuschließen. Die Anzeige für den RPN-Modus sieht wie folgt aus:

Beachten Sie, dass im Display die verschiedenen Ausgabeebenen, von unten nach oben mit 1, 2, 3, usw. durchnummeriert, angezeigt werden. Dies ist der sogenannte *Stack* (*Stapel*) des Taschenrechners. Die verschiedenen Ebenen werden als *Stack-Ebenen* bezeichnet , d. h. Stack-Ebene 1, Stack-Ebene 2, usw.

Grundsätzlich bedeutet RPN dass Sie, wenn Sie z. B. die Operation 3 + 2 in den Taschenrechner eingeben möchten, nicht die Tastenfolge

3 + 2 ENTER benutzen, sondern erst die Operanden in der richtigen Reihenfolge, dann erst den Operator, d. h. 3 ENTER 2 ENTER + eingeben. Während Sie die Operanden , eingeben, erscheinen diese in unterschiedlichen Stack-Ebenen. Geben Sie 3 ENTER ein, erscheint die 3 in Stack-Ebene 1. Geben Sie als Nächstes 2 ENTER ein, wird die 3 eine Ebene nach oben, d. h. in Stack-Ebene 2 verschoben. Geben wir schließlich + ein, sagen wir dem Taschenrechner, dass er den Operatoroder das Programm + auf die Objekte aus Stack-Ebene 1 und 2 anwenden soll. Das Ergebnis, 5, wird in Stack-Ebene 1 angezeigt. Eine Abkürzung für diese Operation lautet: 3 ENTER 2 + .
Versuchen wir einige weitere einfache Operationen, bevor wir uns an die komplizierteren, aus dem vorangegangenen algebraischen Modus wenden:

123/32	1 2 3 ENTER 3 2 ÷
4^2	4 ENTER 2 y^x
$^3\sqrt{27}$	2 7 ENTER 3 ↱ $\sqrt[x]{y}$

Beachten Sie die Position von y und x in den letzten beiden Operationen. Bevor die Taste y^x gedrückt wird, ist die Basis der Exponential-Operation y (Stack-Ebene 2) während der Exponent x (Stack-Ebene 1) ist. Ähnlich verhält es sich mit der Quadratwurzel, y (Stack-Ebene 2) ist die Zahl unter dem Wurzelzeichen während x (Stack-Ebene 1) die Wurzel selbst darstellt.

Versuchen Sie folgendes Beispiel, bei dem 3 Operanden verwendet werden:
$(5 + 3) \times 2$

5 ENTER 3 +	Berechnet zuerst den Wert (5+3).
2 ×	Vervollständigt die Berechnung.

Versuchen wir nun den weiter oben vorgeschlagenen Ausdruck:

$$\sqrt{\frac{3\cdot\left(5-\frac{1}{3\cdot 3}\right)}{23^3}+e^{2.5}}$$

3 . ENTER	Geben Sie 3 in Ebene 1 ein
5 . ENTER	Geben Sie 5 in Ebene 1 ein, 3 wird nach y verschoben
3 . ENTER	Geben Sie 3 in Ebene 1 ein, 5 wird in Ebene 2 und 3 in Ebene 3 verschoben
3 . ×	Setzen Sie 3 und das Multiplikationszeichen, 9 erscheint in Ebene 1
1/x	1/(3x3), letzter Wert in Ebene 1; 5 in Ebene 2, 3 in Ebene 3
−	5 - 1/(3×3) belegt nun Ebene 1, 3 ist in Ebene 2
×	3× (5 - 1/(3×3)) belegt nun Ebene 1.
2 3 . ENTER	Geben Sie 23 in Ebene 1 ein, 14,66666 wird in Ebene 2 verschoben.
3 . Y^x	Geben Sie 3 ein, berechnen Sie 23^3 in Ebene 1. 14,666 befindet sich in Ebene 2.
÷	(3× (5-1/(3×3)))/23^3 in Ebene 1
2 . 5	Geben Sie 2,5 in Ebene 1 ein
← e^x	$e^{2,5}$, erscheint in Ebene 1, in Ebene 2 wird der vorangegangene Wert angezeigt.
+	(3× (5 - 1/(3×3)))/23^3 + $e^{2,5}$ = 12.18369, in Ebene 1.
√x	√((3× (5 - 1/(3×3)))/23^3 + $e^{2,5}$) = 3,4905156, in Ebene 1.

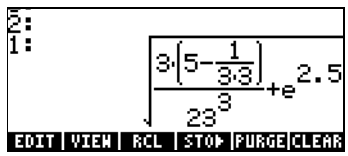
Obwohl der RPN-Modus anfänglich aufwändiger als ALG-Modus erscheint, gibt es vielfache Vorteile in der Verwendung des RPN-Modus. So z. B. können Sie im RPN-Modus sehen, wie sich die Gleichung Schritt für Schritt entfaltet. Dies ist beim Auffinden möglicher Eingabefehler nützlich. Sobald Sie ein wenig Übung in diesem Modus haben und einige Tricks herausfinden, werden Sie den gleichen Vorgang schneller und mit weniger Tastenanschlägen berechnen können. Nehmen wir z. B. die Berechnung von (4×6 - 5)/(1+4×6 - 5). Im RPN-Modus können Sie wie folgt eingeben:

4 ENTER 6 × 5 − ENTER 1 + ÷

offensichtlich, können Sie auch im RPN-Modus, einen Ausdruck in der gleichen Reihenfolge wie im algebraischen Modus, unter Benutzung des EquationWriters eingeben. Beispiel:

↱ EQW √x 3 . × ↰ () 5 . − 1 ÷ 3 . × 3 .
▲▲▲▲▲▲▲
÷ 2 3 . yˣ 3 ▶ ▶ + ↰ eˣ 2 . 5 ENTER

Der daraus resultierende Ausdruck wird in Stack-Ebene 1, wie folgt angezeigt:



Beachten Sie, dass der Ausdruck in Stack-Ebene 1 nach dem Drücken der Taste ENTER erscheint. Drücken Sie jedoch die Taste EVAL an dieser Stelle, wird der numerische Wert des Ausdrucks berechnet. Beachten Sie: Wenn Sie im RPN-Modus die Taste Enter drücken, wenn keine Kommandozeile vorhanden ist, wird dieser Befehl die Funktion DUP auslösen, welche den Inhalt der Stack-Ebene 1 eine Ebene nach oben, also nach 2 kopiert (genauso werden alle weiteren Ebenen eine Zeile nach oben verschoben). Dies ist sehr nützlich, wie Sie in den vorangegangenen Beispielen sehen konnten.

Um den Modus ALG vs. RPN auszuwählen, können Sie auch Systemflag 95 mit folgender Tastenfolge löschen/bereinigen:

MODE FLAGS ▲ ↰ ▲ ↰ ▲ ↰ ▲ ✓CHK OK OK

Alternativ dazu können Sie eine der folgenden Abkürzungen verwenden:

- Im ALG-Modus,
  CF(-95) wählt den RPN-Modus aus

- Im RPN-Modus,
  95 +/- ENTER SF wählt den ALG-Modus aus.

Weitere Informationen zu den Systemflags des Taschenrechners, finden Sie in Kapitel 2.

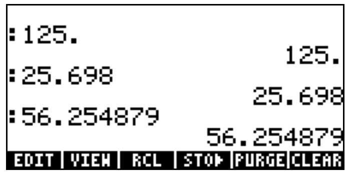
Zahlenformat und Dezimalpunkt oder Komma

Das Wechseln des Zahlenformates erlaubt Ihnen die benutzerspezifische Anpassung, reelle Zahlen im Taschenrechner anzuzeigen. Sie werden sehen, dass dies in Operationen mit Zehnerpotenzen äußerst nützlich ist, oder aber um die Dezimalstellen in einem Ergebnis einzuschränken.

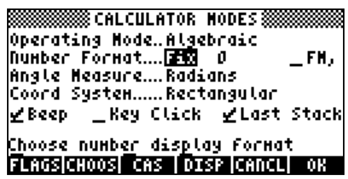
Um ein Zahlenformat auszuwählen, öffnen Sie zuerst die CALCULATOR MODES-Eingabemaske durch Drücken der Schaltfläche MODE. Benutzen Sie anschließend die Pfeiltaste ▼, um die Option *Number format* (Zahlenformat) auszuwählen. Der voreingestellte Wert ist *Std* oder *Standard* format. Im Standardformat werden Gleitpunktzahlen mit maximaler Genauigkeit (12 Wertziffern - Nachkommastellen), die der Taschenrechner erlaubt, angezeigt. Mehr über reelle Zahlen finden Sie in Kapitel 2. Um dieses und weitere Zahlenformate zu veranschaulichen, machen Sie die folgenden Übungen:

- **Standardformat**:
  Dies ist der am häufigsten verwendete Modus, weil dieser Modus Zahlen in der gängigsten Schreibweise anzeigt.
  Drücken Sie die Funktionstaste OK mit dem *Number format* (Zahlenformat) auf *Std* eingestellt, um dann zum Display des Taschenrechners zurückzukehren. Geben Sie die Zahl 123,4567890123456 ein. Beachten Sie, dass diese Zahl 16 Nachkommastellen beinhaltet. Drücken Sie die Taste ENTER. Die Zahl wird auf maximal 12 Wertziffern abgerundet und wird wie folgt angezeigt:

  In der Dezimalanzeige werden Integer-Zahlen trotzdem ohne Dezimalnullen angezeigt. Zahlen mit unterschiedlichen Nachkommastellen, werden im Display so ausgerichtet, dass lediglich die Dezimalstellen, die nötig sind, angezeigt werden. Nachfolgend weitere Beispiele von Zahlen im Standardformat:



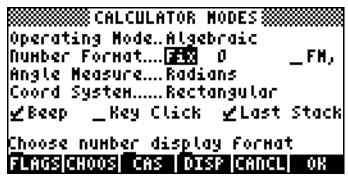
- **Feststehendes Format ohne Dezimalzahlen:** Drücken Sie die Schaltfläche MODE. Benutzen Sie anschließend die Pfeiltaste ▼, um die Option *Number format* (Zahlenformat) auszuwählen. Drücken Sie die Funktionstaste CHOOS (F2), und wählen Sie die Option *Fix* mit der Pfeiltaste ▼.



Beachten Sie, dass das Zahlenformat auf *Fix*, gefolgt von einer Null *(0)* gesetzt ist. Diese Zahl zeigt die Anzahl der Dezimalstellen, welche nach dem Dezimalkomma im Display des Taschenrechners angezeigt werden sollen. Drücken Sie die Funktionstaste OK, um zum Display des Taschenrechners zurückzukehren. Die Zahl wird nun wie folgt angezeigt:

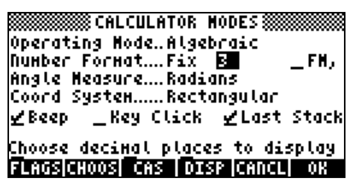
Durch diese Einstellung werden alle Ergebnisse auf die nächste Integer-Zahl abgerundet erzwungen (keine Nachkommastelle wird angezeigt). Im Taschenrechner selbst, wird diese Zahl mit allen 12 Nachkommastellen gespeichert. Je nachdem, wie wir die Anzahl der anzuzeigenden Dezimalstellen ändern, werden zusätzliche Ziffern erneut sichtbar.

- **Feststehendes Format mit Dezimalzahlen:**
  Dieser Modus wird bei der Arbeit mit begrenzter Präzision benutzt. So z. B., wenn Sie finanzmathematische Berechnungen, im FIX 2 Modus durchführen, ist es bequem, da man ganz einfach Währungseinheiten bis zu einer Präzision von 1/100 darstellen kann.
  Drücken Sie die Schaltfläche MODE. Benutzen Sie anschließend die Pfeiltaste , um die Option *Number format* (Zahlenformat) auszuwählen. Drücken Sie die Funktionstaste CHOOS (F2) und wählen Sie die Option *Fix* mit der Pfeiltaste .



Drücken Sie die Pfeiltaste, , um die Null vor der Option *Fix* hervorzuheben. Drücken Sie anschließend die Funktionstaste CHOOS, und wählen Sie mit den Pfeiltasten , sagen wir 3 Dezimalstellen aus.

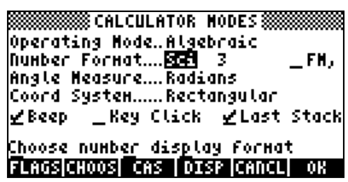
Drücken Sie die Funktionstaste OK, um die Auswahl abzuschließen.



Drücken Sie die Funktionstaste OK, um zum Display des Taschenrechners zurückzukehren. Die Zahl wird nun wie folgt angezeigt:

Beachten Sie, dass die Zahl nun abgerundet und nicht abgeschnitten ist. Somit wird die Zahl 123,4567890123456 in dieser Einstellung als 123,457 und nicht als 123,456 angezeigt, da die Nachkommastelle nach der 6 > 5 ist.

- **Wissenschaftliches Format**
  Das wissenschaftliche Format wird hauptsächlich zum Lösen von Problemen in der Physik, wo Zahlen gewöhnlich mit begrenzter Präzision, multipliziert mit einer Zehnerpotenz, angezeigt werden, benutzt.
  Zum Einstellen dieses Formats drücken Sie die Schaltfläche MODE. Benutzen Sie anschließend die Pfeiltaste ▼, um die Option *Number format* (Zahlenformat) auszuwählen. Drücken Sie die Funktionstaste CHOOS (F2) und wählen Sie die Option *Scientific* (wissenschaftlich) mit der Pfeiltaste ▼. Behalten Sie die Zahl 3 vor *Sci*. (Diese Zahl kann genauso geändert werden, wie wir dies im *Fixed* Zahlenformat mit den Dezimalstellen aus obigem Beispiel abgeändert haben.)

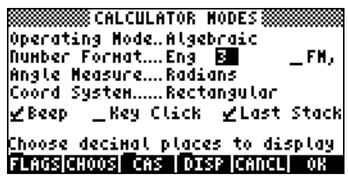


Drücken Sie die Funktionstaste OK, um zum Display des Taschenrechners zurückzukehren. Die Zahl wird nun wie folgt angezeigt:

Dieses Ergebnis, 1,23E2, ist die Darstellung einer Zehnerpotenz des Taschenrechners, d. h. $1{,}235 \times 10^2$. In dieser sogenannten wissenschaftlichen Schreibweise, stellt die dem *Sci*-Zahlenformat vorangestellte Ziffer 3 (wie vorher gezeigt) die Anzahl der Dezimalstellen nach dem Komma dar. Die wissenschaftliche Darstellung beinhaltet immer eine Ganzzahl (Integer), wie oben zu sehen ist. Deshalb ist in diesem Fall die Anzahl der Nachkommastellen vier.

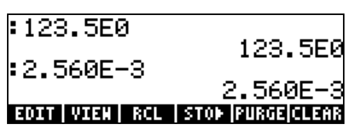
- **Technisches Format**

  Das technische Format ähnelt sehr dem wissenschaftlichen Format, mit der Ausnahme, dass die Zehnerpotenzen ein Vielfaches von drei sind. Zum Einstellen dieses Formates drücken Sie die Schaltfläche MODE. Benutzen Sie anschließend die Pfeiltaste ▼, um die Option *Number format* (Zahlenformat) auszuwählen. Drücken Sie die Funktionstaste CHOOS (F2), und wählen Sie mit der Pfeiltaste ▼ die Option *Engineering* (technisch). Behalten Sie die Zahl 3 vor dem *Eng*. (Diese Zahl kann genauso geändert werden, wie wir dies im *Fixed*-Zahlenformat mit den Dezimalstellen aus vorangegangenem Beispiel abgeändert haben.)

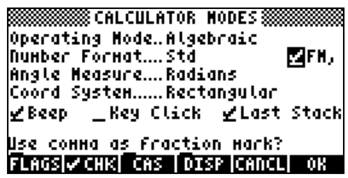


Drücken Sie die Funktionstaste OK, um zum Display des Taschenrechners zurückzukehren. Die Zahl wird nun wie folgt angezeigt:

Da diese Zahl drei Ziffern im Integer-Teil enthält, wird diese im technischen Format mit vier Wertstellen und einer Zehnerpotenz von Null angegeben. So z. B. wird die Zahl 0,00256 wie unten angezeigt:



- **Dezimalkomma vs. Dezimalpunkt**
  Dezimalpunkte in Gleitpunktzahlen können durch ein Komma ersetzt werden, wenn der Benutzer mit diesen besser vertraut ist. Um Dezimalpunkte durch Kommas zu ersetzen, ändern Sie die Option *FM* in der CALCULATOR MODES-Eingabemaske wie folgt auf Kommas (Beachten Sie, dass wir das *Zahlenformat* auf *Std* geändert haben):
- Drücken Sie die Schaltfläche MODE. Drücken Sie anschließend die Pfeiltaste ▼ einmal und die Pfeiltaste nach rechts, ▶, um die Option _ *FM* hervorzuheben. Um Kommas auszuwählen, drücken Sie die Funktionstaste ✓CHK (d. h. die Taste F2). Die Eingabemaske sieht wie folgt aus:
-



- Drücken Sie die Funktionstaste OK, um zum Display des Taschenrechners zurückzukehren. Die Zahl 123,456789012, die Sie vorher eingegeben haben, wird nun wie folgt angezeigt:
-

Winkelmaß

Trigonometrische Funktionen benötigen z. B. Argumente, die Flächenwinkel darstellen. Der Taschenrechner stellt drei verschiedene *Winkelmaß*-Modi für die Arbeit mit Winkeln zur Verfügung, und zwar:

- *Grade*: Ein kompletter Umfang beträgt *360* Grad (*360°*) oder *90* Grad (*90°*) in einem rechten Winkel. Diese Darstellung wird hauptsächlich in der Standardgeometrie, Maschinen- oder Stahlbau und im Vermessungswesen eingesetzt.
- *Radiane*: In einem kompletten Umfang sind 2π Radiane ($2\pi\ ^r$) enthalten bzw. $\pi/2$ Radiane ($\pi/2\ ^r$) in einem rechten Winkel. Diese Notation wird

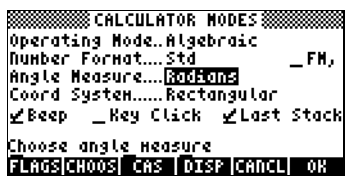
hauptsächlich bei der Lösung mathematischer oder physischer Probleme verwendet. Dies ist der Standardmodus des Taschenrechners.

- *Zentesimalgrade*: Ein kompletter Umfang beträgt *400* Zentesimalgrade (*400* ^{g}) oder *100* Zentesimalgrade (1*00* ^{g}) in einem rechten Winkel. Diese Notation ist ähnlich dem Gradmodus und war eigentlich zur "Vereinfachung" der Gradnotation gedacht, wird jedoch heute selten benutzt.

Das Winkelmaß beeinflusst die trigonometrischen Funktionen wie SIN, COS, TAN und mit ihnen in Zusammenhang stehende Funktionen.

Um in den Winkelmaßmodus zu wechseln, gehen Sie wie folgt vor:

- Drücken Sie die Schaltfläche MODE. Drücken Sie anschließend die Pfeiltaste ▼ zweimal. Wählen Sie nun den Winkelmaßmodus entweder durch Drücken der Taste +/- (zweite von links, Reihe fünf von unten) oder durch Drücken der Funktionstaste CHOOS (F2). Sollten Sie sich für letztere entscheiden, benutzen Sie die Pfeiltasten ▲▼ zur Auswahl des entsprechenden Modus und drücken anschließend die Funktionstaste OK F6, um den Vorgang abzuschließen. Im nachfolgenden Beispiel wurde der Radianmodus gewählt:



Koordinatensystem

Das Koordinatensystem beeinflusst die Eingabe- und Darstellungsart von Vektoren und komplexen Zahlen. Weitere Informationen über komplexe Zahlen und Vektoren erhalten Sie in den Kapiteln 4 und 9.

Zwei- und dreidimensionale Vektorkomponenten und komplexe Zahlen können in jedem der 3-Koordinatensysteme dargestellt werden: Das Kartesische (2-dimensional) oder das rechtwinklige (3-dimensional), das zylindrische (3-dimensional) oder das Polarsystem (2-dimensional) und das sphärische (nur 3-dimensional). In einem Kartesischen oder rechtwinkligen Koordinatensystem hat ein Punkt P drei lineare Koordinaten (x,y,z), gemessen vom Ursprung entlang von 3 zueinander senkrechten Achsen (im 2-D Modus wird z als 0 angenommen). In einem zylindrischen oder einem Polarsystem werden die Koordinaten eines Punktes von (r,θ,z) bestimmt, wobei r eine radiale Distanz, gemessen vom Ursprung auf die xy-Ebene darstellt, θ den Winkel, den diese radiale Distanz r mit der positiven Achse x bildet – gemessen als positive Richtung gegen den Uhrzeigersinn – und z das Gleiche wie die z-Koordinate in einem Kartesischen System darstellt (im 2D-Modus wird z als 0 angenommen). Für das rechtwinklige und das polare System gilt die durch die nachfolgenden Mengen angegebene Beziehung:

$$x = r \cdot \cos(\theta) \qquad r = \sqrt{x^2 + y^2}$$

$$y = r \cdot \sin(\theta) \qquad \theta = \tan^{-1}\left(\frac{y}{x}\right)$$

$$z = z$$

In einem sphärischen System werden die Koordinaten eines Punktes durch (ρ,θ,φ) bestimmt, wobei ρ eine radiale Distanz, gemessen vom Ursprung eines Kartesischen Systems, θ der Winkel, der den Winkel die durch die Projektion der linearen Distanz ρ auf die xy-Achse darstellt (genau wie θ in Polar-Koordinaten) und ϕ den Winkel von der positiven Achse z auf die radiale Distanz ρ darstellt. Für das rechtwinklige und das sphärische System gilt die durch die nachfolgenden Mengen angegebene Beziehung:

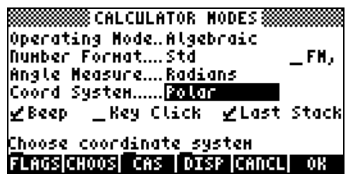
$$x = \rho \cdot \sin(\phi) \cdot \cos(\theta) \qquad \rho = \sqrt{x^2 + y^2 + z^2}$$

$$y = \rho \cdot \sin(\phi) \cdot \sin(\theta) \qquad \theta = \tan^{-1}\left(\frac{y}{x}\right)$$

$$z = \rho \cdot \cos(\phi) \qquad \phi = \tan^{-1}\left(\frac{\sqrt{x^2 + y^2}}{z}\right)$$

Um das Koordinaten-System in Ihrem Taschenrechner zu ändern, führen Sie folgende Schritte durch:

- Drücken Sie die Schaltfläche MODE. Drücken Sie anschließend die Pfeiltaste ▼ dreimal. Wählen Sie nun den Winkelmaßmodus entweder durch Drücken der Taste +/- (zweite von links, Reihe fünf von unten) oder durch Drücken der Funktionstaste CHOOS (F2). Sollten Sie letztere zur Auswahl drücken, benutzen Sie die Pfeiltasten ▲▼ zur Auswahl des entsprechenden Modus und drücken anschließend die Funktionstaste OK F6, um den Vorgang abzuschließen. Im nachfolgenden Beispiel wird der polare Koordinaten-Modus gewählt:



Beep (Piepsen), Key Click (Tastenklick) und Last Stack (letzter Stack)

Die letzte Zeile in der CALCULATOR MODES-Eingabemaske, enthält folgende Optionen:

_Beep _Key Click _Last Stack

Wählen Sie das Häkchen neben einer der Optionen aus, wird die entsprechende Option aktiviert. Diese Optionen werden nachfolgend beschrieben.

_Beep : Wenn ausgewählt, ist der Beeper des Taschenrechners aktiviert. Diese Operation dient hauptsächlich für Fehlermeldungen, verfügt jedoch über einige weitere Funktionen, wie: BEEP.

_Key Click : Wird diese Funktion ausgewählt, wird bei jedem Tastenanschlag ein "Klick" hörbar.

_Last Stack: Behält den Inhalt der letzten Stack-Eingabe für die Weiterverwendung mit den Funktionen UNDO und ANS (siehe Kapitel 2).

Die _*Beep* Option kann bei Fehlermeldungen für den Anwender nützlich sein. Diese Option sollten vor Betreten eines Klassenzimmers oder einer Bibliothek ausschalten, in dem Sie den Taschenrechner benutzen möchten.
Die _*Key Click* Option kann auch als hörbare Überprüfung für die Richtigkeit der Tastenanschläge dienen.
Die _*Last Stack* Option ist besonders dann vor Vorteil, wenn wir den letzten Vorgang rückgängig machen möchten, wenn dieser für eine weitere Berechnung benötigt wird.

Um eine dieser drei Optionen zu wählen bzw. abzuwählen, drücken Sie zuerst die Taste MODE. Als Nächstes

- Benutzen Sie die Pfeiltaste ▼ viermal, um die die Option _*Last Stack* auszuwählen. Verwenden Sie die Funktionstaste ✓CHK (d. h. die Taste F2), um die Auswahl zu ändern.
- Drücken Sie die Pfeiltaste ◄, um die Option _*Key Click* auszuwählen. Verwenden Sie die Funktionstaste ✓CHK (d. h. die Taste F2), um die Auswahl zu ändern.
- Drücken Sie die Pfeiltaste ◄, um die Option _*Beep* auszuwählen. Verwenden Sie die Funktionstaste ✓CHK (d. h. die Taste F2), um die Auswahl zu ändern.
  Drücken Sie die Funktionstaste OK F6, um den Vorgang abzuschließen.

Auswahl der CAS-Einstellungen

CAS steht für *C*omputer *A*lgebraic *S*ystem (Algebraisches System des Taschenrechners). Dies ist das mathematische Herzstück des Taschenrechners, in welchem die symbolischen mathematischen Operationen und Funktionen programmiert sind und auch ausgeführt werden. Das CAS bietet eine Reihe von Einstellungen, die nach Typ oder Operationsart eingestellt werden können. Dies sind:

- die unabhängige Standardvariable
- numerischer vs. symbolischer Modus
- Näherungs- vs. exakter Modus
- Verbose vs. Non-verbose Modus
- Einzelschrittmodus für Operationen
- aufsteigendes Potenzformat für Polynome

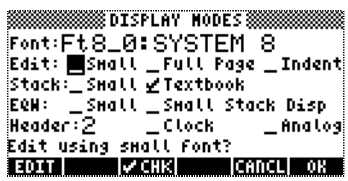
- genauer Modus
- Vereinfachung von irrationalen Ausdrücken

Weitere Details zur Auswahl der CAS Einstellungen finden Sie in Anhang C.

Auswahl der verschiedenen Anzeige-Modi

Durch Auswahl der verschiedenen Anzeigemodi kann das Display des Taschenrechners wie gewünscht angepasst werden. Um die möglichen Displayeinstellungen anzusehen, gehen Sie wie folgt vor:

- Drücken Sie die Schaltfläche MODE, um die CALCULATOR MODES-Eingabemaske zu starten. Innerhalb der CALCULATOR MODES-Eingabemaske drücken Sie die Funktionstaste DISP (F4), um die Eingabemaske DISPLAY MODES anzuzeigen.
-

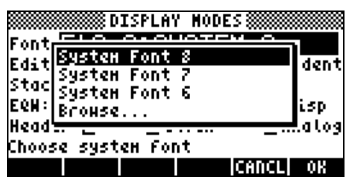


- Benutzen Sie die Pfeiltasten, um zwischen den einzelnen Optionen der DISPLAY MODES-Eingabemaske zu navigieren: ◄►▼▲.
- Um eine der obigen Einstellungen aus- oder abzuwählen, die die Auswahl eines Häkchens benötigen, wählen Sie zuerst den Unterstrich vor der von Ihnen gewünschten Option, und bewegen Sie anschließend die Funktionstaste ✓CHK solange, bis die gewünschte Einstellung erreicht ist. Sobald eine Option gewählt wurde, erscheint ein Häkchen über dem Unterstrich (d. h. im obigen Beispiel die Option *Textbook* im *Stack*: Linie oben). Nicht ausgewählte Optionen haben kein Häkchen über dem Unterstrich vor der gewünschten Option (z. B. die Optionen *_Small, _Full page* und *_Indent* bei *Edit*: Linie oben).
- Um die Schrift für das Display auszuwählen, heben Sie das Feld vor der Option *Font*: in der DISPLAY MODES-Eingabemaske vor, und benutzen Sie die Funktionstaste CHOOS (F2).
- Nachdem Sie nun alle gewünschten Optionen für die Eingabemaske des DISPLAY MODES ausgewählt oder abgewählt haben, drücken Sie die

Funktionstaste OK. So kehren Sie zur CALCULATOR MODES-Eingabemaske zurück. Um zur Normalansicht des Taschenrechners zurückzukehren, drücken Sie die Taste OK ein weiteres Mal.

Auswahl der Schrift im Display

Durch Veränderung der Schriftgröße, können Sie den Taschenrechner wie gewünscht einstellen. Wenn Sie z. B. eine 6-Pixel Schrift verwenden, können Sie in Ihrem Display bis zu 9 Stack Ebenen anzeigen. Folgen Sie diesen Anweisungen, um die Schrift für Ihr Display auszuwählen:
Drücken Sie die Schaltfläche MODE, um die CALCULATOR MODES-Eingabemaske zu starten. Innerhalb der CALCULATOR MODES-Eingabemaske drücken Sie die Funktionstaste DISP (F4), um die Eingabemaske DISPLAY MODES anzuzeigen. Das Feld *Font*: ist hervorgehoben und die Option *Ft8_0:system 8* ist ausgewählt. Dies ist die standardmäßig eingestellte Schrift. Wenn Sie nun die Funktionstaste CHOOS (F2) drücken, erhalten Sie eine Auflistung aller im System vorhandenen Schriften, wie unten angezeigt:



Die vorhandenen Optionen sind drei Standard *System Fonts* (Größe 8, 7 und 6) und eine *Browse…* (Such)-Option. Über letztere Option können Sie den Speicher des Taschenrechners nach weiteren Schriften durchsuchen, welche Sie möglicherweise selbst erstellt (siehe Kapitel 23) oder auf Ihren Taschenrechner aufgespielt haben.
Üben Sie das Ändern der Schriftart, indem Sie die Schrift des Taschenrechners von 7 auf 6 umstellen. Drücken Sie die Funktionstaste OK, um die Auswahl abzuschließen. Nachdem Sie nun eine Schrift ausgewählt haben, drücken Sie die Funktionstaste OK, um zur CALCULATOR MODES-Eingabemaske zurückzukehren. Um an dieser Stelle zur Normalansicht des Taschenrechner zurückzukehren, drücken Sie die Funktionstaste OK erneut und beachten Sie,

wie sich die Stack-Anzeige verändert, um sich der neuen Schriftart anzupassen.

Auswahl der Eigenschaften des Zeileneditors

Drücken Sie die Schaltfläche MODE, um die CALCULATOR MODES-Eingabemaske zu starten. Drücken Sie die Funktionstaste DISP (F4) innerhalb der CALCULATOR MODES-Eingabemaske, um die Eingabemaske DISPLAY MODES anzuzeigen. Drücken Sie die Pfeiltaste ▼ einmal, um zur *Edit* Zeile (Bearbeitungszeile) zu gelangen. Diese Zeile weist drei Merkmale auf, die verändert werden können. Sind diese Eigenschaften ausgewählt (mit einem Häkchen davor) sind folgende Effekte aktiviert:

_Small	Die Schrift wird kleiner dargestellt
_Full page	Erlaubt es, den Cursor ans Zeilenende zu bewegen
_Indent	Automatischer Zeileneinzug, wenn eine Zeilenschaltung vorgenommen wird

Genaue Anweisungen zur Anwendung des Zeileneditors finden Sie in Kapitel 2 dieser Anleitung.

Auswahl der Stack-Eigenschaften

Drücken Sie die Schaltfläche MODE, um die CALCULATOR MODES-Eingabemaske zu starten. Innerhalb der CALCULATOR MODES-Eingabemaske drücken Sie die Funktionstaste DISP (F4), um die Eingabemaske DISPLAY MODES anzuzeigen. Drücken Sie die Pfeiltaste ▼ zweimal, um zur *Stack*-Zeile zu gelangen. Diese Zeile weist zwei Einstellungen auf, die geändert werden können. Sind diese Eigenschaften ausgewählt (mit einem Häkchen gekennzeichnet), sind folgende Effekte aktiviert:

_Small	Die Schrift wird kleiner dargestellt. Maximiert die Anzahl der Informationen, die auf dem Display angezeigt werden. Beachten Sie, dass diese Auswahl die für die Stack-Anzeige ausgewählte Schriftart überschreibt.
_Textbook	Zeigt mathematische Ausdrücke in grafisch mathematischer Schreibweise an

Um diese Einstellungen zu veranschaulichen, wählen Sie entweder den algebraischen oder den PRN-Modus, und benutzen Sie den EquationWriter, um folgende bestimmte Integrale einzugeben:

[↱] EQW [↱] ∫ [0] [▶] [↰] ∞ [▶] [↰] e^x [+/-] [X] [▶] [X] [ENTER]

Im algebraischen Modus, wenn weder *_Small* noch *_Textbook* ausgewählt wurden, sieht die nachfolgende Ansicht für das Ergebnis dieser Eingabe wie folgt aus:

Wenn nur die Option *_Small* ausgewählt wurde, sieht die Eingabe wie folgt aus:

Ist aber die Option *_Textbook* ausgewählt (Standardwert), unabhängig davon, ob die Option *_Small* ausgewählt wurde, sieht das Ergebnis der Anzeige wie folgt aus:

Auswahl der Eigenschaften für den EquationWriter (EQW) (Gleichungseditor)

Drücken Sie die Schaltfläche [MODE], um die CALCULATOR MODES-Eingabemaske zu starten. Innerhalb der CALCULATOR MODES-Eingabemaske drücken Sie die Funktionstaste DISP ([F4]), um die Eingabemaske DISPLAY MODES anzuzeigen. Drücken Sie die Pfeiltaste [▼] dreimal, um zur Zeile EQW (EquationWriter = Gleichungseditor) zu gelangen. Diese Zeile weist zwei Einstellungen auf, die geändert werden können. Sind diese Eigenschaften ausgewählt (mit einem Häkchen gekennzeichnet), sind folgende Auswahlmöglichkeiten aktiviert:

_Small	Ändert die Schrift auf klein, während Sie den EquationEditor (Gleichungseditor) benutzen
_Small Stack Disp	Zeigt eine kleine Schriftart im Stack für die Anzeige im Format Textbook (Textbuch) an

Genaue Anweisungen zur Benutzung des EquationWriters (EQW) werden an anderer Stelle in dieser Anleitung beschrieben.

So wird z. B. die oben dargestellte Integrale, $\int_0^\infty e^{-X}\,dX$, wenn Sie *_Small Stack Disp im EQW in der Eingabemaske des DISPLAY MODES* auswählen, wie folgt dargestellt:

Auswahl der Größe für die Kopfzeile

Drücken Sie die Schaltfläche MODE, um die CALCULATOR MODES-Eingabemaske zu starten. Innerhalb der CALCULATOR MODES-Eingabemaske drücken Sie die Funktionstaste DISP (F4), um die Eingabemaske DISPLAY MODES anzuzeigen. Drücken Sie die Pfeiltaste ▼ viermal, um zur Zeile *Header* (Kopfzeile) zu gelangen. Dem Feld *Header* ist standardmäßig der Wert 2 zugewiesen. Dies bedeutet, dass der obere Teil des Displays zwei Zeilen enthält, eine, in der die aktuellen Einstellungen des Taschenrechners angezeigt werden und eine zweite, in der das aktuelle Unterverzeichnis im Speicher des Taschenrechners angezeigt wird (Diese Zeilen wurden in einem vorangegangenen Abschnitt der Anleitung beschrieben). Der Anwender kann diese Einstellungen auf 1 oder 0 setzen, um die Anzahl der Kopfzeilen im Display zu verringern.

Auswahl der Anzeige für die Uhr

Drücken Sie die Schaltfläche MODE, um die CALCULATOR MODES-Eingabemaske zu starten. Innerhalb der CALCULATOR MODES-Eingabemaske drücken Sie die Funktionstaste DISP (F4), um die Eingabemaske DISPLAY MODES anzuzeigen. Drücken Sie die Pfeiltaste ▼ viermal, um zur Zeile *Header* (Kopfzeile) zu gelangen. Das Feld *Header*

(Kopfzeile) wird hervorgehoben. Benutzen Sie die rechte Pfeiltaste (▶), um den Unterstrich vor der Option *_Clock* oder *_Analog* auszuwählen. Drücken Sie die Funktionstaste ✓CHK solange, bis die gewünschte Einstellung erreicht ist. Ist die Auswahl *_Clock* hervorgehoben, wird die Uhrzeit und das Datum in der rechten oberen Ecke des Displays angezeigt. Ist auch die Option *_Analog* ausgewählt, wird eine analoge anstelle einer digitalen Uhr angezeigt. Ist die Option *_Clock* nicht ausgewählt, oder die Kopfzeile nicht sichtbar bzw. zu klein, wird das Datum und die Uhrzeit im Display nicht angezeigt.

Kapitel 2
Einführung in den Taschenrechner

In diesem Kapitel werden eine Reihe von Basisoperationen des Taschenrechners erläutert, einschließlich der Anwendung des EquationWriters und der Manipulation von Datenobjekten im Taschenrechner. Studieren Sie die Beispiele in diesem Kapitel genau, um die Möglichkeiten des Taschenrechners für zukünftige Anwendungen genau zu erfassen.

Taschenrechner-Objekte

Alle Zahlen, Ausdrücke, Zeichen, Variablen usw., die im Taschenrechner erstellt oder manipuliert werden können, werden als Objekt bezeichnet. Einige der nützlichsten Objekttypen werden nachfolgend aufgelistet.

Reelle Zahlen. Diese Objekte stellen eine positive oder negative Zahl mit 12 Nachkommastellen und einem Exponenten zwischen -499 und +499 dar. Beispiele von reellen Zahlen sind: 1, -5, 56,41564 1,5E45, -555,74E-95

Bei der Eingabe einer reellen Zahl können Sie die Taste EEX zum Eintragen des Exponenten und die Taste +/- zum Ändern des Vorzeichens des Exponenten oder der Mantisse benutzen.

Beachten Sie, dass die reelle Zahl immer mit einem Dezimalkomma eingegeben werden muss, auch wenn die Zahl keine Nachkommastellen besitzt. Andernfalls wird die Zahl als Integer betrachtet, die jedoch ein anderes Objekt des Taschenrechners darstellt. Reelle Zahlen verhalten sich, wie Sie es von einer Zahl in einer mathematischen Operation erwarten würden.

Integer-Zahlen. Diese Objekte stellen Integer-Zahlen dar (Ganzzahlen ohne Dezimalstellen) und haben keine Grenzen (ausgenommen die des Speichers im Taschenrechner). Beispiele von Integer Zahlen sind: 1, 564654112, -4131654673546467654654 87. Beachten Sie, dass diese Zahlen kein Dezimalkomma enthalten.

Wegen ihres Speicherformats behalten Integer-Zahlen immer deren volle Genauigkeit bei der Berechnung. Wird beispielsweise die Operation 30/14 mit Integer-Zahlen durchgeführt, erhalten Sie das Ergebnis 15/7 und nicht

2,142. Um ein Ergebnis als Realzahl (Gleitkommazahl) zu erzwingen, benutzen Sie die Funktion →NUM ‹→NUM .

Integer-Zahlen werden wegen Ihrer vollen Genauigkeit in Rechenoperationen häufig in CAS-basierten Funktionen verwendet.

Wird im CAS (siehe Anhang C) der APPROX (Näherungs) ausgewählt, werden Integer-Zahlen automatisch in reelle Zahlen umgewandelt. Sollten Sie nicht planen, das CAS zu benutzen, wird empfohlen, gleich in den Näherungsmodus zu wechseln. Weitere Informationen dazu erhalten Sie in Anhang C.

Häufig werden Integer- mit reellen Zahlen gemischt oder eine Integer- als reelle Zahl interpretiert. Der Taschenrechner wird eine derartige Vermischung von Objekten feststellen und Sie fragen, ob Sie in den Näherungsmodus wechseln möchten.

Komplexe Zahlen sind erweiterte reelle Zahlen , welche die Einheit imaginäre Zahl , $i^2 = -1$ enthalten. So wird z. B. die komplexe Zahl *3 + 2i* als (3, 2) in den Taschenrechner eingegeben.

Komplexe Zahlen können Kartesisch oder im Polarmodus, abhängig von den Einstellungen des Taschenrechners, dargestellt werden. Beachten Sie dabei, dass komplexe Zahlen immer Kartesisch gespeichert werden, lediglich die Anzeige ist betroffen. Dadurch bleibt die Genauigkeit in den Berechnungen weitgehendst erhalten.

Die meisten mathematischen Funktionen arbeiten mit komplexen Zahlen. Daher ist es nicht erforderlich, dass Sie eine spezielle "komplex +" Funktion, zum Hinzufügen von komplexen Zahlen verwenden, denn Sie können die gleiche + Funktion auf reelle oder Integer-Zahlen anwenden.

Operationen mit Vektoren und Matrizen verwenden Objekte des Typs 3, reelle **Arrays** (Zahlenketten) , und falls benötigt Typ 4, **komplexe Arrays**. Objekte des Typs 2**, Strings** (Zeichenketten), sind einfache mit der alpha-numerischen Tastatur erzeugte (zwischen Anführungszeichen gesetzte) Textzeilen.

Eine **Liste** ist lediglich eine Ansammlung von Objekten zwischen zwei geschwungenen Klammern, im RPN-Modus durch Leerschritte (die Leertaste ist als SPC beschriftet) oder im algebraischen Modus durch Kommas getrennt. Listen, Objekte des Typs 5, sind besonders bei der Berechnung von Zahlensammlungen nützlich. So können z. B. die Spalten einer Tabelle als Listen eingegeben werden. Falls gewünscht, kann die Tabelle auch als Matrix oder Array eingegeben werden.

Objekte des Typs 8 sind *Programme in der User RPL* Sprache. Dies sind einfach Gruppen von Informationen, die zwischen den Symbolen << und >> eingegeben werden.

Programmen zugeordnet sind auch Objekte des Typs 6 und 7, **Globale** bzw. **Lokale Namen**. Diese Namen oder Variablen werden zur Speicherung aller Arten von Objekten benutzt. Das Konzept von globalen oder lokalen Namen hängt vom Gültigkeits- oder dem Einflussbereich einer Variablen in einem Programm ab.

Ein **algebraisches Objekt** , oder einfach **Algebraik** (Objekt des Typs 9), ist ein gültiger algebraischer Ausdruck zwischen zwei Apostrophen.

Ganze Dualzahlen , Objekte des Typs 10, werden in einigen Anwendungen der Informatik verwendet.

Grafische Objekte , Objekte des Typs 11, speichern die vom Taschenrechner erstellten Grafiken.

Gekennzeichnete Objekte, Objekte des Typs 12, werden als Ausgabe vieler Programme verwendet, um die Ergebnisse zu identifizieren. So bedeutet z. B. im gekennzeichneten Objekt Mean: 23,2, das Wort Mean: die Kennzeichnung zur Identifikation der Zahl 23,2 als Mittelwert einer Stichprobe.

Einheitenobjekte , Objekte des Typs 13, sind numerische Werte, an die eine physikalische Einheit angehängt ist.

Verzeichnisse , Objekte des Typs 15, sind Speicherstellen, welche bei der Organisation von Variablen, ähnlich den Verzeichnissen eines PC, behilflich sind.

Bibliotheken, Objekte des Typs 16, sind Programme im Speicher des Taschenrechners, auf welche innerhalb eines Verzeichnisses (oder Unterverzeichnisses) in Ihrem Taschenrechner zugegriffen werden kann. In deren Verwendung ähneln diese *built-in functions (integrierte Funktionen)*, Objekte des Typs 18, und *built-in commands (integrierte Befehlen)* den Objekten des Typs 19.

Ausdrücke im Display bearbeiten

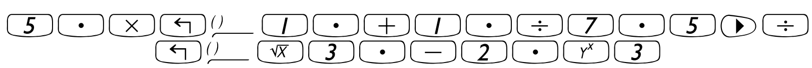
In diesem Abschnitt werden Beispiele von Ausdrücken gezeigt, welche direkt im Display des Taschenrechners (algebraische Veränderung oder RPN-Stack) bearbeitet und verändert werden können.

Erstellen von arithmetischen Ausdrücken

Für dieses Beispiel wählen wir den algebraischen Modus und ein *Fix* (festes) Format mit 3 Dezimalstellen als Anzeige im Display. Wir geben nun den nachfolgenden arithmetischen Ausdruck ein:

$$5.0 \cdot \frac{1.0 + \frac{1.0}{7.5}}{\sqrt{3.0} - 2.0^3}$$

Um diesen Ausdruck einzugeben verwenden Sie folgende Tastenfolge:



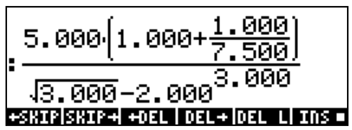
Der daraus resultierende Ausdruck sieht wie folgt aus: 5.*(1.+1./7.5)/(√3.-2.^3).

Drücken Sie die Taste ENTER, um den Ausdruck wie folgt anzuzeigen:

Beachten Sie dabei, dass, wenn Ihr CAS auf EXACT eingestellt ist (siehe Anhang C) und Sie Ihre Eingabe für Ganzzahlen über Integer-Werte machen, das Ergebnis eine symbolische Menge ist, z. B.

5 × ← () 1 + 1 ÷ 7 . 5 ▶ ÷
← () √x 3 − 2 y^x 3

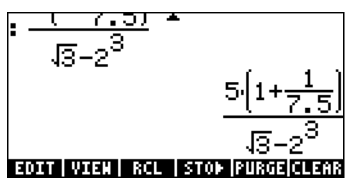
Bevor Sie ein Ergebnis erstellen, werden Sie darauf hingewiesen, auf den Approx mode (Näherungsmodus) umzustellen. Akzeptieren Sie die Änderung, um nachfolgendes Ergebnis zu erzielen (angezeigt im Fix decimal mode (festen Dezimalmodus) mit drei Nachkommastellen – siehe Kapitel 1):



Wenn der Ausdruck direkt in den Stack eingegeben wird, wird der Taschenrechner in diesem Fall versuchen, einen Wert für den Ausdruck zu berechnen, sobald Sie die Taste ENTER drücken,. Wird der Ausdruck aber in Anführungszeichen eingegeben, wird der Taschenrechner den Ausdruck, genauso ausgeben, wie Sie diesen eingegeben haben. Im nachfolgenden Beispiel, geben wir den gleichen Ausdruck, wie oben ein, aber verwenden dazu Anführungszeichen. In diesem Fall stellen wir auf den algebraischen Modus um, den CAS-Modus auf Exact (entfernen das Häkchen bei *_Approx*), und stellen den Anzeige-Modus auf *Textbook*. Die dafür erforderlichen Tastenanschläge sind wie folgt:

' 5 × ← () 1 + 1 ÷ 7 . 5 ▶ ÷
← () √x 3 − 2 y^x 3 ENTER

Das Ergebnis wird wie folgt angezeigt:



Um den Ausdruck auszuwerten können wir die Funktion EVAL, wie nachfolgend gezeigt, anwenden:

EVAL ← ANS ENTER

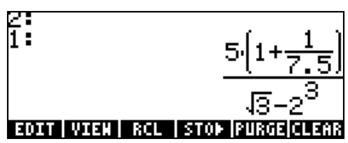
Wie im vorangegangenen Beispiel, werden Sie auch diesmal gefragt, ob Sie das CAS auf *Approx* umstellen möchten. Sobald Sie dies getan haben, erhalten Sie das gleiche Ergebnis wie zuvor.

Eine Alternative zu dem vorher eingegebenen Ausdruck in Anführungszeichen auszuwerten, ist die Anwendung der Option [→] →NUM . Um den Ausdruck aus dem bestehenden Stack wieder herzustellen, benutzen Sie folgende Tastenfolge: [◄][◄][→] →NUM

Wir geben nun den obigen Ausdruck ein, während der Taschenrechner auf den RPN-Modus eingestellt ist. Wir setzen das CAS auf *Exact* und die Anzeige auf *Textbook*. Die Tastenfolge zur Eingabe des Ausdrucks zwischen Anführungszeichen bleibt gleich, wie oben, d. h.

['][5][×][←] () [1][+][1][÷][7][.][5][▶][÷]
[←] () [√x][3][−][2][y^x][3][ENTER]

Dies ergibt die nachfolgende Ausgabe:



Drücken Sie die Taste [ENTER] ein weiteres Mal, um zwei Kopien zur Auswertung des Ausdrucks im Stack zu behalten. Als Erstes werten wir den Ausdruck mit der Funktion *EVAL* aus, und anschließend verwenden wir die Funktion *→NUM*. Werten Sie zuerst den Ausdruck mit der Funktion EVAL aus. Dieser Ausdruck ist semi-symbolisch und zwar enthält er sowohl Gleitkomma-Komponenten als auch eine √3 im Ergebnis. Als Nächstes, schalten wir die Stack Anordnung um und werten den Ausdruck mit der Funktion →NUM aus.

[▶]	Tauschen Sie die Stack Ebenen 1 und 2 (mit dem Befehl SWAP) aus
[→] →NUM	Werten Sie diesen mit der Funktion →NUM aus

Dieses Ergebnis ist nun rein numerisch, so dass die Ergebnisse im Stack, obwohl Sie den gleichen Ausdruck darstellen, unterschiedlich aussehen. Um aber zu überprüfen, ob diese das gleiche Ergebnis liefern, subtrahieren wir beide Werte und berechnen diese mit der Funktion EVAL:

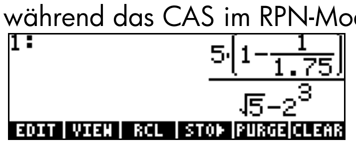
[−]	Subtrahieren Sie Ebene 1 von Ebene 2
[EVAL]	Berechnen Sie mithilfe der Funktion EVAL

Das Ergebnis ist Null (0).

Anmerkung: Vermeiden Sie es, Integer- mit reellen Zahlen zu vermischen, um Konflikte in der Berechnung zu vermeiden. Für viele Anwendungen in der Physik und Technik, einschließlich numerischer Lösung von Gleichungen, Statistikanwendungen usw. funktioniert der APPROX-Modus (siehe Anhang C) besser. Für mathematische Anwendungen, z. B. Infinitesimalrechnung, Vektorenanalyse, Algebra usw. wird der EXACT-Modus bevorzugt. Machen Sie sich mit beiden Methoden vertraut und lernen Sie, wie Sie für unterschiedliche Berechnungen (siehe Anhang C) aus einem in den anderen Modus umschalten können.

Bearbeiten von arithmetischen Ausdrücken

Angenommen, wir geben den folgenden Ausdruck zwischen Anführungszeichen ein, während das CAS im RPN-Modus auf EXACT steht:



und nicht wie im nachfolgenden Ausdruck: $5\cdot\frac{1+\frac{1}{7.5}}{\sqrt{3}-2^3}$. Es wurde ein inkorrekter Ausdruck durch folgende Eingabe angegeben:

['] [5] [×] [←] [()] [1] [+] [1] [÷] [1] [.] [7] [5] [▶] [÷] [←] [()] [√x] [5] [−] [2] [yˣ] [3] [ENTER]

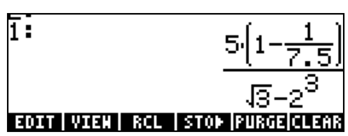
Um in den Zeileneditor zu gelangen drücken Sie [←] [▼]. Der Ausdruck sieht nun wie folgt aus:

Der Cursor zur Bearbeitung ist ein blinkender nach links gerichteter Pfeil, der sich über dem ersten Zeichen der zu bearbeitenden Zeile befindet. Da in diesem Fall die Bearbeitung im Löschen einiger Zeichen und Ersetzen dieser durch Andere besteht, werden wir die Pfeiltasten ◀ ▶ dazu benutzen, den Cursor auf dem zu verändernden Zeichen zu positionieren und anschließend die Löschtaste ⬅ betätigen, um diese Zeichen zu entfernen.

Die nachfolgenden Tastenanschläge vervollständigen die Bearbeitung in diesem Fall:

- Drücken Sie die rechte Pfeiltaste ▶, bis sich der Cursor rechts des Dezimalkommas in der Zahl 1,75 befindet
- Drücken Sie die Löschtaste ⬅ zweimal, um das Zeichen 1 zu löschen.
- Drücken Sie die rechte Pfeiltaste ▶ einmal, um den Cursor rechts der Zahl 7 zu positionieren
- Tippen Sie das Komma mit der Taste ⋅ ein
- Drücken Sie die rechte Pfeiltaste ▶, bis sich der Cursor rechts neben der Zahl √5 befindet
- Drücken Sie die Löschtaste ⬅ einmal, um das Zeichen 5 zu löschen
- Geben Sie mit der Taste 3 eine *3* ein
- Drücken Sie ENTER, um zum Stack zurückzukehren

Der bearbeitete Ausdruck befindet sich nun im Stack.



Die Bearbeitung einer Eingabezeile im algebraischen Modus erfolgt genau wie im RPN-Modus. Sie können dieses Beispiel im algebraischen Modus wiederholen, um die Aussage zu überprüfen.

Erstellen von algebraischen Ausdrücken

Algebraische Ausdrücke beinhalten nicht nur Zahlen, sondern auch Namen von Variablen. Als Beispiel geben wir nachfolgenden algebraischen Ausdruck ein:

$$\frac{2L\sqrt{1+\frac{x}{R}}}{R+y}+2\frac{L}{b}$$

Wir stellen den algebraischen Operationsmodus am Taschenrechner ein, CAS auf *Exact* und die Anzeige auf *Textbook*. Um diesen algebraischen Ausdruck einzugeben, verwenden wir folgende Tastenfolge:

' 2 × ALPHA L × √x ← () 1 + ALPHA ← X ÷ ALPHA R ▶ ÷
← () ALPHA R + ALPHA ← Y ▶ + 2 × ALPHA L ÷ ALPHA ← B

Drücken Sie ENTER, um das folgende Ergebnis zu erhalten:

Dieser Ausdruck kann ebenfalls im RPN-Modus eingegeben werden, was zum gleichen Ergebnis wie im algebraischen Modus führt.

Bearbeiten von algebraischen Ausdrücken

Algebraische Ausdrücke werden mit dem Zeileneditor ähnlich wie arithmetische Ausdrücke bearbeitet (siehe Beispiel oben). Angenommen, wir möchten den oben eingegebenen Ausdruck wie unten gezeigt verändern

$$\frac{2L\sqrt{1+\frac{x^2}{R}}}{R+x}+2\sqrt{\frac{L}{b}}$$

Um diesen algebraischen Ausdruck mit dem Zeileneditor zu bearbeiten benutzen wir ↰ ▼. Damit wird der Zeileneditor gestartet und der zu bearbeitende Ausdruck sieht wie folgt aus:

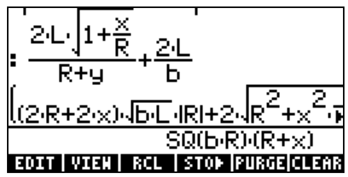
Der Cursor zur Bearbeitung ist ein blinkender nach links gerichteter Pfeil, der sich über dem ersten Zeichen der zu bearbeitenden Zeile befindet. Wie in einem früheren Beispiel zur Bearbeitung von Zeilen werden wir die Pfeiltasten ◀ ▶ zur richtigen Positionierung des Cursors benutzen und anschließend die Löschtaste ⌫, um entsprechende Zeichen zu löschen.

Die nachfolgenden Tastenanschläge vervollständigen die Bearbeitung in diesem Fall:

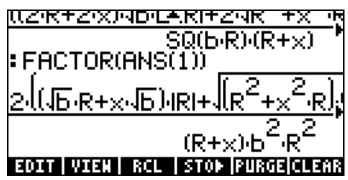
- Drücken Sie die rechte Pfeiltaste ▶, bis sich der Cursor rechts von x befindet
- Tippen Sie y^x 2, um die Potenz 2 für x einzugeben
- Drücken Sie die rechte Pfeiltaste ▶, bis sich der Cursor rechts von y befindet
- Drücken Sie die Löschtaste ⌫ einmal, um das Zeichen y zu löschen
- Drücken Sie ALPHA ↰ X um ein x einzugeben
- Drücken Sie die rechte Pfeiltaste ▶ viermal, um den Cursor rechts des Sternchens * zu positionieren
- Drücken Sie √x um das Symbol für Quadratwurzel einzugeben
- Drücken Sie ↰ () , um ein Klammernpaar einzugeben (gibt es immer paarweise)
- Drücken Sie die rechte Pfeiltaste ▶ einmal und anschließend die Löschtaste ⌫ ebenfalls einmal, um die rechte Klammer des obigen Klammerpaares zu löschen.
- Drücken Sie die rechte Pfeiltaste ▶ viermal, um den Cursor rechts von *b* zu positionieren
- Geben Sie ↰ () ein, um ein zweites Klammerpaar zu öffnen

- Drücken Sie die Löschtaste ⌫ einmal, um die linke Klammer des oben eingefügten Klammerpaares zu löschen.
- Drücken Sie die Taste ENTER, um zur Normalanzeige des Taschenrechners zurückzukehren.

Nachfolgend das Ergebnis:



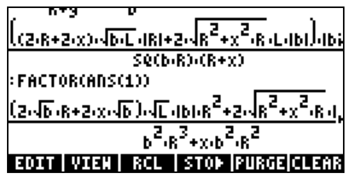
Beachten Sie, dass der Ausdruck um Faktoren wie |R|, den absoluten Wert und SQ(b·R), die Quadratwurzel von b·R, erweitert wurde einzugeben. Um zu sehen, ob wir dieses Ergebnis vereinfachen können, benutzen wir FACTOR(ANS(1)) im ALG-Modus:



- Drücken Sie ↰ ▼, um den Zeileneditor ein weiteres Mal zu starten. Die Lösung lautet nun:

- Drücken Sie ein weiteres Mal ENTER, um zur Normalanzeige zurückzukehren.

Um den gesamten Ausdruck im Display zu sehen, ändern wir die Option auf _*Small Stack Disp* in der *DISPLAY MODES*-Eingabemaske (siehe Kapitel 1). Nachdem Sie diese Änderung durchgeführt haben, sieht Ihre Anzeige wie folgt aus:

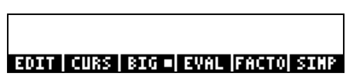


Anmerkung: Um griechische oder andere Buchstaben in algebraische Ausdrücke einzugeben, benutzen Sie das Menü CHARS. Dieses Menü wird mit der Tastenkombination → CHARS gestartet. Detailinformationen zu diesem Thema finden Sie in Anhang D.

Erstellen von Ausdrücken mithilfe des EquationWriters (EQW) (Gleichungsschreibers)

Der EquationWriter ist ein mächtiges Werkzeug, welches Ihnen nicht nur erlaubt, eine Gleichung einzugeben und anzusehen, sondern auch Änderungen vorzunehmen und zusätzliche Funktionen für beliebige einzelne Teile der Gleichung einzugeben und anzuwenden. Mit dem EquationWriter (EQW) können Sie komplexe mathematische Operation direkt oder im Einzelschrittmodus durchführen, genauso wie Sie es mit Papier und Bleistift z. B. bei der Lösung von Berechnungsaufgaben tun würden.

Der EquationWriter wird durch Drücken der Tastenkombination → → EQW gestartet (dritte Taste vierte Reihe von oben). Die Anzeige sieht wie folgt aus.



Die sechs Funktionstasten für den EquationWriter aktivieren die nachfolgenden Funktionen:

EDIT : ermöglicht es dem Anwender, eine Eingabe mit dem Zeileneditor bearbeiten (siehe obige Beispiele)

CURS : markiert einen Ausdruck und fügt diesem einen grafischen Cursor hinzu

BIG : falls ausgewählt (die Auswahl wird durch ein Zeichen in der Beschriftung angezeigt) wird die Schriftgröße 8 im Editor verwendet (die größte vorhandene Schrift)

EVAL : damit können Sie einen im EquationWriter hervorgehobenen Ausdruck, symbolisch oder numerisch auswerten (ähnlich wie ↱ EVAL)

FACTO : ermöglicht es Ihnen, einen im EquationWriter hervorgehobenen Ausdruck zu faktorieren (falls eine Faktorierung möglich ist)

SIMP : ermöglicht es Ihnen, einen Ausdruck in der EquationWriter Anzeige zu vereinfachen (sofern man diese gemäß den algebraischen Regeln des CAS vereinfachen kann)

Drücken Sie die Taste NXT, erscheinen zwei weitere Funktionsmenüs, wie unten aufgezeigt:

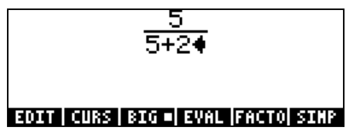
Die sechs Funktionstasten für den EquationWriter starten die nachfolgenden Funktionen:

CMDS : ermöglicht Ihnen den Zugriff auf CAS-Befehle, welche in alphabetischer Reihenfolge aufgelistet sind. Dies bewährt sich bei der Eingabe von CAS Befehlen in einen im EquationWriter vorhandenen Ausdruck.

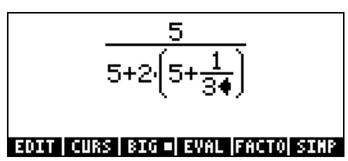
HELP : startet die CAS-Hilfefunktion des Taschenrechners, um Informationen und Beispiele zu CAS-Befehlen zur Verfügung zu stellen.
Einige Beispiele in der Anwendung des EquationWriters werden unten aufgezeigt.

Erstellen von arithmetischen Ausdrücken

Die Eingabe von arithmetischen Ausdrücken in den EquationWriter ist ähnlich wie die Eingabe dieser in Anführungszeichen in den Stack. Der Hauptunterschied besteht darin, dass die in den EquationWriter eingegebenen Ausdrücke im "textbook"-Stil (wie in einen Texteditor) anstelle einer Eingabe Zeile für Zeile geschrieben werden. Sobald Sie ein Teilungszeichen (d. h. [÷]) in den EquationWriter eingeben, wird ein Bruch erzeugt und der Cursor in den Zähler gesetzt. Benutzen Sie die Pfeiltaste nach unten, um zum Nenner zu gelangen. Versuchen Sie beispielsweise, folgende Tastenfolge in den EquationWriter einzugeben: [5][÷][5][+][2]
Das Ergebnis ist der Ausdruck

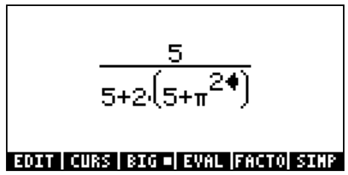


Der Cursor wird als ein nach links gerichteter Pfeil angezeigt. Der Cursor zeigt die aktuelle Position, in der eine Änderung vorgenommen werden kann, an. Die Eingabe eines Zeichens, Funktionsname oder Operation, wird an der Stelle erfolgen, an der sich der Cursor aktuell befindet. Geben Sie beispielsweise an der Cursorposition, wie im Bild dargestellt, Folgendes ein:
[×][↰]() [5][+][1][÷][3]
Der bearbeitete Ausdruck sieht wie folgt aus:

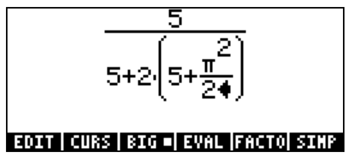


Angenommen, Sie möchten den Ausdruck des Nenners in der Klammer ändern, d. h. (5+1/3) durch (5+π^2/2) ersetzen. Als Erstes benutzen wir die Löschtaste ([◄]), löschen den Ausdruck 1/3 im Bruch und ersetzen diesen Teil wie folgt durch π^2/2: [◄][◄][◄][↰]π [y^x][2]

An dieser Stelle angekommen, sieht Ihre Anzeige wie folgt aus:



Um den Nenner 2 in den Ausdruck einzufügen, müssen wir den kompletten Ausdruck π^2 hervorheben (markieren). Dazu drücken wir die rechte Pfeiltaste (▶) einmal. An dieser Stelle fügen wir folgende Tastenfolge ein: ÷ 2
Der Ausdruck sieht nun wie folgt aus:



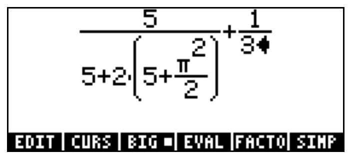
Angenommen, Sie möchten den Bruch 1/3 zu diesem Ausdruck hinzufügen, d. h. Sie möchten folgenden Ausdruck eingeben:

$$\frac{5}{5+2\cdot(5+\frac{\pi^2}{2})}+\frac{1}{3}$$

Als Erstes müssen wir den gesamten ersten Ausdruck entweder mit der rechten Pfeiltaste (▶) oder der Taste nach oben (▲) hervorheben, und diesen Vorgang so oft wiederholen, bis der gesamte Ausdruck markiert ist (in diesem Fall sieben Mal), wobei Ihre Anzeige dann wie folgt aussehen sollte:

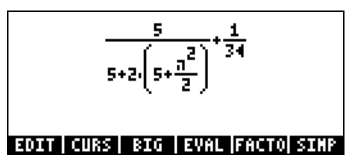
ANMERKUNG: Alternativ, von der Ursprungsposition des Cursors ausgehend (im Nenner rechts von der 2 im Ausdruck $\pi^2/2$), kann auch die Tastenkombination ↱ ▲, interpretiert als (↱ ⏶), verwendet werden.

Sobald der Ausdruck, wie oben gezeigt, hervorgehoben ist, tippen Sie folgende Tastenfolge ein, + 1 ÷ 3, um den Bruch 1/3 hinzuzufügen. Sie erhalten dann:



Verringern der Schriftgröße des Ausdrucks

Um den Ausdruck in einer kleineren Schrift darzustellen (was bei einem langen und weitverzweigten Ausdruck hilfreich sein kann), drücken Sie einfach die Funktionstaste BIG F3 . In diesem Fall sieht Ihre Anzeige wie folgt aus:

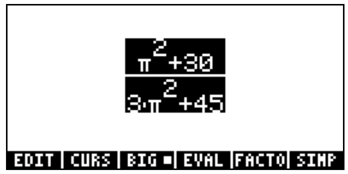


Um zur größeren Schriftart zurückzukehren, drücken Sie die Funktionstaste BIG F3 ein weiteres Mal.

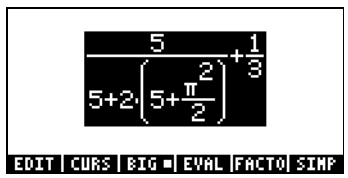
Auswerten/Berechnen des Ausdrucks

Um den Ausdruck (oder Teile des Ausdrucks) innerhalb des EquationWriters zu berechnen, markieren Sie den Teil, den Sie berechnen möchten und drücken die Funktionstaste EVAL F4 .

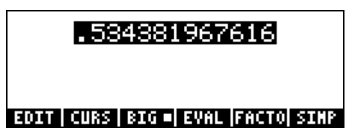
So z. B., um den gesamten Ausdruck in diesem Beispiel zu berechnen, markieren Sie zuerst den gesamten Ausdruck, indem Sie ↱ ⏶ drücken. Anschießend drücken Sie die Funktionstaste EVAL F4 . Befindet sich Ihr Taschenrechner im exakten CAS-Modus (d. h. der *_Approx* CAS Modus ist nicht ausgewählt), erhalten Sie das nachfolgende symbolische Ergebnis:



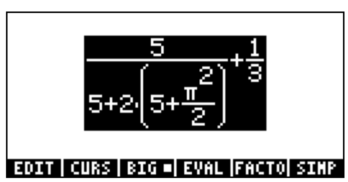
Möchten Sie den vorherigen, noch nicht berechneten Ausdruck, wiedergeben, benutzen Sie die Funktion UNDO, d. h. ↱ *UNDO* (die erste Taste in der dritten Reihe von oben). Der wiederhergestellte Ausdruck wird, wie vorhin markiert, angezeigt:



Wünschen Sie eine Gleitkommaberechnung (numerisch), benutzen Sie die Funktion →*NUM* (d. h. ↱ *→NUM*). Das Ergebnis sieht wie folgt aus:



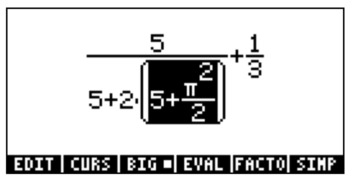
Benutzen Sie Funktion UNDO (↱ *UNDO*) ein weiteres Mal, um den ursprünglichen Ausdruck wieder herzustellen:



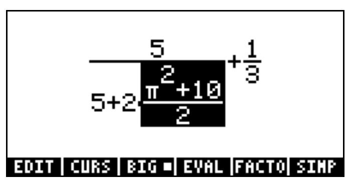
Berechnen eines Unterausdrucks

Angenommen, aus obigem Ausdruck möchten Sie lediglich den in Klammer stehenden Ausdruck im Nenner des ersten Bruches berechnen. Benutzen Sie dazu die Pfeiltasten, um diesen bestimmten Unterausdruck auszuwählen. Nachfolgend eine Möglichkeit, wie Sie dies tun können:

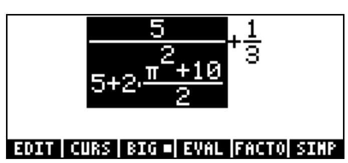
▼ Hebt nur den ersten Bruch hervor
▼ Hebt den Zähler des ersten Bruches hervor
► Hebt den Nenner des ersten Bruches hervor
▼ Hebt das erste Glied im Nenner des ersten Bruches hervor
► Hebt das zweite Glied im Nenner des ersten Bruches hervor
▼ Hebt den ersten Faktor im zweiten Glied im Nenner des ersten Bruches hervor
► Hebt den in Klammer stehenden Ausdruck im Nenner des ersten Bruches hervor



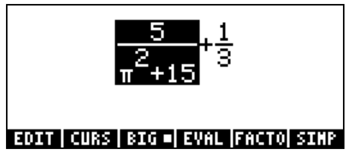
Da dies der Unterausdruck ist, den wir berechnen möchten, können wir an dieser Stelle die Funktionstaste EVAL F4 drücken, welche uns nachfolgendes Ergebnis anzeigt:



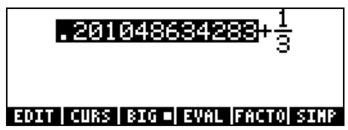
Eine weitere symbolische Berechnung: Angenommen, an dieser Stelle möchten wir lediglich den Bruch auf der linken Seite berechnen. Drücken Sie die Pfeiltaste (▲) dreimal, um diesen Bruch auszuwählen, was dann wie folgt aussieht:



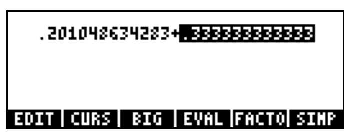
Anschießend drücken Sie die Funktionstaste EVAL F4, um den nachfolgenden Ausdruck zu erhalten:



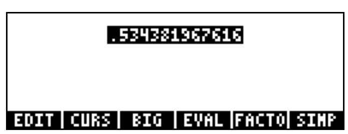
Versuchen wir es an dieser Stelle nun mit einer numerischen Berechnung dieses Gliedes. Verwenden Sie dazu →NUM, um nachfolgendes Ergebnis zu erhalten:



Heben wir nun den Bruch auf der rechten Seite hervor, um auch für dieses Glied eine numerische Berechnung zu erhalten, und lassen wir die Summe der beiden Dezimalwerte unter Verwendung von →NUM F3 in kleiner Schrift anzeigen, dann erhalten wir:

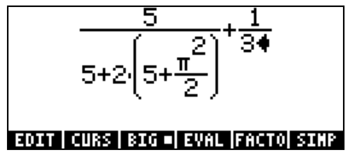


Um den Ausdruck hervorzuheben und im EquationWriter zu berechnen, benutzen wir F4, wodurch wir folgendes Ergebnis erzielen:

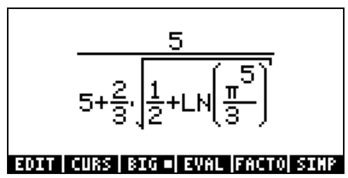


Bearbeiten von arithmetischen Ausdrücken

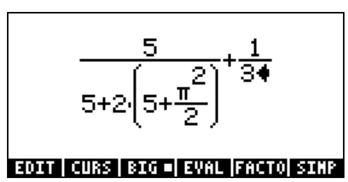
Nachfolgend zeigen wir einige Bearbeitungsmerkmale des EquationWriters als Beispiel. Wir beginnen, indem wir den im vorherigen Beispiel verwendeten Ausdruck eingeben:



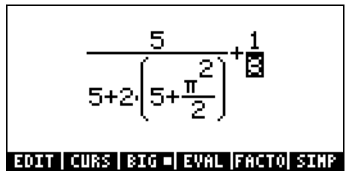
Dann verwenden wir die Bearbeitungsmerkmale des EquationWriters, um den Ausdruck wie folgt umzuwandeln:



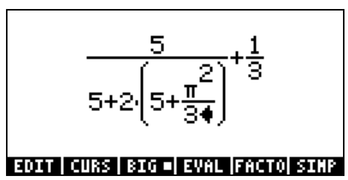
In vorangegangenen Übungen haben wir die Pfeiltasten zur Markierung von Unterausdrücken für Berechnungen verwendet. In diesem Fall benutzen wir sie, um einen bestimmten Bearbeitungscursor auszuwählen. Nachdem Sie die Eingabe des ursprünglichen Ausdrucks vervollständigt haben, wird der Eingabecursor (ein nach links gerichteter Pfeil) rechts von der 3 im Nenner des zweiten Bruches, wie unten gezeigt, stehen:



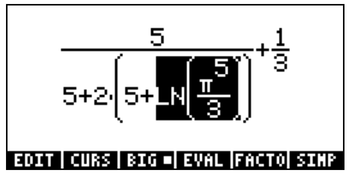
Drücken Sie die Pfeiltaste (▼), um zum reinen Bearbeitungscursor zu gelangen. Die Anzeige sieht nun wie folgt aus:



Mit der linken Pfeiltaste (◄) können Sie den Cursor im Allgemeinen nach links bewegen, dieser hält aber bei jeder einzelnen Komponente des Ausdrucks. Nehmen wir z. B. an, dass wir als erstes den Ausdruck $\pi^2/2$ in den Ausdruck *LN*($\pi^5/3$) umwandeln möchten. Mit der aktivierten reinen Cursortaste, wie oben angezeigt, drücken Sie die Pfeiltaste ◄ zweimal, um die 2 im Nenner von $\pi^2/2$ hervorzuheben. Drücken Sie als Nächstes die Löschtaste ⬅ einmal, um den Cursor in einen Einfügecursor umzuwandeln. Drücken Sie dann die Taste ⬅ ein zweites Mal, um die 2 zu löschen und dann 3, um die 3 einzugeben. An dieser Stelle sieht Ihre Anzeige wie folgt aus:

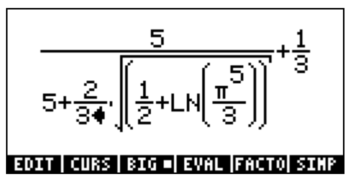


Als Nächstes drücken Sie die Pfeiltaste (▼), um mit dem reinen Bearbeitungscursor die 3 im Nenner von $\pi^2/3$ hervorzuheben. Drücken Sie die Pfeiltaste (◄)ein weiteres Mal, um den Exponenten im Ausdruck $\pi^2/3$ hervorzuheben. Drücken Sie als Nächstes die Löschtaste (⬅) einmal, um den Cursor in einen Einfügecursor umzuwandeln. Drücken Sie die Taste ⬅ ein zweites Mal, um die 2 zu löschen, um dann die 5 über die Taste 5 einzugeben. Drücken Sie die Pfeiltaste (▲) dreimal, um den Ausdruck $\pi^5/3$ hervorzuheben. Tippen Sie anschließend ↱ *LN*, um die Funktion *LN* auf diesen Ausdruck anzuwenden. Die Anzeige sieht nun wie folgt aus:

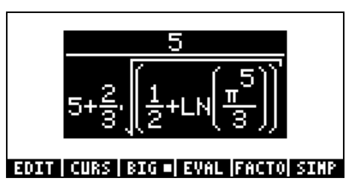


Als Nächstes werden wir die 5 innerhalb der Klammern in ½ ändern, indem wir nachfolgende Tastenanschläge benutzen:

Dann markieren wir den gesamten Ausdruck in der Klammer und fügen das Quadratwurzelzeichen wie folgt ein:
Als Nächstes konvertieren wir die 2 vor der Klammer des Nenners in 2/3 wie folgt:
An dieser Stelle sieht Ihre Anzeige wie folgt aus:



Der letzte Schritt besteht darin, 1/3 rechts vom Ausdruck zu entfernen. Dies wird wie folgt erreicht:
Die endgültige Version sieht nun so aus:



Zusammenfassung: Um einen Ausdruck im EquationWriter zu bearbeiten sollten Sie die Pfeiltasten ( ) benutzen. Um den Ausdruck, auf welchen Sie eine Funktion anwenden möchten, hervorzuheben (z. B. die Funktion LN und Quadratwurzel im obigen Beispiel), drücken Sie an einer beliebigen Stelle wiederholt die Pfeiltaste (▼), um zum reinen Bearbeitungscursor zu gelangen. In diesem Modus verwenden Sie dann die Pfeiltasten (◀ ▶), um im Ausdruck von einem Glied zum nächsten zu wechseln. An einem der Punkte, den Sie bearbeiten möchten, angekommen benutzen Sie die Löschtaste (⬅), um den Einfügecursor zu wählen und fahren mit der Bearbeitung des Ausdrucks fort.

Erstellen von algebraischen Ausdrücken

Ein algebraischer Ausdruck ähnelt einem arithmetischen, mit der Ausnahme, dass in dem algebraischen auch noch englische oder griechische Buchstaben eingefügt werden können. Der Vorgang beim Erstellen eines algebraischen Ausdrucks ist der gleiche wie beim Erstellen eines arithmetischen Ausdrucks, mit Ausnahme, dass eine alphabetische Tastatur zur Verfügung steht.

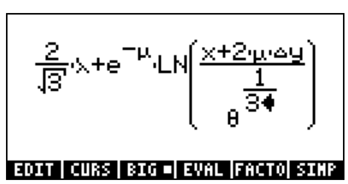
Nehmen wir nachfolgendes Beispiel, um die Anwendung des EquationWriters bei der Eingabe eines algebraischen Ausdrucks zu veranschaulichen. Angenommen, wir möchten folgenden Ausdruck eingeben.

$$\frac{2}{\sqrt{3}}\lambda + e^{-\mu} \cdot LN\left(\frac{x + 2\mu \cdot \Delta y}{\theta^{1/3}}\right)$$

Verwenden Sie dazu folgende Tastenfolge:

[2] [÷] [√x] [3] [▶] [▶] [×] [ALPHA] [↱] [N] [+] [↰] e^x [+/-] [ALPHA] [↱] [M]
[▶] [▶] [×] [↱] LN [ALPHA] [↰] [X] [+] [2] [×] [ALPHA] [↱] [M] [×] [ALPHA] [↱] [C]
[ALPHA] [↰] [Y] [▲] [▲] [▲] [÷] [ALPHA] [↱] [T] [y^x] [1] [÷] [3]

Dies ergibt die nachfolgende Ausgabe:



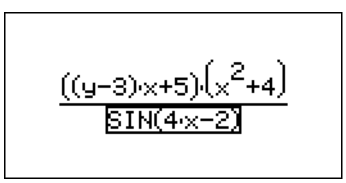
In diesem Beispiel haben wir mehrere englische Kleinbuchstaben verwendet, und zwar x (ALPHA ← X), mehrere griechische Buchstaben, und zwar λ (ALPHA → N), aber auch eine Kombination aus englischen und griechischen Buchstaben, Δy (ALPHA → C
ALPHA ← Y). Sie erinnern sich: Um einen englischen Kleinbuchstaben eingeben zu können, benötigen Sie die Kombination ALPHA ← gefolgt von dem Buchstaben, den Sie eingeben möchten. Sie können auch Sonderzeichen mithilfe des Menüs CHARS (→ CHARS) eingeben, wenn Sie sich nicht alle Tastenkombinationen für diese merken möchten Eine Auflistung häufig verwendeter ALPHA →-Tastenkombinationen wurde in einem vorangegangenen Abschnitt aufgeführt.

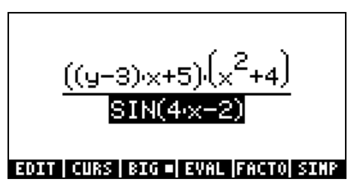
Der Ausdrucksbaum

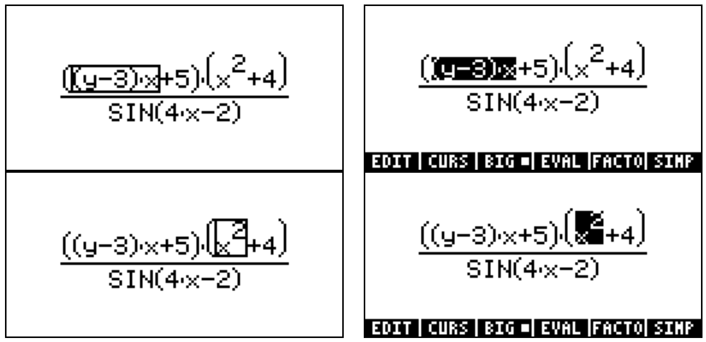
Der Ausdrucksbaum ist ein Diagramm, das anzeigt, auf welche Weise der EquationWriter einen Ausdruck darstellt (interpretiert). Ein ausführliches Beispiel finden Sie in Anhang E.

Die Funktion CURS

Die Funktion CURS (CURS) im EquationWriter (Taste F2) konvertiert die Anzeige in eine grafische Anzeige und erzeugt einen grafischen Cursor, der über die Pfeiltasten (◄►▲▼) gesteuert werden kann, um Unterausdrücke auszuwählen. Der mit CURS ausgewählte Unterausdruck wird in der grafischen Anzeige mit einem Rahmen dargestellt. Nachdem Sie einen Unterausdruck ausgewählt haben, können Sie die Taste ENTER drücken, um den ausgewählten Unterausdruck im EquationWriter hervorgehoben anzuzeigen. Durch Drücken der Taste ENTER zeigen die nachfolgenden Abbildungen unterschiedlich ausgewählte Unterausdrücke und die entsprechende EquationWriter-Anzeige an.





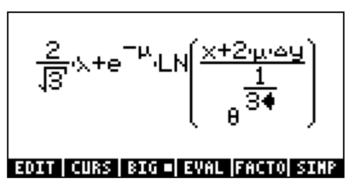


Bearbeiten von algebraischen Ausdrücken

Bei der Bearbeitung von algebraischen Ausdrücken gelten die gleichen Regeln wie bei der Bearbeitung von algebraischen Gleichungen:

- Benutzen Sie die Pfeiltasten (◄►▲▼), um den Ausdruck zu markieren
- Drücken Sie wiederholt den Pfeil (▼), um zum reinen Bearbeitungscursor zu gelangen. In diesem Modus verwenden Sie dann die Pfeiltasten (◄►), um sich von einem Glied zum nächsten im Ausdruck zu bewegen.
- Benutzen Sie die Löschtaste (◄) an einem Bearbeitungspunkt, um den Eingabecursor zu wählen und bearbeiten Sie den Ausdruck.

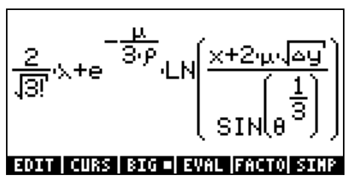
Um den reinen Bearbeitungscursor in Aktion zu sehen, beginnen wir mit dem algebraischen Ausdruck, welchen wir im obigen Beispiel eingefügt haben:



Drücken Sie den Pfeil ▼, da wo er sich aktuell befindet, um zum reinen Bearbeitungscursor zu gelangen. Die 3 im Exponenten θ wird hervorgehoben. Benutzen Sie die Pfeiltaste ◄, um sich von einem Element des Ausdrucks zum anderen zu bewegen. Die Auswahlreihenfolge des reinen Bearbeitungscursors in unserem Beispiel lautet wie folgt (drücken Sie wiederholt die linke Pfeiltaste ◄):

1. Die 1 im Exponenten von 1/3
2. θ
3. Δy
4. μ
5. 2
6. x
7. μ in der Exponentialfunktion
8. λ
9. 3 in der √3
10. die 2 im Bruch 2/√3

An dieser Stelle können wir den reinen Bearbeitungscursor in einen Einfügecursor ändern, indem wir die Löschtaste (◄) drücken. Benutzen wir nun diese beiden Cursor (den reinen Bearbeitungscursor und den Einfügecursor), um den aktuellen Ausdruck wie folgt zu ändern:



Haben Sie die Übung von vorhin gleich durchgeführt, sollten jetzt der folgende reine Bearbeitungscursor auf der Zahl 2 im ersten Faktor des Ausdrucks stehen. Führen Sie folgende Tastenanschläge durch, um den Ausdruck zu bearbeiten:

▶ ALPHA ↱ 2 Trägt die Faktorielle für die 3 in der Quadratwurzel ein

(das Eintragen der Faktoriellen ändert den cursor in einen Auswahlcursor)

▼▼▶▶ Wählt das μ in der Exponentialfunktion

÷ 3 × ALPHA ↱ F Ändert das Argument der Exponentialfunktion

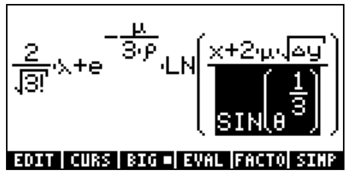
▶▶▶▶ Wählt *Δy* aus

√x Platziert ein Quadratwurzelsymbol über *Δy*

(auch dieser Vorgang ändert den Cursor in den auswahlcursor)

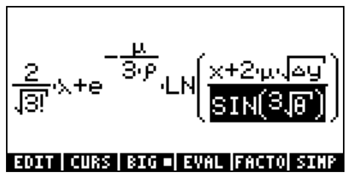
▼▼▶▲▲ SIN Wählen Sie $\theta^{1/3}$, und tragen Sie die Funktion SIN ein.

Die Anzeige sieht wie folgt aus :

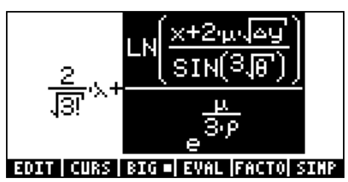


Berechnen eines Unterausdrucks

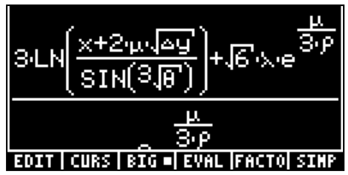
Da wir den Unterausdruck $\mathrm{SIN}(\theta^{1/3})$ bereits hervorgehoben haben, drücken wir nun die Funktionstaste EVAL F4 , um diesen Unterausdruck zu berechnen. Die Lösung lautet:



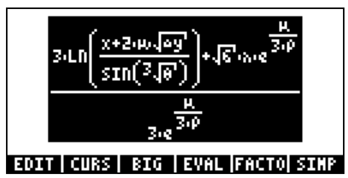
Einige algebraische Ausdrücke können nicht weiter vereinfacht werden. Versuchen Sie folgende Tastenanschläge: $\uparrow$ F4 . Sie werden feststellen, dass lediglich das vollständige Argument der Funktion *LN* hervorgehoben wird, sonst erfolgen keine Aktionen. Dies ist der Fall, weil der Ausdruck nach den Regeln des CAS nicht weiter berechnet (oder vereinfacht) werden kann. Versuchen Sie die Tastenfolge $\uparrow$ F4 erneut, gibt es weiterhin keine Änderungen im Ausdruck. Ein weiterer Versuch mit dieser Tastenfolge $\uparrow$ F4 hingegen, ändert den Ausdruck, wie folgt:



Betätigen Sie diese Tastenfolge $\uparrow$ F4 ein weiteres Mal, erscheinen weitere Änderungen:



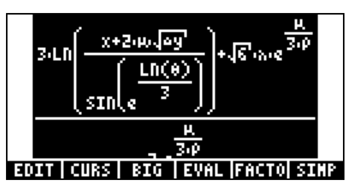
Dieser Ausdruck passt nicht mehr in den Anzeigebildschirm des EquationWriters. Der gesamte Ausdruck lässt sich aber in einer kleineren Schrift anzeigen. Drücken Sie die Funktionstaste BIG F3 , um das folgende Ergebnis zu erhalten:



Auch mit der größeren Schrift ist es möglich, sich durch den gesamten Ausdruck mit dem reinen Bearbeitungscursor zu bewegen. Versuchen Sie die Tastenfolge F3 ▼▼▼▼, um den reinen Bearbeitungscursor über den Faktor 3 im ersten Glied des Zählers zu bewegen. Drücken Sie die Pfeiltaste ▶, um sich durch den Ausdruck zu bewegen.

Einen Ausdruck vereinfachen

Drücken Sie die Funktionstaste BIG F3 , um erneut die Anzeige wie in der vorangegangenen Abbildung (siehe oben) zu erhalten. Drücken Sie nun die Funktionstaste SIMP F3 , um zu sehen, ob dieser Ausdruck, wie er sich im EquationWriter befindet, noch weiter vereinfacht werden kann. Das Ergebnis ist die nachfolgende Anzeige:



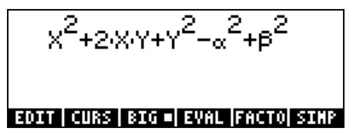
Die Anzeige zeigt das Argument der Funktion SIN, und zwar, $\sqrt[3]{\theta}$ umgewandelt in $e^{\frac{LN(\theta)}{3}}$. Dies mag zwar nicht wie eine Vereinfachung aussehen, es ist jedoch in so fern eine Vereinfachung, als dass die Quadratwurzelfunktion mit der inversen Funktion exp-LN ersetzt wurde.

Faktorieren eines Ausdrucks

In diesem Beispiel werden wir versuchen einen Polynomausdruck zu faktorieren. Um mit dem vorangegangenen Beispiel weiterzumachen, drücken Sie die Taste ENTER. Starten Sie dann den EquationWriter erneut durch Drücken der Tasten ↱ EQW. Geben Sie folgende Gleichung ein:

X Y^x 2 ▶ + 2 × X × ALPHA Y + ALPHA Y Y^x 2 ▶ − ALPHA ↱ A Y^x 2 ▶ ▶ + ALPHA ↱ B Y^x 2

Dies ergibt dann:

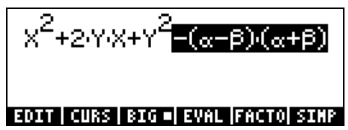


Wählen wir nun die ersten 3 Glieder des Ausdruckes aus, und versuchen wir diesen Unterausdruck zu faktorieren: ↱ ▲ ▼ ↱ ▶ ↱ ▶. Dies ergibt:

Drücken Sie die Funktionstaste FACTO, um Folgendes zu erhalten:

Um zum ursprünglichen Ausdruck zurückzukehren, drücken Sie ↱ UNDO . Als Nächstes geben Sie die Tastefolge ▼▼▼▶▶▶▶▶▶▶▲ ▲▲↱▶ ein, um die letzten beiden Glieder im Ausdruck zu markieren, d. h.,

Drücken Sie die Funktionstaste FACTO, um Nachfolgendes zu erhalten:



Um zum ursprünglichen Ausdruck zurückzukehren, drücken Sie ↱ UNDO . Markieren wir nun den gesamten Ausdruck, indem wir die Pfeiltaste (▲) einmal drücken. Drücken Sie die Funktionstaste FACTO, um Folgendes zu erhalten:

Um zum ursprünglichen Ausdruck zurückzukehren, drücken Sie ↱ UNDO .

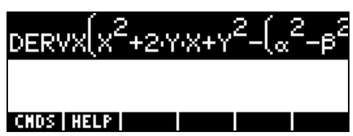
Anmerkung: Drücken Sie die Funktionstaste EVAL oder SIMP, sobald der gesamte Ausdruck hervorgehoben ist, erhalten Sie die folgende Vereinfachung des Ausdruckes:

Verwenden der Menütaste CMDS

Verwenden wir den Polynomenausdruck aus vorangegangenem Beispiel und drücken die Taste NXT, um die Funktionstasten der Menüs CMDS und HELP anzuzeigen. Diese beiden Befehle gehören zum zweiten Teil des Softwaremenüs im EquationWriter. Versuchen wir nun ein Beispiel, in dem die Funktionstaste CMDS zur Anwendung kommt: Drücken Sie die Funktionstaste CMDS, um eine Auflistung der CAS-Befehle zu erhalten:

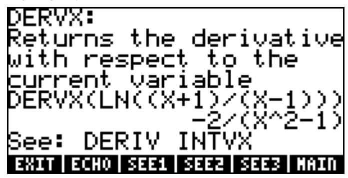
Als Nächstes wählen Sie wie folgt den Befehl DERVX (die Ableitungsfunktion in Bezug auf die Variable X, die aktuelle unabhängige Variable im CAS): ALPHA D ▼ ▼ ▼ . Nun wird der Befehl DERVX ausgewählt:

Drücken Sie die Funktionstaste OK (F6), um Folgendes zu erhalten:



Anschließend drücken Sie die Taste NXT, um zum ursprünglichen EquationWriter Menü zurückzukehren und dann die Funktionstaste EVAL (F4), um diese Ableitungsfunktion zu berechnen. Die Lösung lautet:

Verwenden des Menüs HELP (Hilfe)
Drücken Sie die Taste NXT, um die Funktionstasten CMDS und HELP anzuzeigen. Drücken Sie die Funktionstaste HELP, um eine Auflistung der CAS-Befehle zu erhalten: Drücken Sie dann ALPHA D ▼ ▼ ▼, um den Befehl DERVX auszuwählen. Drücken Sie die Funktionstaste OK (F6), um Informationen zum Befehl DERVX zu erhalten:



Eine genaue Erklärung zur Verwendung der Hilfefunktion für das CAS finden Sie in Kapitel 1. Um zum EquationWriter zurückzukehren, drücken Sie die Funktionstaste EXIT. Drücken Sie ENTER, um den EquationWriter zu verlassen.

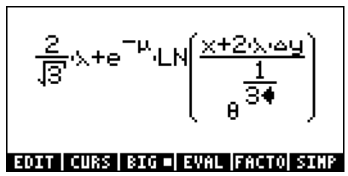
Verwenden der Bearbeitungs-Funktionen BEGIN, END, COPY, CUT und PASTE
Um die Bearbeitung zu vereinfachen, egal ob Sie sich im EquationWriter oder im Stack befinden, werden im Taschenrechner fünf Bearbeitungsfunktionen zur Verfügung gestellt BEGIN, END, COPY, CUT und PASTE, die in Kombination mit der rechten Shift-Taste (↱) und den entsprechenden Tasten (2,1), (2,2), (3,1), (3,2) und (3,3) gestartet werden. Dies sind die ganz links angeordneten Tasten der Reihen 2 und 3. Die Aktionen dieser Bearbeitungsfunktionen sind wie folgt:

BEGIN: markiert den Anfang einer zu bearbeitenden Zeichenkette
END: markiert das Ende einer zu bearbeitenden Zeichenkette
COPY: kopiert die Zeichenkette, die mit BEGIN und END ausgewählt wurde
CUT: schneidet die Zeichenkette, die mit BEGIN und END ausgewählt wurde, aus
PASTE: fügt die zuvor kopierte oder ausgeschnittene Zeichenkette an der aktuellen Cursorposition ein

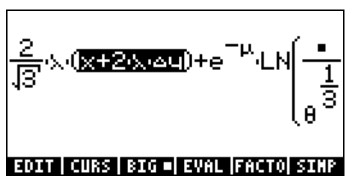
Um ein Beispiel zu sehen, starten wir den EquationWriter und geben den nachfolgenden Ausdruck ein (wurde in einem vorangegangenen Beispiel benutzt):

[2] [÷] [√x] [3] [▶] [▶] [×] [ALPHA] [→] [M] [+] [←] [eˣ] [+/-] [ALPHA] [→] [M]
[▶] [▶] [×] [→] [LN] [ALPHA] [←] [X] [+] [2] [×] [ALPHA] [→] [M] [×] [ALPHA] [→] [C]
[ALPHA] [←] [Y] [▲] [▲] [▲] [÷] [ALPHA] [→] [T] [yˣ] [1] [÷] [3]

Der ursprüngliche Ausdruck sieht wie folgt aus.



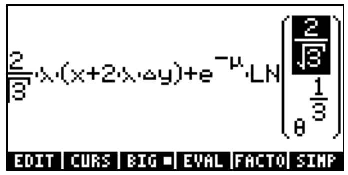
Wir möchten nun den Unterausdruck $x+2\cdot\lambda\cdot\Delta y$ aus dem Argument der Funktion *LN* entfernen und nach rechts von λ im ersten Glied verschieben. Eine Möglichkeit dies zu tun, lautet: [▼] [◀] [◀] [◀] [▲] [▲] [▲] [→] [CUT] [◀] [◀] [▲] [×] [→] [PASTE]
Der veränderte Ausdruck sieht wie folgt aus:



Anschließend kopieren wir den Bruch $2/\sqrt{3}$ aus dem Ausdruck ganz links und fügen ihn in den Zähler des Arguments der Funktion LN ein. Versuchen Sie folgende Tastenanschläge:

[▼] [▼] [◀] [◀] [▲] [▲] [▲] [→] [COPY] [▼] [▼]
[→] [▶] [◀] [◀] [◀] [→] [PASTE]

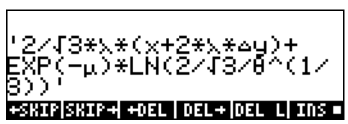
Die Anzeige sieht wie folgt aus:



Wenn Sie im EquationWriter arbeiten, benötigen Sie die Funktionen BEGIN und END nicht, da Zeichenketten mit den Pfeiltasten ausgewählt werden können. Die Funktionen BEGIN und END werden hauptsächlich dann benötigt, wenn wir uns im Zeileneditor befinden. Wählen wir z. B. den Ausdruck $x+2\cdot\lambda\cdot\Delta y$ in diesem Ausdruck aus, jedoch unter Verwendung des Zeileneditors innerhalb des EquationWriters, wird werden folgende Tastaturanschläge benötigt:

↱ ▲ F1

Die Anzeige des Zeileneditors sieht wie folgt aus (Die Anführungszeichen werden gezeigt nur wenn Taschenrechner im RPN-Modus) :



Um den gewünschten Unterausdruck auszuwählen, verwenden Sie:

► ► ► ► ► ► ► ► ↱ BEGIN
► ► ► ► ► ► ► ► ► ► ► ↱ END

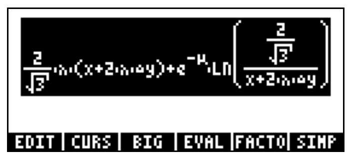
In der Anzeige unten ist der gewünschte Unterausdruck hervorgehoben:

Wir können nun diesen Ausdruck kopieren und wie folgt in den Nenner des Argumentes der Funktion LN einfügen:

↱ COPY ► ► … (27 Mal) … ►
◄ ◄ … (9 Mal) … ◄ ↱ PASTE

Die Anzeige des Zeileneditors sieht nun wie folgt aus:

Drücken Sie [ENTER], erhalten Sie den Ausdruck im EquationWriter (in kleiner Schrift drücken Sie die Funktionstaste BIG [F3]):



Drücken Sie [ENTER], um den EquationWriter zu verlassen.

Erstellen und bearbeiten von Summen, Ableitungsfunktionen und Integralen

Summen, Ableitungsfunktionen und Integrale werden im Allgemeinen bei Infinitesimalrechnungs-, Wahrscheinlichkeits- und Statistik-Anwendungen eingesetzt. In diesem Abschnitt zeigen wir einige Beispiele solcher Operationen, erstellt mit dem EquationWriter.

Summen

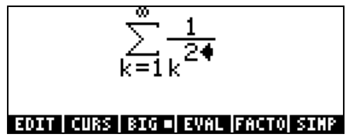
Wir benutzen den EquationWriter, um folgende Summe einzugeben:

$$\sum_{k=1}^{\infty} \frac{1}{k^2}$$

Drücken Sie [→] EQW , um den EquationWriter zu starten. Anschließend drücken Sie dann [→] Σ , um das Summenzeichen einzugeben. Beachten Sie, dass die Anzeige des EquationWriters Eingabefelder für den Index der Summe und für die zu summierende Menge zur Verfügung stellt. Um diese Eingabefelder auszufüllen, benutzen Sie die Tastefolge:

[ALPHA][←][K][▶][1][▶][←] ∞ [▶][1][÷][ALPHA][←][K][y^x][2]

Die daraus resultierende Anzeige sieht wie folgt aus:



Um den entsprechenden Ausdruck im Zeileneditor anzuzeigen, drücken Sie ↱ ▲ und die Funktionstaste F1 , um folgende Ansicht zu erhalten:

Dieser Ausdruck zeigt eine allgemeine Form einer Summe, die direkt in den Stack oder Zeileneditor eingegeben wurde:
Σ(*index = starting_value*, *ending_value*, *summation expression*)
Drücken Sie ENTER, um zum EquationWriter zurückzukehren. Die so entstandene Anzeige ist nicht die Summe, die wir eingegeben haben, sondern deren symbolischer Wert, und zwar:

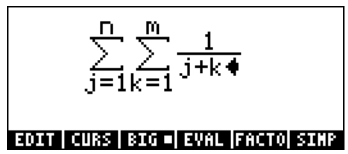
Um die Summe wieder herzustellen, verwenden Sie ↱ UNDO . Um die Summe neu zu berechnen, verwenden Sie die Funktionstaste F4 . Diese zeigt erneut, dass gilt:

$$\sum_{k=1}^{\infty} \frac{1}{k^2} = \frac{\pi^2}{6} .$$

Sie können den EquationWriter benutzen, um zu beweisen, dass gilt:

$$\sum_{k=1}^{\infty} \frac{1}{k} = +\infty .$$

Diese Summe (eine unendliche Reihe darstellend) wird als divergent bezeichnet.
Auch doppelte Summen sind möglich, so z. B.:



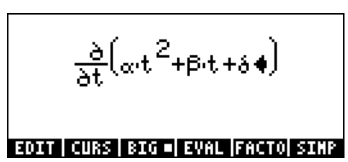
Ableitungsfunktionen
Wir benutzen den EquationWriter, um folgende Ableitungsfunktion einzugeben:

$$\frac{d}{dt}(\alpha \cdot t^2 + \beta \cdot t + \delta)$$

Drücken Sie ↱ EQW , um den EquationWriter zu starten. Drücken Sie anschließend ↱ ∂ , um zu dem (partiellen) Ableitungsfunktionszeichen zu gelangen. Beachten Sie, dass das Zeichen bei der Eingabe in den EquationWriter eine Eingabemöglichkeit für den abzuleitenden Ausdruck und die Ableitungsvariable zur Verfügung stellt. Um diese Eingabefelder auszufüllen, benutzen Sie die Tastefolge:

ALPHA ↰ T ▶ ALPHA ↱ A × ALPHA ↰ T y^x 2
▶ ▶ + ALPHA ↱ B × ALPHA ↰ T + ALPHA ↱ D

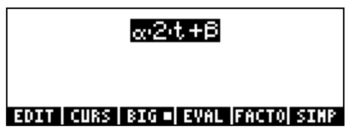
Die Anzeige sieht wie folgt aus.



Um den entsprechenden Ausdruck im Zeileneditor anzuzeigen, drücken Sie ↱ ▲ und die Funktionstaste F1 , um folgende Ansicht zu erhalten:

Daraus ersehen wir, dass der allgemeine Ausdruck einer Ableitungsfunktion im Zeileneditor oder Stack wie folgt lautet:
∂*Variable*(*Funktion von Variablen*)

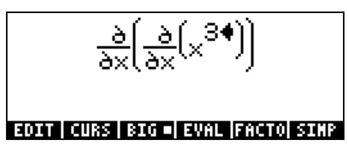
Drücken Sie ENTER, um zum EquationWriter zurückzukehren. Die so entstandene Anzeige ist nicht die Ableitungsfunktion, die wir eingegeben haben, sondern deren symbolischer Wert, und zwar:



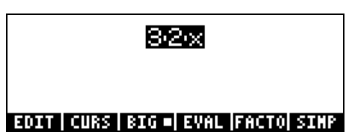
Um die Ableitungsfunktion wiederherzustellen, verwenden Sie ↱ UNDO . Um die Ableitungsfunktion neu zu berechnen, verwenden Sie die Funktionstaste F4 . Diese zeigt erneut, dass gilt:

$$\frac{d}{dt}(\alpha \cdot t^2 - \beta \cdot t + \delta) = 2\alpha \cdot t + \beta \ .$$

Auch eine Ableitung aus einer Ableitung ist möglich, so z. B.:



welche ausgewertet

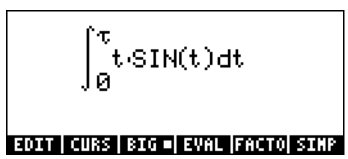


ergibt.

Anmerkung: Für eine Teilableitung ist die Notation $\frac{\partial}{\partial x}(\ )$ einwandfrei. Die richtige Notation für eine Gesamtableitung (d. h. eine Ableitung einer Variablen) lautet $\frac{d}{dx}(\ )$. Der Taschenrechner macht jedoch keinen Unterschied zwischen Teil- und Gesamtableitungen.

Bestimmte Integrale

Wir benutzen den EquationWriter, um folgende bestimmte Integrale $\int_0^{\tau} t \cdot \sin(t) \cdot dt$ einzugeben. Drücken Sie [↱] EQW , um den EquationWriter zu starten. Drücken Sie anschließend [↱] ∫ , um zu dem Integralzeichen zu gelangen. Beachten Sie, dass das Zeichen, wenn in den EquationWriter eingegeben, eine Eingabemöglichkeit für die Grenzwerte der Integrale, den Integranden und die Integrationsvariable zur Verfügung stellt. Um diese Eingabefelder auszufüllen, benutzen Sie die Tastefolge:
[0] [▶] [ALPHA] [↱] [U] [▶] [ALPHA] [↰]
[T] [×] [SIN] [ALPHA] [↰] [T] [▶] [ALPHA] [↰] [T] . Die Anzeige sieht wie folgt aus:



Um den entsprechenden Ausdruck im Zeileneditor anzuzeigen, drücken Sie [▲][▲] und die Funktionstaste [F1] , um folgende Ansicht zu erhalten:

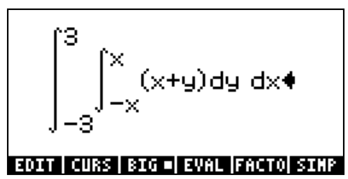
Daraus ersehen wir, dass der allgemeine Ausdruck einer Ableitungsfunktion im Zeileneditor oder Stack wie folgt lautet: ∫(*lower_limit, upper_limit,integrand,variable_of_integration*)

Drücken Sie ENTER, um zum EquationWriter zurückzukehren. Die entstandene Anzeige ist nicht die bestimmte Integrale, die wir eingegeben haben, sondern deren symbolischer Wert, und zwar:

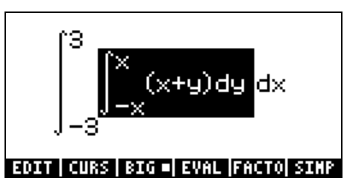
Um die Ableitungsfunktion wieder herzustellen, verwenden Sie ↱ UNDO . Um die Ableitungsfunktion neu zu berechnen, verwenden Sie die Funktionstaste F4 . Diese zeigt erneut, dass

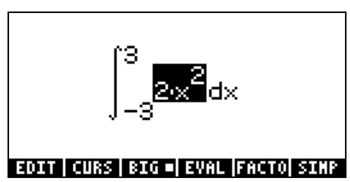
$$\int_0^{\tau} t \cdot \sin(t) \cdot dt = \sin(\tau) - \tau \cdot \cos(\tau)$$

Auch Doppelintegrale sind möglich. So zum Beispiel:



welche ausgewertet 36 ergibt. Eine Teilauswertung ist auch möglich, beispielsweise:

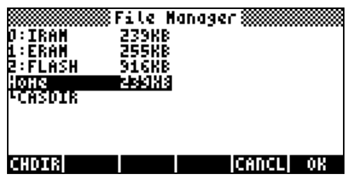




Der Wert dieser Integrale beträgt 36.

Organisieren der Daten im Taschenrechner

Sie können Daten in Ihrem Taschenrechner organisieren, indem Sie Variablen in einer Verzeichnisstruktur ablegen (speichern). Um den Speicher des Taschenrechners zu verstehen, sehen wir uns zunächst einmal das Datenverzeichnis an. Drücken Sie die Tastenkombination ← FILES (erste Taste in der zweiten Reihe von oben), um zu dem Datenmanager des Taschenrechners zu gelangen:



Diese Ansicht ist eine Momentaufnahme des Taschenrechnerspeichers und der Verzeichnisstruktur. In der Anzeige sehen wir, dass der Taschenrechner drei Speicherschnittstellen (oder Speicherpartitionen) hat, Schnittstelle *0:IRAM*, Schnittstelle *1:ERAM*, und Schnittstelle *2:FLASH*. Speicherschnittsellen werden dazu benutzt, Anwendungen von Drittanbietern oder Bibliotheken, aber auch Backups zu speichern. Auch die Größe der drei Schnittstellen wird angegeben. In der vierten und den darauf folgenden Zeilen der Anzeige wird die Verzeichnisstruktur des Taschenrechners angezeigt. Das oberste Verzeichnis (im Moment hervorgehoben) ist das *Home*-Verzeichnis und enthält ein vordefiniertes Unterverzeichnis *CASDIR*. Die File Manager (Datei-Manager) Ansicht verfügt über drei Funktionen, die Funktionstasten zugeordnet sind:

CHDIR (F1): Wechselt in das ausgewählte Verzeichnis
CANCL (F5): Aktion abbrechen
OK (F6): Auswahl bestätigen

Um z. B. ins Verzeichnis CASDIR zu wechseln, drücken Sie die Pfeiltaste ▼ und anschließend CHDIR (F1). Dadurch wird das *File Manager*-Fenster geschlossen, und wir erhalten die Normalanzeige des Taschenrechners. In der zweiten Zeile von oben werden Sie in der Anzeige { HOME CASDIR } sehen, was anzeigt, dass Sie sich im CASDIR innerhalb des HOME-Verzeichnisses befinden.

Funktionen zur Manipulation von Variablen

In dieser Anzeige stehen insgesamt 20 Befehle zur Verfügung, welche Funktionstasten zugeordnet sind und der Erstellung, Bearbeitung und Manipulation von Variablen dienen. Die ersten sechs Funktionen sind wie folgt:

EDIT zum Bearbeiten einer hervorgehobenen Variablen
COPY zum Kopieren einer hervorgehobenen Variablen
MOVE zum Verschieben einer hervorgehobenen Variablen
RCL zum Wiederherstellen des Inhalts einer hervorgehobenen Variablen
EVAL zum Berechnen einer hervorgehobenen Variablen
TREE zum Anzeigen der Verzeichnisstruktur in der sich die Variable befindet

Wenn Sie die Taste NXT drücken, erhalten Sie das nächste zur Verfügung stehende Funktionsset:

PURGE zum Löschen oder Bereinigen einer Variablen
RENAM zum Umbenennen einer Variablen
NEW zum Erstellen einer neuen Variablen
ORDER zum Anordnen mehrerer Variablen in einem Verzeichnis
SEND zum Senden einer Variablen an einen anderen Taschenrechner oder Computer
RECV zum Empfangen einer Variablen von einem anderen Taschenrechner oder Computer

Wenn Sie die Taste NXT drücken, erhalten Sie das dritte zur Verfügung stehende Funktionsset:

HALT um vorrübergehend zum Stack zurückzukehren
VIEW um den Inhalt einer Variablen anzuzeigen
EDITB um den Inhalt einer binären Variablen zu bearbeiten (ähnlich wie EDIT)
HEADE um das Verzeichnis, in dem sich die Variable aus der Kopfzeile befindet, anzuzeigen
LIST stellt eine Liste von Variablennamen und Beschreibungen bereit
SORT um Variablen nach einem bestimmten Sortierkriterium zu anzuordnen

Wenn Sie die Taste NXT drücken, erhalten Sie das letzte zur Verfügung stehende Funktionsset:

XSEND   um eine Variable mit dem X-Modem-Protokoll zu senden
CHDIR   um das Verzeichnis zu wechseln

Um zwischen den verschiedenen Funktionsmenü-Befehlen zu navigieren, können Sie nicht nur die Taste NEXT (NXT), sondern auch die Taste PREV (← PREV) verwenden.

Der Benutzer soll nun diese Funktionen selbst ausprobieren. Deren Anwendung ist ziemlich einfach.

Das HOME-Verzeichnis

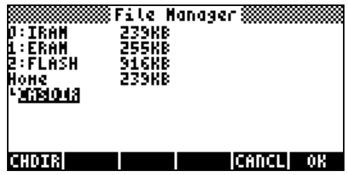
Wie bereits erwähnt, ist das HOME-Verzeichnis das Basis-Verzeichnis des Taschenrechners. Um zu dem HOME-Verzeichnis zu gelangen, können Sie die Funktion UPDIR (← UPDIR) benutzen – wiederholen Sie diesen Vorgang solange bis der Ausdruck { HOME } in der zweiten Zeile Ihres Displays erscheint. Alternativ dazu können Sie auch ← (halten) UPDIR verwenden, drücken Sie ENTER im algebraischen Modus. In diesem Beispiel enthält das HOME-Verzeichnis nichts weiter als das CASDIR (CAS-Verzeichnis). Drücken Sie die Taste VAR, werden die Variablen der Funktionstasten angezeigt:

Unterverzeichnisse

Um Ihre Daten in einer gut organisierten Verzeichnisstruktur zu speichern, können Sie Unterverzeichnisse im HOME-Verzeichnis anlegen und weitere Unterverzeichnisse in diesen, in einer Hierarchie, ähnlich wie diese in Form von Ordnern auf modernen Computern zu finden ist. Die Unterverzeichnisse können so benannt werden, dass der Name bereits über den Inhalt des Unterverzeichnisses Auskunft gibt, Sie können den Namen jedoch frei wählen.

Das Unterverzeichnis CASDIR

Das Unterverzeichnis CASDIR enthält einige für die ordnungsgemäße Funktion des Taschenrechners erforderlichen Variablen des CAS (Computer Algebraisches System, siehe Anhang C). Um den Inhalt des Verzeichnisses anzuzeigen, können wir die Tastenkombination ← FILES benutzen, die den *File Manager* (Dateimanager) erneut öffnet:



Diesmal ist CASDIR in der Anzeige hervorgehoben. Um den Inhalt des Verzeichnisses anzuzeigen, drücken Sie die Funktionstaste OK (F6) oder ENTER, um die folgende Ansicht zu erhalten:

In der Anzeige ist eine Tabelle, die die Variablen im CASDIR beschreibt. Dies sind im Speicher des Taschenrechners vordefinierte Variablen, welche bestimmte Parameter für die CAS-Operation festlegen (siehe Anhang C). Die obige Tabelle enthält vier Spalten:

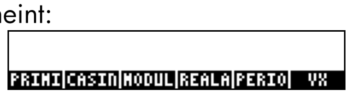
- In der ersten Spalte ist der Typ der Variablen angezeigt (z. B. 'EQ' bedeutet eine Variable des Typs Gleichung, |R zeigt an, dass es sich um eine Variable mit einem reellem Wert handelt, { } bedeutet eine Liste, *nam* bedeutet 'ein globaler Name' und das Symbol stellt eine grafische Variable dar.
- Die zweite Spalte stellt den Namen der Variablen, d. h. *PRIMIT, CASINFO, MODULO, REALASSUME, PERIOD, VX* und *EPS* dar.
- Spalte 3 enthält eine weitere Spezifikation für den Typ der Variablen, z. B. bedeutet *ALG* algebraischer Ausdruck, *GROB* steht für grafisches Objekt, *INTG* bedeutet eine numerische Integer-Variable, *LIST* stellt eine Liste von Daten dar, *GNAME* bedeutet globaler Name und *REAL* ist eine numerische reelle (Gleitkommazahl-) Variable.
- Die vierte und letzte Spalte ist die Größe in Byte der abgeschnittenen Variablen ohne Dezimalstellen (d. h. Halbbyte). So benötigt z. B. die Variable PERIOD 12,5 Byte, während die Variable REALASSUME 27,5

Byte benötigt (1 Byte = 8 Bit, 1 Bit ist die kleinste Einheit im Speicher von Computern und Taschenrechnern).

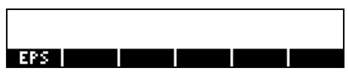
CASDIR-Variablen im Stack

Drücken Sie die Taste ON, wird die vorangegangene Anzeige geschlossen und Sie erhalten die Normalanzeige des Taschenrechners. Standardmäßig kommen wir zum TOOL-Menü zurück:

Wir können die Variablen im aktuellen Verzeichnis CASDIR ansehen, indem wir die Taste VAR drücken (erste Taste in der zweiten Reihe von oben). Folgende Anzeige erscheint:



Drücken Sie die Taste NXT, sehen Sie eine weitere, in diesem Verzeichnis gespeicherte Variable:



- Um z. B. den Inhalt der Variablen EPS anzuzeigen, drücken Sie ↱ EPS. Diese zeigt den Wert von EPS als .0000000001 an.
- Um den Wert einer numerischen Variablen anzuzeigen, müssen wir nur die entsprechende Funktionstaste für die Variable drücken. So zeigt z. B. Drücken von EPS gefolgt von ENTER den gleichen Wert der Variablen im Stack, wenn der Taschenrechner auf *Algebraic* gesetzt ist. Befindet sich der Taschenrechner im *RPN*-Modus, müssen Sie nur die Funktionstaste für ENTER drücken.
- Um den vollen Namen einer Variablen anzuzeigen, drücken Sie erst das Apostroph ' und dann die der Variablen entsprechende Funktionstaste. So z. B. für die im Stack aufgelistete Variable PERIO verwenden wir ' PERIO, was den String 'PERIOD' ausgibt. Diese Prozedur gilt für beide Operationsmodi des Taschenrechners, den *algebraischen* wie auch den *RPN*-Modus.

Variablen in CASDIR

Die in CASDIR enthaltenen Standardvariablen sind:

PRIMIT	Letzte berechnete Stammfunktion, keine Standardvariable, sondern eine die wir in einem vorangegangenen Beispiel erstellt haben
CASINFO	ein Graph das CAS-Informationen liefert
MODULO	Modulo für modulare Arithmetik (Standard = 13)
REALASSUME	Auflistung von Variablennamen, von welchen angenommen wird, dass sie reelle Werte darstellen
PERIOD	Intervall für trigonometrische Funktionen (Standard = 2π)
VX	Name der unabhängigen Standardvariablen (Standard = X)
EPS	Wert des kleinen Inkrementes (Epsilon), (Standard = 10^{-10})

Diese Variablen werden für die Funktion des CAS benutzt.

Verzeichnis- und Variablen-Namen tippen

Um Namen für Variablen einzugeben, müssen Sie eine Zeichenfolge auf einmal eingeben, welche entweder nur aus Buchstaben oder aus einer Kombination von Buchstaben und Zahlen bestehen kann. Wenn Sie nicht die Tasten ALPHA, oder Tastenkombination ALPHA ← oder ALPHA → beim Eingeben jedes Buchstabens betätigen möchten, können Sie auch lediglich die Taste ALPHA halten und den Namen einfach eingeben. Sie können aber auch die alphabetische Tastatur temporär feststellen und einen kompletten Namen eingeben, bevor Sie diese wieder freigeben. Folgende Tastenkombinationen stellen die alphabetische Tastatur fest:

ALPHA ALPHA stellt die alphabetische Tastatur in Großbuchstaben fest. Wenn die Tastatur so fest eingestellt ist, können Sie Kleinbuchstaben mit gehaltener ← Taste erzeugen, während Sie Sonderzeichen mit gehaltener → Taste eingeben können. Ist die alphabetische Tastatur in Großbuchstaben festgestellt, können Sie diese in Kleinbuchstaben mit der Tastenkombination ← ALPHA umändern.

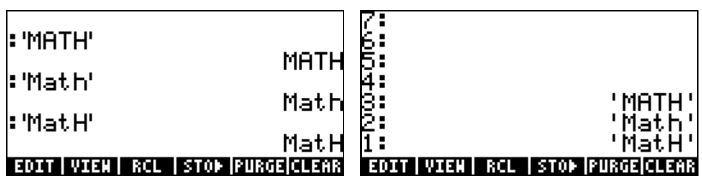
ALPHA ALPHA ← ALPHA stellt die alphabetische Tastatur auf Kleinbuchstaben fest. So eingestellt, müssen Sie, bevor Sie einen Großbuchstaben eingeben möchten, die Taste ← drücken. Um die Eingabe für Kleinbuchstaben rückgängig zu machen, drücken Sie ← ALPHA

Um die Eingabe für Großbuchstaben rückgängig zu machen, drücken Sie ALPHA

Versuchen wir nun einige Beispiele Verzeichnisse/Variablen-Namen in den Stack einzugeben. Angenommen, Sie befinden sich im algebraischen Modus (obwohl diese Anweisungen genauso im RPN-Modus funktionieren), versuchen Sie die nachfolgende Tastenkombination. Mit nachfolgenden Anweisungen werden wir die folgenden Wörter eingeben 'MATH', 'Math' und 'MatH'

' ALPHA ALPHA M A T H ENTER
' ALPHA ALPHA M ← A ← T ← H ENTER
' ALPHA ALPHA M ← ALPHA A T ← H ENTER

Im Display des Taschenrechners wird folgendes angezeigt (links algebraischer Modus, rechts RPN-Modus):



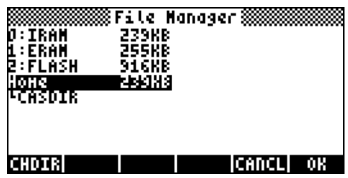
Anmerkung: Wenn Systemflag 60 gesetzt ist, können Sie die alphabetische Tastatur einfach mit der Taste ALPHA feststellen. Weitere Informationen zu Systemflags finden Sie in Kapitel 1.

Erstellen von Unterverzeichnissen

Unterverzeichnisse können entweder in der FILES-Umgebung oder mit dem Befehl CRDIR erstellt werden. Nachfolgend werden beide Ansätze zur Erstellung von Unterverzeichnissen vorgestellt.

Verwenden des FILES-Menüs

Unabhängig vom Operationsmodus des Taschenrechners (algebraisch oder RPN) können wir eine Verzeichnisstruktur basierend auf dem HOME-Verzeichnis, anhand der im FILES-Menü aktivierten Funktionen erstellen. Drücken Sie ← FILES , um das FILES-Menü zu starten. Sofern das HOME-Verzeichnis nicht bereits hervorgehoben ist, d. h.



benutzen Sie die Pfeiltasten (▲ ▼), um es hervorzuheben. Drücken Sie anschließend die Funktionstaste OK (F6). Die Anzeige sieht wie folgt aus:

wobei Sie feststellen können, dass sich zurzeit nur ein Objekt im HOME-Verzeichnis, und zwar das Unterverzeichnis CASDIR, befindet. Erstellen wir nun ein weiteres Unterverzeichnis mit den Namen MANS (für Handbücher – MANualS) in welchem wir die Variablen, welche wir im Laufe dieser Anleitung erstellen, speichern möchten. Um dieses Unterverzeichnis zu erstellen, drücken Sie erst: NXT NEW (F3). Sie erhalten die folgende Eingabemaske:

Das Eingabefeld *Object*, das erste Eingabefeld in der Maske und ist standardmäßig hervorgehoben. Dieses Feld kann den Inhalt einer neuen Variablen, die wir erstellen, enthalten. Da es noch keine Inhalte für das neue

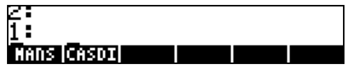
Unterverzeichnis an dieser Stelle gibt, überspringen wir dieses Eingabefeld einfach mit der Pfeiltaste ▼ einmal. Nun wird das Feld *Name* hervorgehoben.

An dieser Stelle geben wir den Namen des neuen Unterverzeichnisses (oder der Variablen, welches auch der Fall sein mag) wie folgt ein:
ALPHA ALPHA M A N S ENTER

Der Cursor springt ins Kontrollfeld *_Directory*. Drücken Sie die Funktionstaste ✓CHK (F3) um anzugeben, dass Sie ein Verzeichnis erstellen und anschließend OK, um die Eingabemaske zu verlassen. Die Variablenauflistung für das HOME-Verzeichnis können Sie in der nachfolgenden Anzeige sehen:

In der Anzeige sehen Sie, dass ein neues Verzeichnis (MANS) innerhalb des HOME-Verzeichnisses existiert.

Als Nächstes erstellen wir ein Unterverzeichnis mit dem Namen INTRO (für Einführung – INTROduction) innerhalb des Verzeichnisses MANS, zur Speicherung der Variablen unserer Beispiele der nächsten Abschnitte dieses Kapitels. Drücken Sie die Taste ON, um zur Normalanzeige des Taschenrechners (das Menü TOOLS wird angezeigt) zurückzukehren. Drücken Sie dann VAR, um die Inhalte des HOME-Verzeichnisses in den Funktionstasten anzuzeigen. Die Anzeige könnte wie folgt aussehen (haben Sie weitere Variablen innerhalb des HOME-Verzeichnisses erstellt, werden diese ebenfalls bei den Funktionstasten angezeigt):



Um in MANS-Verzeichnis zu wechseln, drücken Sie die entsprechende Funktionstaste (in diesem Fall die F1) und, falls im algebraischen Modus, die Taste ENTER. In der zweiten Zeile der Verzeichnisstruktur wird {HOME MANS} angezeigt. Die Funktionstasten aber, werden keine Beschriftung, wie unten gezeigt aufweisen, weil noch keine Variablen für dieses Verzeichnis gespeichert wurden.
Erstellen wir nun ein neues Unterverzeichnis INTRO, indem wir wie folgt vorgehen:

← FILES OK NXT NEW ▼ ALPHA ALPHA I N T R O ENTER ✓CHK OK

Drücken Sie die Taste ON gefolgt von der Taste VAR, um sich den Inhalt des Verzeichnisses MANS wie folgt anzuzeigen:

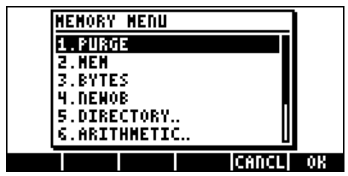
Drücken Sie nun die Funktionstaste INTRO, um ins Unterverzeichnis INTRO zu wechseln. Ein leeres Verzeichnis wird angezeigt. Später werden wir einige Beispiele zur Erstellung von Variablen erzeugen.

Verwenden des Befehls CRDIR

Mit dem Befehl CRDIR können Verzeichnisse erstellt werden. Diesen Befehl erreicht man über die Taste Befehlskatalog (die Taste → CAT, zweite in der vierten Reihe von oben), über die Programmiermenüs (die Taste ← PRG entspricht der Taste → CAT) oder einfach durch direktes Eintippen.

- Über die Taste Katalog
  Drücken Sie → CAT ALPHA C. Benutzen Sie die Pfeiltasten (▲▼), um CRDIR zu finden. Drücken Sie die Funktionstaste OK, um den Befehl zu starten.
- Über die Programmiermenüs
  Drücken Sie ← PRG. Auf diese Wese erhalten Sie das nachfolgende Aktionsmenü zur Programmierung:

Benutzen Sie die Pfeiltaste (▼), um Option *2. MEMORY…* auszuwählen oder einfach nur die 2. Drücken Sie anschließend OK. Dadurch erhalten Sie das nachfolgende Pull-Down-Menü:

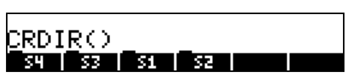


Benutzen Sie die Pfeiltaste (▼), um Option *5. DIRECTORY…* auszuwählen oder einfach nur die 5. Drücken Sie anschließend OK. Dadurch erhalten Sie das nachfolgende Pull-Down-Menü:

Benutzen Sie die Pfeiltaste (▼), um Option *5. CRDIR …* auszuwählen und dann OK.

Befehl DRDIR im algebraischen Modus

Sobald Sie mit einer der obengenannten Möglichkeiten das CRDIR ausgewählt haben, steht Ihnen dieser Befehl im Stack wie folgt zur Verfügung:



An dieser Stelle, müssen Sie einen Verzeichnisnamen, sagen wir *chap1*, eingeben:

ALPHA ALPHA ← ALPHA C H A P 1 ALPHA ENTER

Der Name des neuen Verzeichnisses wird im Funktionstastenmenü angezeigt, z. B.

Befehl DRDIR im RPN-Modus

Um CRDIR im RPN-Modus zu benutzen, muss, bevor Sie den Befehl starten, der Name des Verzeichnisses im Stack bereits existieren. So zum Beispiel:

ALPHA ALPHA ← ALPHA C H A P 2 ALPHA ENTER

Starten Sie den Befehl CRDIR mit einer der oben genannten Möglichkeiten, z. B. über die Taste → CAT :

Drücken Sie die Funktionstaste OK, um den Befehl zur Erstellung des Unterverzeichnisses zu aktivieren:

Zwischen den Unterverzeichnissen hin und her wechseln

Um in der Verzeichnisstruktur weiter nach unten zu wechseln, müssen Sie die, dem Unterverzeichnis entsprechende Funktionstaste, in welches Sie wechseln möchten, eintippen. Die Auflistung der Variablen in einem Unterverzeichnis, erhalten Sie, wenn Sie die Taste VAR (VARiablen) drücken. Um in ein übergeordnetes Verzeichnis der Funktion zu wechseln, benutzen Sie die Funktion UPDIR, d. h. Sie geben ← UPDIR ein.

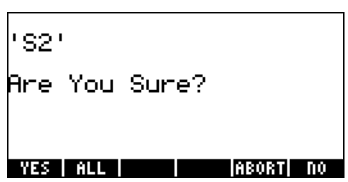
Alternativ, können Sie auch das FILES-Menü dazu benutzen, d. h. Sie drücken ← FILES . Benutzen Sie die Pfeiltasten (▲ ▼), um das Unterverzeichnis, in welches Sie wechseln möchten, auszuwählen, und anschließend CHDIR (Change DIRectory = Verzeichnis wechseln) oder die Funktionstaste F1 . Dadurch werden Ihnen die Inhalte des Unterverzeichnisses, in welches Sie gewechselt haben, im Funktionstastenmenü angezeigt.

Löschen von Unterverzeichnissen

Um ein Unterverzeichnis zu löschen, verwenden Sie eine der nachfolgenden Prozeduren:

Verwenden des FILES-Menüs

Drücken Sie die Taste ← FILES , um das FILES-Menü auszuwählen. Wählen Sie das Verzeichnis, in welchem sich das Unterverzeichnis, das Sie löschen möchten, befindet und drücken dann, falls nötig CHDIR. Das FILES Menü wird geschlossen und die Inhalte des ausgewählten Verzeichnisses angezeigt. In diesem Fall müssen Sie die Taste ENTER drücken. Drücken Sie die Funktionstaste OK, um sich die Inhalte des Verzeichnisses auf dem Display anzuzeigen. Wählen Sie das Unterverzeichnis (oder die Variable), das Sie löschen möchten, aus. Drücken Sie NXT PURGE. Sie sehen eine ähnliche Anzeige, wie unten:



Die Zeichenfolge 'S2' ist in diesem Fall der Name des zu löschenden Unterverzeichnisses. Das Funktionstastenmenü enthält folgende Optionen:

YES	(F1)	Fahren Sie mit dem Löschen des Unterverzeichnisses (der Variablen) fort
ALL	(F2)	Fahren Sie mit dem Löschen aller Unterverzeichnisse

(Variablen) fort
ABORT (F5) Unterverzeichnis (Variable) nicht aus einer Liste löschen
NO (F6) Unterverzeichnis (Variable) nicht löschen

Nachdem Sie nun einen dieser vier Befehle ausgewählt haben, kommen Sie zur Anzeige der Inhalte des Unterverzeichnisses zurück. Der Befehl ABORT, bringt jedoch eine Fehlermeldung:

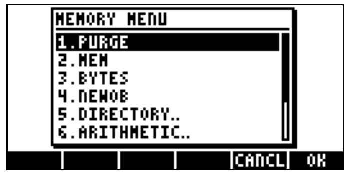
und Sie müssen die Taste OK drücken, bevor Sie zur Auflistung der Variablen zurückkehren.

Verwenden des Befehls PGDIR

Mit dem Befehl PGDIR können Verzeichnisse bereinigt werden. Genau wie der Befehl CRDIR, ist der Befehl PGDIR über die Taste ‿CAT oder über die Taste ‿PRG auswählbar oder kann direkt eingegeben werden.

- Über die Taste Katalog
  Drücken Sie ‿CAT ALPHA ALPHA P G . Der Befehl PGDIR sollte nun hervorgehoben sein. Drücken Sie die Funktionstaste OK, um den Befehl zu starten.
- Über die Programmiermenüs
  Drücken Sie ‿PRG . Dadurch erhalten Sie das nachfolgende Aktionsmenü zur Programmierung:

Benutzen Sie die Pfeiltaste (▼), um Option *2. MEMORY…* auszuwählen und dann OK. Dadurch erhalten Sie das nachfolgende Pull-Down-Menü:



Benutzen Sie die Pfeiltaste (▼), um Option *5. DIRECTORY …* auszuwählen. Drücken Sie anschließend OK. Dadurch erhalten Sie das nachfolgende Pull-Down-Menü:

Benutzen Sie die Pfeiltaste (▼), um Option *6. PGDIR…* auszuwählen und dann OK.

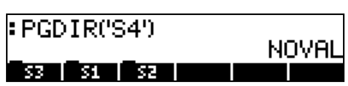
Befehl PGDIR im algebraischen Modus

Sobald Sie mit einer der obengenannten Möglichkeiten das PGDIR ausgewählt haben, steht Ihnen dieser Befehl im Stack wie folgt zur Verfügung:

An dieser Stelle müssen Sie den Namen eines existierenden Verzeichnisses eingeben, sagen wir *S4*:

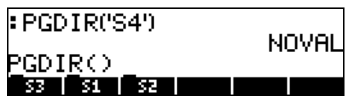
ALPHA S 4 ENTER

Als Ergebnis wird das Unterverzeichnis S4 gelöscht:

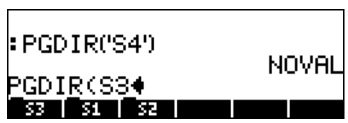


Anstatt den Namen des Verzeichnisses einzutippen, können Sie auch einfach die entsprechende Funktionstaste aus der Auflistung der PGDIR ( ) Befehle drücken, z. B.

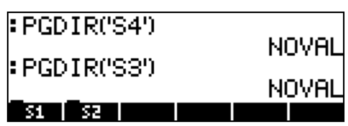
Drücken Sie OK, um Nachfolgendes zu erhalten:



Anschließend drücken Sie S3, um 'S3' als das Argument zu PGDIR einzugeben.



Drücken Sie ENTER, um das Unterverzeichnis zu löschen:



Befehl PGDIR im RPN-Modus

Um PGDIR im RPN-Modus zu benutzen, muss, bevor Sie den Befehl starten, der Name des Verzeichnisses im Stack bereits existieren. So zum Beispiel:
' ALPHA S 2 ENTER

Starten Sie den Befehl PGDIR mit einer der oben genannten Möglichkeiten, z. B. über die Taste → CAT :

Drücken Sie die Funktionstaste OK, um den Befehl zum Löschen des Unterverzeichnisses zu aktivieren:

Anwendung des PURGE Befehls aus dem TOOL-Menü

Das TOOL-Menü erreicht man durch Drücken der Taste TOOL (algebraischer und RPN-Modus werden angezeigt):

Der Befehl PURGE kann über die Funktionstaste PURGE (F5) aktiviert werden. In den nachfolgenden Beispielen möchten wir das Unterverzeichnis *S1* löschen:

- Algebraischer Modus: Drücken Sie PURGE VAR S1 ENTER
- RPN-Modus: Drücken Sie VAR ' S1 ENTER TOOL PURGE VAR

Variablen

Variablen sind ähnlich wie Dateien auf der Festplatte eines Computers. Eine Variable kann ein Objekt speichern (numerische Werte, algebraische Ausdrücke, Listen, Vektoren, Matrizen, Programme usw.). Auch Unterverzeichnisse können als Variablen dargestellt werden (eigentlich ist ein Unterverzeichnis im Taschenrechner gleichzeitig eine Art Taschenrechner-Objekt).

Variablen werden über deren Namen aufgerufen, welche aus einer beliebigen Kombination von Buchstaben und Zahlen bestehen können, wobei aber der Anfangsbuchstabe immer ein Buchstabe sein muss (englisch oder griechisch). Einige Sonderzeichen, wie z. B. der Pfeil (→) kann im Variablennamen verwendet werden, aber nur in Kombination mit einem

Buchstaben. Somit ist '→A' ein gültiger Name für eine Variable, '→' hingegen nicht. Beispiele von gültigen Variablennamen sind: 'A', 'B', 'a', 'b', 'α', 'β', 'A1', 'AB12', '→A12','Vel','Z0','z1', usw.

Eine Variable kann nicht denselben Namen wie eine Funktion im Taschenrechner haben. Sie können also keine Variable mit den Namen SIN erstellen, weil ein Befehl mit dem Namen SIN im Taschenrechner existiert. Nachfolgend ist eine Auflistung reservierter Variablennamen im Taschenrechner: ALRMDAT, CST, EQ, EXPR, IERR, IOPAR, MAXR, MINR, PICT, PPAR, PRTPAR, VPAR, ZPAR, der_, e, i, n1,n2, …, s1, s2, …, ΣDAT, ΣPAR, π, ∞

Variablen können in Unterverzeichnissen organisiert werden.

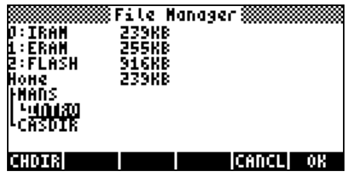
Erstellen von Variablen

Genau wie wir dies mit Unterverzeichnissen bisher gemacht haben, lassen sich auch Variablen über das FILES-Menü erstellen. So z. B. möchten wir in einem zuvor erstellten Unterverzeichnis {HOME MANS INTRO} nachfolgende Variablen mit den unten aufgeführten Werten erstellen.

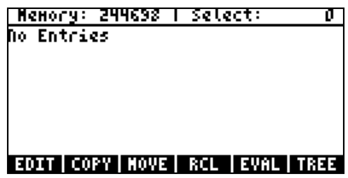
Name	*Inhalt*	*Typ*
A	12.5	reell
α	-0.25	reell
A12	3×10^5	reell
Q	'r/(m+r)'	Algebraik
R	[3,2,1]	Vektor
z1	3+5i	komplex
p1	<< → r 'π*r^2' >>	Programm

Verwenden des FILES-Menüs

Benutzen wir das FILES-Menü, um die Variable A einzugeben. Nehmen wir an, dass wir uns im Unterverzeichnis {HOME MANS INTRO} befinden. Um in dieses Unterverzeichnis zu gelangen, drücken Sie: ← FILES und wählen Sie das Unterverzeichnis INTRO, wie unten gezeigt:



Drücken Sie OK um ins Verzeichnis zu gelangen. Sie bekommen eine Anzeige "no entries" – keine Einträge (das Unterverzeichnis INTRO ist an dieser Stelle noch leer)



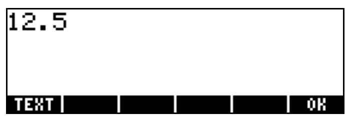
Drücken Sie die Taste NXT, um zum nächsten Funktionstastenmenü zu gelangen und drücken Sie die Funktionstaste NEW. Sie erhalten die folgende Eingabemaske für NEW VARIABLE (neue Variable):

Um die Variable A (siehe Tabelle oben) einzugeben, müssen wir zuerst den Inhalt dieser eingeben, und zwar die Zahl 12,5 und anschließend deren Namen, wie folgt: 1 2 . 5
OK ALPHA A OK. Die Anzeige sieht wie folgt aus:

Drücken Sie OK ein weiteres Mal, um die Variable zu erstellen. Die neue Variable wird wie folgt in der Auflistung angezeigt:

Die Auflistung zeigt eine reelle Variable (|R) mit dem Namen A, welche einen Speicherplatz von 10,5 Byte belegt. Um sich den Inhalt der Variablen anzuzeigen, drücken Sie NXT VIEW.

- Drücken Sie die Funktionstaste GRAPH (F1), um den Inhalt in grafischer Form darzustellen.
-



- Drücken Sie die Funktionstaste TEXT (F1), um den Inhalt in Textform darzustellen.
- Drücken Sie OK, um zur Variablenliste zurückzukehren
- Drücken Sie ON ein weiteres Mal, um zur Normalanzeige zurückzukehren. Die Variable A sollte nun bei den Funktionstasten angezeigt werden:

Verwenden des Befehls STO▶

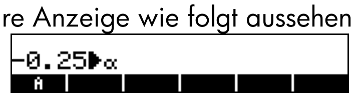
Ein einfacherer Weg eine Variable zu erstellen, ist über den Befehl STO (d. h. die Taste STO▶). Für die Erstellung der noch fehlenden Variablen, werden die Beispiele im algebraischen und im RPN-Modus angezeigt.

Name	***Inhalt***	***Typ***
α	-0.25	reell

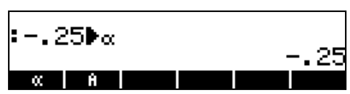
A12	3×10^5	reell
Q	‘r/(m+r)'	Algebraik
R	[3,2,1]	Vektor
z1	3+5i	komplex
p1	<< → r 'π*r^2' >>	Programm

Algebraischer Modus

Drücken Sie nachfolgende Tastenfolge, um den Wert -0,25 in die Variable α zu speichern: [0] [.] [2] [5] [+/-] [STO▶] [ALPHA] [↱] [A] . An dieser Stelle wird Ihre Anzeige wie folgt aussehen:



Dieser Ausdruck bedeutet, dass der Wert -0,25 in α gespeichert wird (das Symbol ▶ deutet darauf hin). Drücken Sie nun [ENTER], um die Variable zu erzeugen. Die Variable wird nun bei den Funktionstasten angezeigt.



Nachfolgend die Tastenkombinationen zur Eingabe der noch verbleibenden Variablen:

A12: [3] [EEX] [5] [STO▶] [ALPHA] [A] [1] [2] [ENTER]

Q: ['] [ALPHA] [↰] [R] [÷] [↰] [( )] [ALPHA] [↰] [M] [+] [ALPHA] [↰] [R] [▶] [▶] [STO▶] [ALPHA] [Q] [ENTER]

R: [↰] [[ ]] [3] [↱] [,] [2] [↱] [,] [1] [▶] [STO▶] [ALPHA] [R] [ENTER]

z1: [3] [+] [5] [×] [↰] [i] [STO▶] [ALPHA] [↰] [Z] [1] [ENTER] (Bestätigen Sie den Wechsel in den *Complex Modus, falls gefragt*).

p1: [↱] [« »] [↱] [→] [ALPHA] [↰] [R] ['] [↰] [π] [×] [ALPHA] [↰] [R] [Y^x] [2] [▶] [▶] [▶] [STO▶] [ALPHA] [↰] [P] [1] [ENTER] ..

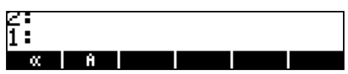
An dieser Stelle, sieht Ihre Anzeige wie folgt aus:

Am unteren Rand werden Sie sechs oder sieben Variablen feststellen: *p1, z1, R, Q, A12, α.*

RPN-Modus

Drücken Sie nachfolgende Tastenfolge, um den Wert -0,25 in eine Variable α zu speichern: [0][.][2][5][+/-][ENTER] [ALPHA][→][A][ENTER]. An dieser Stelle wird Ihre Anzeige wie folgt aussehen:

Dieser Ausdruck bedeutet, dass der Wert -0,25 bereit zur Speicherung nach α ist. Drücken Sie nun [STO▶], um die Variable zu erzeugen. Die Variable wird nun bei den Funktionstasten angezeigt.



Um den Wert 3×10^5 in die Variable A12 einzugeben, kann auch eine kürzere Version der Prozedur benutzt werden:
[3][EEX][5]['][ALPHA][A][1][2][ENTER] [STO▶]
Nachfolgend eine Methode den Inhalt von Q einzugeben:
Q: ['][ALPHA][←][R][÷][←][()]
[ALPHA][←][M][+][ALPHA][←][R][▶][▶] ['][ALPHA][Q][ENTER] [STO▶]
Um den Wert von R einzugeben, kann auch eine kürzere Prozedur benutzt werden:
R: [←][[]] [3][SPC][2][SPC][1][▶] ['][ALPHA][R] [ENTER][STO▶]
Beachten Sie bitte, dass man die Elemente eines Vektors im RPN-Modus besser durch einen Leerschritt ([SPC]), als durch das Komma ([→][,]) voneinander trennt, wie dies im algebraischen Modus verwendet wurde.

z1: ['] [3] [+] [5] [×] [←] i ['] [ALPHA] [←] [Z] [1] [STO▶]
(Bestätigen Sie den Wechsel in den *Complex Modus, falls gefragt*).
p1: [→] « » [→] → [ALPHA] [←] [R] ['] [←] π [×]
[ALPHA] [←] [R] [y^x] [2] [▶] [▶] [▶] ['] [ALPHA] [←] [P] [1] [▶] [ENTER] [STO▶].
An dieser Stelle, sieht Ihre Anzeige wie folgt aus:

Am unteren Rand werden Sie sechs oder sieben Variablen feststellen: *p1, z1, R, Q, A12, α*.

Überprüfen der Inhalte von Variablen

Als Übung in den Inhalt einer Variablen reinzuspähen, werden wir die sieben vorhin eingegeben Variablen benutzen. Wir haben gezeigt wie das FILES-Menü dazu verwendet wird den Inhalt der Variablen A aus einer früheren Übung anzusehen. In diesem Abschnitt zeigen wir einen einfacheren Weg in den Inhalt der Variablen reinzuschauen.

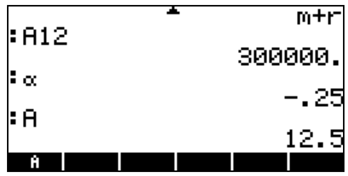
Funktionstaste für die Variable drücken

Diese Prozedur zeigt den Inhalt einer Variablen an, sofern diese einen numerischen, algebraischen Wert oder ein Array enthält. So zum Beispiel können Sie zur Überprüfung des Inhalts der oben aufgeführten Variablen nachfolgende Tasten drücken:

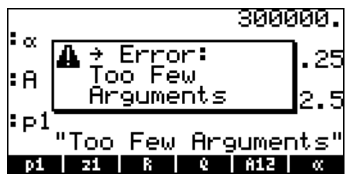
Algebraischer Modus

Drücken Sie nachfolgende Tastenfolge: [VAR] [z1] [ENTER] [R] [ENTER] [Q] [ENTER]. An dieser Stelle sieht Ihre Anzeige wie folgt aus:

Als Nächstes, drücken Sie nachfolgende Tastenfolge: A12 ENTER α ENTER NXT A ENTER. An dieser Stelle sieht Ihre Anzeige wie folgt aus:



Wenn Sie nun die Funktionstaste, die *p1* entspricht, drücken, erhalten Sie eine Fehlermeldung (versuchen Sie es mit NXT p1 ENTER)



Anmerkung: Durch Drücken von p1 ENTER versuchen wir das Programm *p1* zu starten (run). Dieses Programm aber erwartet eine numerische Eingabe. Versuchen Sie folgendes Beispiel: ON p1 ← () 5 ENTER. Die Lösung lautet:

Das Programm hat folgende Struktur: << → r 'π*r^2' >>
Die Symbole « » weisen auf ein Programm in der RPL-Sprache hin (die ursprüngliche Programmiersprache der HP 28/48 Taschenrechner und auch in der 49G Reihe der HP Taschenrechner vorhanden). Die Zeichen → r weisen darauf hin, dass eine Eingabe, welche als *r* bezeichnet wird, dem Programm zur Verfügung gestellt wird. Das Programm nimmt den Wert *r* und berechnet den algebraischen Ausdruck 'π*r^2'. Im obigen Beispiel nimmt *r* den Wert 5 an, somit wird der Wert von πr^2 als $\pi \cdot 25$ wiedergegeben. Dieses

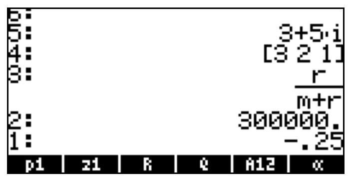
Programm berechnet somit eine Kreisfläche mit einem gegebenen Radius von *r*.

RPN-Modus

Im RPN-Modus, müssen Sie lediglich die entsprechende Funktionstaste drücken um den Inhalt einer numerischen oder algebraischen Variablen zu erhalten. Im vorliegenden Fall, können wir versuchen in die oben erstellten Variablen *z1*, *R*, *Q*, *A12*, α, und *A*, wie folgt hineinzuspähen:

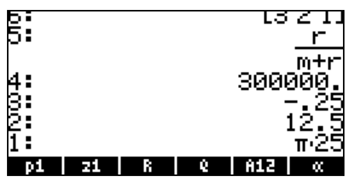
VAR z1 R Q A12 α

An dieser Stelle sieht Ihre Anzeige wie folgt aus:



Um den Inhalt von A anzuzeigen, drücken Sie NXT A

Um das Programm *p1* mit r = 5 zu starten, drücken Sie: NXT 5 p1.



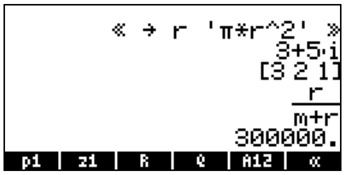
Beachten Sie, um das Programm im RPN-Modus auszuführen, müssen wir nun die Eingabe (5) machen und die entsprechende Funktionstaste drücken. (Im algebraischen Modus, müssen Klammern zur Eingabe des Argumentes gesetzt werden).

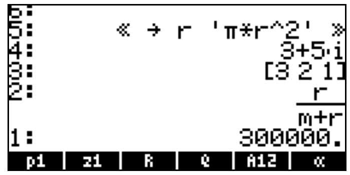
Verwendung der rechten Shift-Taste ↱ gefolgt von der entsprechenden Funktionstaste.

Dieser Ansatz den Inhalt einer Variablen anzusehen, funktioniert gleichermaßen im algebraischen wie auch im RPN-Modus. Versuchen Sie nachfolgende Beispiele in beiden Modi:

VAR ↱ p1 ↱ z1 ↱ R ↱ Q ↱ A12

Nach obiger Eingabe erscheint folgende Anzeige (algebraischer Modus links und RPN-Modus rechts)

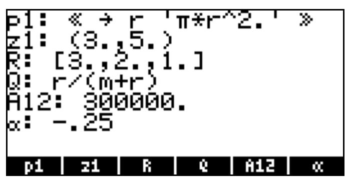




Beachten Sie, dass in diesem Fall der Inhalt des Programms p1 in der Anzeige erscheint. Um die noch verbleibenden Variablen in diesem Verzeichnis anzusehen, gehen Sie folgendermaßen vor:

→ NXT →

Anzeigen der Inhalte aller Variablen im Display
Benutzen Sie die Tastenkombination → ▼ um den Inhalt aller Variablen auf dem Display anzuzeigen. So zum Beispiel:



Drücken Sie die Taste ON, um zur Normalanzeige des Taschenrechners zurückzukehren.

Inhalte von Variablen ersetzen

Der Austausch des Variableninhalts, ist als Speichern eines unterschiedlichen Wertes in demselben Variablennamen zu betrachten. Somit, können wir die oben erstellten Variablen als Beispiele zur Veranschaulichung des Austauschs des Variableninhalts nehmen.

Verwenden des Befehls STO▶
Zur Veranschaulichung nehmen wir die sechs vorhin erstellten Variablen *p1, z1, R, Q, A12, a* und *A* und fahren fort den Inhalt der Variablen *A12* (im

Moment eine numerische Variable) mit dem algebraischen Ausdruck ‘β/2’, unter Verwendung des Befehls STO▶ zu ändern. Zuerst im algebraischen Modus:

' ALPHA ↱ B ÷ 2 ▶ STO▶ A12 ENTER

Überprüfen Sie den neuen Inhalt der Variablen *A12* mithilfe von ↱ A12 .

Im RPN-Modus:

' ALPHA ↱ B ÷ 2 ENTER ' A12 ENTER STO▶

oder vereinfacht

' ALPHA ↱ B ÷ 2 ▶ ' A12 STO▶

Verwenden der linken Shift-Taste ↰ gefolgt von der Funktionstaste der Variablen (RPN)

Dies ist eine äußerst einfache Art den Inhalt von Variablen zu ändern, funktioniert aber nur im RPN-Modus. Die Prozedur besteht darin, den neuen Inhalt der Variablen zu tippen und in den Stack einzugeben, anschließend die linke Shift-Taste, gefolgt von der der Variablen zugeordneten Funktionstaste zu drücken. So z. B., wenn wir im RPN-Modus den Inhalt der Variablen *z1* auf ‘*a+b·i* ’ ändern möchten, verwenden wir:

' ALPHA ↰ A + ALPHA ↰ B × ↰ i ENTER

Dadurch wird der algebraische Ausdruck ‘*a+b·i* ’ in Stack-Ebene 1 eingetragen. Um dieses Ergebnis in die Variable *z1* einzugeben, verwenden wir: VAR ↰ z1

Um uns den Inhalt von *z1* anzusehen, verwenden wir: ↱ z1

Ein äquivalenter Weg dies im algebraischen Modus zu tun ist folgender:

ALPHA ↰ A + ALPHA ↰ B × ↰ i ENTER STO▶ z1 ENTER

Um uns den neuen Inhalt von *z1* anzusehen, verwenden wir: ↱ z1

Verwenden der Variablen ANS(1) (Algebraischer Modus)

Im algebraischen Modus können wir die Variable ANS(1) verwenden, um den Inhalt einer Variablen auszutauschen. So z. B. ist die Prozedur, den Inhalt von *z1* auf '*a+bi*' abzuändern, folgende: ← ANS STO▶ z1 ENTER. Um uns den neuen Inhalt von *z1* anzusehen, verwenden wir: → z1

Kopieren von Variablen

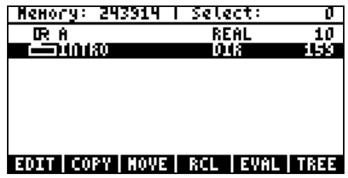
Die nachfolgenden Übungen zeigen uns verschiedene Wege Variablen aus einem Unterverzeichnis in ein anderes zu kopieren.

Verwenden des FILES-Menüs

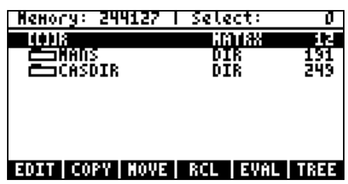
Um eine Variable von einem Unterverzeichnis in ein anderes zu kopieren können wir das FILES-Menü verwenden. So haben wir z. B. im Unterverzeichnis {HOME MANS INTRO} folgende Variablen *p1, z1, R, Q, A12,* α und *A*. Angenommen, wir möchten die Variable *A* kopieren und eine Kopie dieser im Unterverzeichnis {HOME MANS} ablegen. Weiterhin werden wir die Variable *R* kopieren und eine Kopie im HOME-Verzeichnis ablegen. So führen Sie den Vorgang durch: Drücken Sie ← FILES OK, um nachfolgende Variablenliste zu erzeugen:

Benutzen Sie die Pfeiltaste ▼ um die Variable A auszuwählen (es ist die letzte in der Auflistung), anschließend drücken Sie COPY. Der Taschenrechner wird sich mit der Anzeige 'PICK DESTINATION:' (Ziel wählen) melden

Benutzen Sie die Pfeiltaste (▲), um das Unterverzeichnis MANS auszuwählen und drücken Sie dann OK. Drücken Sie nun ← UPDIR, erscheint in der Anzeige der Inhalt des Unterverzeichnisses MANS (beachten Sie dass die Variable *A*, wie erwartet, in der Liste auftaucht):



Drücken Sie ON INTRO ENTER (im algebraischen Modus) oder ON INTRO (im RPN-Modus), um zum Verzeichnis INTRO zurückzukehren. Drücken Sie ← FILES OK, um die Variablenliste des Verzeichnisses {HOME MANS INTRO} zu erzeugen: Benutzen Sie die Pfeiltaste (▼), um die Variable *R* auszuwählen und dann COPY. Benutzen Sie die Pfeiltaste (▲), um das Verzeichnis HOME auszuwählen und drücken dann OK. Drücken Sie nun ← UPDIR zweimal, erscheint in der Anzeige das HOME-Verzeichnis, in welchem sich eine Kopie von *R* befindet.



History im algebraischen Modus verwenden

Nachfolgend einen Weg um mithilfe von History (Stack) eine Variable aus einem Verzeichnis in ein anderes zu kopieren, wenn sich der Taschenrechner im algebraischen Modus befindet. Angenommen, wir befinden uns im Unterverzeichnis {HOME MANS INTRO} und möchten den Inhalt der Variablen *z1* ins Unterverzeichnis {HOME MANS} kopieren. Verwenden Sie dazu folgende Tastenfolge: → z1 STO▸ z1 ENTER. Dadurch werden einfach die Inhalte von *z1* in sich selbst gespeichert (*z1* bleibt unverändert). Als Nächstes, verwenden Sie ← UPDIR ENTER um ins {HOME MANS}

Unterverzeichnis zu wechseln. Die Anzeige des Taschenrechners sieht wie folgt aus:

Benutzen Sie die Löschtaste ◄ ◄ ◄ (dreimal), um die letzten drei Zeilen im Display zu entfernen. An dieser Stelle ist der Stack bereit den Befehl ANS(1)▶z1 auszuführen. Drücken Sie ENTER, um den Befehl auszuführen. Verwenden Sie anschließend ↱, um den Inhalt der Variablen zu überprüfen.

Verwendung des Stacks im RPN-Modus

Um die Verwendung des Stacks beim Kopieren einer Variablen aus einem Unterverzeichnis in ein anderes zu demonstrieren, nehmen wir an Sie befinden sich im Unterverzeichnis {HOME MANS INTRO} und wir den Inhalt der Variablen *z1* ins HOME-Verzeichnis kopieren möchten. Verwenden Sie nachfolgende Prozedur: ↱ ENTER ' ENTER

Damit zeigen Sie den Inhalt und den Namen der Variablen im Stack an. Die Anzeige des Taschenrechners sieht wie folgt aus:

Verwenden Sie nun ↰ UPDIR ↰ UPDIR um ins HOME-Verzeichnis zu wechseln und drücken Sie STO▸, um den Vorgang zu beenden. Mit ↱ können Sie den Inhalt der Variablen überprüfen.

Zwei oder mehrere Variablen im algebraischen Modus in den Stack kopieren

Nachfolgende Übung ist zur Demonstration des Kopiervorgangs, zweier oder mehrerer Variablen über den Stack im algebraischen Modus, gedacht. Angenommen, wir befinden uns im Unterverzeichnis {HOME MANS INTRO} und möchten die Variablen *R* und *Q* ins Unterverzeichnis {HOME MANS} kopieren. Die dafür erforderlichen Tastenanschläge, um diesen Vorgang abzuschließen sind wie folgt:

[→] [R] [STO▶] [R] [ENTER]
[→] [Q] [STO▶] [Q] [ENTER]
[←] UPDIR [ENTER]
[◄] [◄] [◄] [ENTER]
[◄] [◄] [◄] [◄] [ENTER]

Um den Inhalt der Variablen zu überprüfen, verwenden Sie [→] [R] und [→] [Q].
Dieser Vorgang kann verallgemeinert werden, um drei oder mehrere Variablen zu kopieren.

Zwei oder mehrere Variablen im RPN-Modus in den Stack kopieren

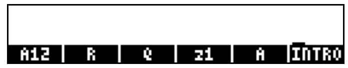
Nachfolgende Übung ist zur Demonstration des Kopiervorgangs, zweier oder mehrerer Variablen über den Stack im RPN-Modus, gedacht. Erneut nehmen wir an, dass wir uns im Unterverzeichnis {HOME MANS INTRO} befinden und die Variablen *R* und *Q* ins Unterverzeichnis {HOME MANS} kopieren möchten. Die dafür erforderlichen Tastenanschläge, um diesen Vorgang abzuschließen, sind wie folgt:

[→] [R] ['] [R] [ENTER]
[→] [Q] ['] [Q] [ENTER]
[←] UPDIR [STO▶] [STO▶]

Um den Inhalt der Variablen zu überprüfen, verwenden Sie [→] [R] und [→] [Q].
Dieser Vorgang kann verallgemeinert werden, um drei oder mehrere Variablen zu kopieren.

Die Variablen in einem Verzeichnis neu anordnen

In diesem Abschnitt veranschaulichen wir, wie wir den Befehl ORDER zur Neuanordnung der Variablen in einem Verzeichnis verwenden. Nehmen wir an, dass wir vom Unterverzeichnis {HOMES MANS}, in welchem sich die Variablen *A12, R, Q, z1, A* und das Unterverzeichnis *INTRO* befinden, ausgehen – wie unten gezeigt. (Kopieren Sie A12 von INTRO nach MANS).

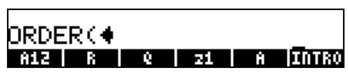


Algebraischer Modus

In diesem Fall befindet sich der Taschenrechner im algebraischen Modus. Angenommen, wir möchten die Anordnung der Variablen auf *INTRO, A, z1, Q, R, A12* ändern. Um die Funktion ORDER zu starten, gehen Sie wie folgt vor:

← PRG ▼ OK	Wählen Sie MEMORY aus dem Programmiermenü
▼▼▼▼ OK	Wählen Sie DIRECTORY aus dem Menü MEMORY
▲▲ OK	Wählen Sie ORDER aus dem Menü DIRECTORY

In der Anzeige wird die folgende Eingabezeile aufgezeigt:



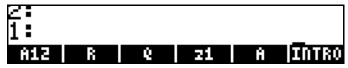
Als Nächstes lassen wir uns die neue Anordnung der Variablen anzeigen, indem wir deren Namen in Anführungszeichen setzen:

← {} ' INTRO ▶ → , ' A
▶ → , ' z1 ▶ → , ' Q ▶
→ , ' R ▶ → , ' A12 ENTER

Im Display sehen Sie die neue Anordnung der Variablen:

RPN-Modus

Im RPN-Modus, wird die Auflistung der neu angeordneten Variablen vor Anwendung des Befehls ORDER angezeigt. Angenommen, wir beginnen mit derselben Situation wie oben, aber im RPN-Modus, d. h.



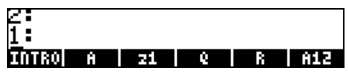
Die neu angeordnete Liste wird wie folgt erzeugt:

{} INTRO A z1 Q R A12 ENTER

Anschließend geben Sie den Befehl ORDER, wie vorhin ein, d. h.

PRG ▼ OK Wählen Sie MEMORY aus dem Programmiermenü
▼▼▼▼ OK Wählen Sie DIRECTORY aus dem Menü MEMORY
▲▲ OK Wählen Sie ORDER aus dem Menü DIRECTORY

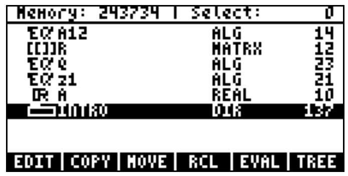
Das Ergebnis ist die nachfolgende Anzeige:



Verschieben von Variablen über das FILES-Menü

Um eine Variable von einem Unterverzeichnis in ein anderes zu verschieben können wir das Menü FILES verwenden. So haben wir z. B. im Unterverzeichnis {HOME MANS INTRO} folgende Variablen *p1, z1, R, Q, A12,* α und *A*. Angenommen, wir möchten die Variable *A12* ins Unterverzeichnis {HOME MANS} zu kopieren. So führen Sie den Vorgang durch: Drücken Sie FILES OK, um die Variablenliste anzuzeigen. Benutzen Sie die Pfeiltaste ▼, um die Variable *A12* auszuwählen und dann MOVE. Der Taschenrechner wird sich mit der Anzeige 'PICK DESTINATION:' (Ziel wählen) melden. Benutzen Sie die Pfeiltaste ▲, um das Unterverzeichnis MANS auszuwählen und drücken dann OK. Nun sehen Sie den Inhalt des Unterverzeichnisses {HOME MANS INTRO}:

Beachten Sie, dass die Variable A12 nicht mehr da ist. Drücken Sie nun ← UPDIR , wird Ihnen der Inhalt des Unterverzeichnisses MANS, einschließlich der Variablen *A12* angezeigt.



Anmerkung: Mithilfe des Stacks können Sie eine Variable verschieben, indem Sie kopieren mit löschen einer Variable kombinieren. Wie Sie Variablen löschen können, wird im nächsten Abschnitt demonstriert.

Löschen von Variablen

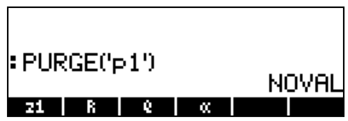
Variablen können mithilfe der Funktion PURGE gelöscht werden. Auf diese Funktion kann direkt mithilfe des TOOL (TOOL) oder FILES-Menüs ← FILES OK zugegriffen werden.

Verwenden des Befehls FILES

Der Befehl FILES kann dazu verwendet werden Variablen einzeln zu löschen. Um eine Variable aus einem vorgegebenen Unterverzeichnis zu löschen, können wir das FILES-Menü verwenden. So sind z. B. im Unterverzeichnis {HOME MANS INTRO} folgende Variablen *p1, z1, R, Q,* α und *A* verblieben. Angenommen, wir löschen die Variable *A*. So führen Sie den Vorgang durch: Drücken Sie ← FILES OK, um die Variablenliste anzuzeigen. Benutzen Sie die Pfeiltaste ▼, um die Variable A auszuwählen (es ist die letzte in der Auflistung), anschließend drücken Sie NXT PURGE YES. Nun sehen Sie den Inhalt des Unterverzeichnisses {HOME MANS INTRO} ohne die Variable A:

Anwenden der Funktion PURGE im Stack im algebraischen Modus

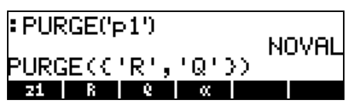
Wir beginnen wieder im Unterverzeichnis {HOME MANS INTRO}, in welchem sich die Variablen *p1, z1, Q, R* und α befinden. Wir wenden nun den PURGE Befehl an, um die Variable p1 zu löschen. Drücken Sie TOOL PURGE VAR p1 ENTER. In der Anzeige wird nun gemeldet, dass die Variable *p1* entfernt wurde:



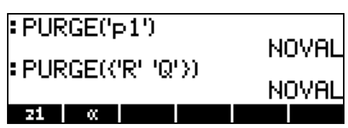
Mit dem PURGE Befehl können Sie mehr als eine Variable löschen, indem Sie deren Namen in die PURGE-Argumentliste eintragen. Wenn wir nun z. B. die Variablen R und Q gleichzeitig löschen möchten, können wir nachfolgende Übung versuchen. Drücken Sie:

TOOL PURGE ← {} ' VAR R ▶ → , ' VAR Q

In der Anzeige erscheint zu diesem Zeitpunkt folgender Befehl, der nun zur Ausführung bereitsteht:



Um den Löschvorgang der Variablen durchzuführen, drücken Sie ENTER. Die Anzeige weist nun nur noch die verbleibenden Variablen auf:



Anwenden der Funktion PURGE im Stack im RPN-Modus

Wir beginnen wieder im Unterverzeichnis {HOME MANS INTRO}, in welchem nun nur noch die Variablen *p1, z1, Q, R* und α vorhanden sind. Wir wenden nun den PURGE Befehl an, um die Variable p1 zu löschen.

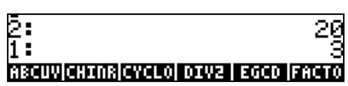
Drücken Sie ' p1 ENTER TOOL PURGE. In der Anzeige wird nun gemeldet, dass die Variable *p1* entfernt wurde:

Um zwei Variablen gleichzeitig zu löschen, sagen wir die Variablen *R* und *Q*, müssen wir zuerst eine Liste erstellen (im RPN-Modus müssen die Elemente der Liste nicht durch Komma, wie im algebraischen Modus, getrennt werden): VAR ← { } ' R ▶ ' Q ENTER. Drücken Sie anschließend TOOL PURGE, um die Variablen zu löschen.

Die Funktionen UNDO und CMD

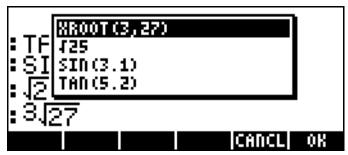
Die Funktionen UNDO und CMD sind dann nützlich, wenn Sie kürzlich gelöschte Befehle wiederherstellen möchten oder eine Operation rückgängig machen möchten, weil Ihnen ein Fehler unterlaufen ist. Diese Funktionen funktionieren in Zusammenhang mit der Taste HIST: UNDO bekommen Sie durch die Tastenfolge → UNDO, während Sie CMD über die Tastenfolge ← CMD erhalten.

Um die Funktion UNDO zu veranschaulichen, versuchen Sie nachfolgende Übung im algebraischen (ALG) Modus: 5 × 4 ÷ 3 ENTER. Der Befehl UNDO (→ UNDO) wird das Ergebnis einfach löschen. Die gleiche Übung im RPN-Modus, erfolgt über nachfolgende Tastenfolge: 5 ENTER 4 ENTER × 3 ENTER ÷. Benutzen Sie → UNDO an dieser Stelle, wird die letzte Operation (20/3) rückgängig gemacht, während die ursprünglichen Dateien im Stack erhalten bleiben:



Um den Befehl CMD zu veranschaulichen, geben wir nachfolgendes im algebraischen Modus ein. Drücken Sie ENTER nach jeder Eingabe.

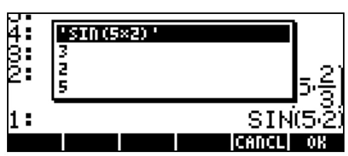
Als Nächstes verwenden Sie die Funktion CMD (← CMD), um die letzten vier Befehle, die der Anwender eingegeben hat, anzuzeigen, d. h.



Mithilfe der Pfeiltasten (▲ ▼) können Sie durch diese Befehle nach oben und nach unten navigieren; dabei können Sie jeden hervorheben, den Sie neu eingeben möchten. Sobald Sie den Befehl ausgewählt haben, den Sie eingeben möchten, drücken Sie OK.
Die Funktion CMD funktioniert im RPN-Modus genauso, ausgenommen dass die Befehlsliste nur Zahlen oder Algebraiks auflisten wird. Die eingegebenen Funktionen werden nicht angezeigt. Versuchen Sie z. B. nachfolgendes Beispiel im RPN-Modus:

5 ENTER 2 ENTER 3 ÷ × SIN
' SIN 5 × 2 ENTER .

Drücken ← CMD erzeugt die nachfolgende Auswahlbox:

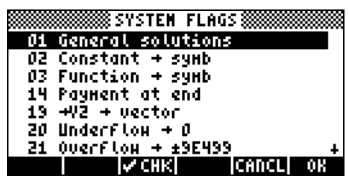


Wie Sie sehen können, werden die in der ersten Berechnung verwendeten Zahlen, 3, 2 und 5 in der Auswahlbox angezeigt, genauso die Algebraik 'SIN(5x2)', aber nicht die Funktion SIN, welche zuvor in die Algebraik eingefügt wurde.

Flags

Ein Flag ist ein Boolescher Wert, welcher gesetzt oder gelöscht werden kann (wahr oder falsch), der eine bestimmte Einstellung des Taschenrechners oder eine Option in einem Programm setzen kann. Flags, werden im Taschenrechner über Zahlen identifiziert. Im Taschenrechner existieren insgesamt 256 Flags, durchnumeriert zwischen -128 und 128. Die positiven Flags werden als Anwenderflags bezeichnet und diesem für Programmierzwecke zur Verfügung gestellt. Die durch negative Zahlen dargestellten Flags werden als Systemflags bezeichnet und beeinflussen die Arbeitsweise des Taschenrechners.

Um sich die aktuellen Systemflageinstellungen anzusehen, drücken Sie die Taste MODE und anschließend die Funktionstaste FLAGS (d. h. F1). Sie erhalten eine Anzeige mit der Überschrift *SYSTEM FLAGS*, in welchem die Zahlen der Flags und die entsprechenden Einstellungen angezeigt werden.



(**Anmerkung:** In dieser Anzeige, da nur Systemflags vorhanden sind, wird lediglich der absolute Wert der Flag Zahl angegeben). Man sagt ein Flag ist gesetzt (*set*), wenn ein Häkchen (✓) vor der Flag-Zahl angezeigt wird. Andernfalls ist das Flag nicht *set (gesetzt)* oder *cleared (initialisiert)*. Um den Status eines Systemflags zu ändern, drücken Sie die Funktionstaste ✓CHK! während das Flag, das Sie verändern möchten, hervorgehoben ist, oder Sie benutzen die Taste +/-. Sie können Sie Pfeiltasten (▲▼) benutzen, um in der Liste der Systemflags zu navigieren.

Obwohl es 128 Systemflags gibt, werden nicht alle benutzt und einige davon gehören einfach nur zur internen Systemsteuerung. Systemflags, auf die der Anwender keinen Zugriff besitzt, werden auch nicht angezeigt. In Kapitel 24 wird eine komplette Liste der Flags vorgestellt.

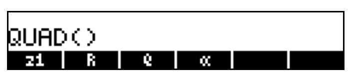
Beispiel einer Flageinstellung allgemeine Lösungen vs. Hauptwert

So z. B. ist der Standardwert für Systemflag 01 *Allgemeine Lösungen* . Dies bedeutet, dass, wenn eine Gleichung mehrere Lösungen hat, werden alle Lösungen vom Taschenrechner angezeigt, höchstwahrscheinlich in einer Liste . Drücken Sie die Funktionstaste ✓CHK können Sie das Systemflag 01 auf Hauptwert setzen. Diese Einstellung zwingt den Taschenrechner einen einzigen Wert anzuzeigen, auch als Hauptwert der Lösung bezeichnet.

Um dies in Aktion zu sehen, setzen Sie zuerst das Systemflag 01 (d. h. wählen Sie *Principal Value (Hauptwert)*). Drücken Sie OK zweimal, um zur Normalanzeige des Taschenrechners zurückzukehren. Versuchen wir eine quadratische Gleichung , sagen wir $t^2+5t+6 = 0$ mit dem Befehl QUAD zu lösen.

Algebraischer Modus

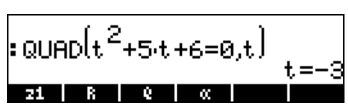
Benutzen Sie folgende Tastenfolge: ↱ _CAT ALPHA Q . (Benutzen Sie dann die Pfeiltasten ▲ ▼ , um QUAD auszuwählen) drücken Sie anschließend OK .



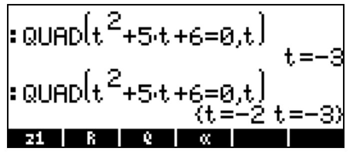
Um die Gleichung als erstes Argument der Funktion QUAD einzugeben, verwenden Sie die Tastenfolge:

↱ _EQW ALPHA ↰ T y^x 2 ▶ + 5 × ALPHA ↰ T + 6 ▲ ▲
↱ _= 0 ENTER
↱ _, ALPHA ↰ T ENTER

Die Lösung lautet:



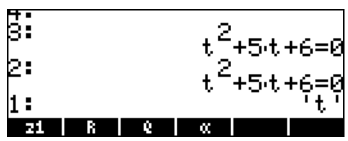
Ändern Sie nun die Einstellung von Flag 1 auf *General solutions (allgemeine Lösungen):* MODE FLAGS ✓CHK OK OK . Versuchen Sie eine weitere Lösung: ▲ ▲ ENTER ENTER . Im Ergebnis werden nun zwei Werte angezeigt:



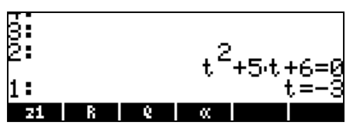
RPN-Modus

Erster Satz Systemflags 01 (d. h. *Principal Value – Hauptwert*). Drücken Sie OK zweimal, um zur Normalanzeige des Taschenrechners zurückzukehren. Anschließend tippen Sie die quadratische Gleichung wie folgt ein:

→ EQW ALPHA ← T y^x 2 ▶ + 5 × ALPHA ← T + 6 ▲ ▲
→ = 0 ENTER
ENTER (behalten Sie eine zweite Kopie im RPN-Stack)
' ALPHA ← T ENTER



Verwenden Sie nachfolgende Tastenfolge, um den QUAD Befehl zu starten: → CAT ALPHA Q . (Benutzen Sie dann die Pfeiltasten ▲▼, um QUAD auszuwählen) drücken Sie anschließend OK . In der Anzeige wird der Hauptwert angezeigt:



Ändern Sie nun die Einstellung von Flag 01 auf *General solutions (allgemeine Lösungen):* MODE FLAGS ✓CHK OK OK . Versuchen Sie eine weitere Lösung: ◀ ' ALPHA ← T ENTER → CAT ALPHA Q (Benutzen Sie dann die Pfeiltasten ▲▼, um den Befehl QUAD auszuwählen) drücken Sie anschließend OK . In der Anzeige bekommen Sie die nachfolgenden zwei Lösungen:

Weitere erwähnenswerte Flags

Nehmen wir nochmals die aktuelle Flag Einstellung, indem wir die Taste MODE drücken und anschließend die Funktionstaste FLAGS. Stellen Sie sicher, dass Sie das Systemflag 01, welches in einer früheren Übung gesetzt wurde, bereinigen. Verwenden Sie die Pfeiltasten (▲ ▼), um in der Systemflag Liste zu navigieren.

Einige interessante Flags und deren bevorzugter Wert für die Übungen aus dieser Anleitung sind:

02 Constant → symb: Konstante Werte (z. B. π) werden als Symbole beibehalten

03 Function → symb: Funktionen werden nicht automatisch ausgewertet, statt dessen werden diese als symbolische Ausdrücke geladen.

*27 'X+Y*i' → (X,Y)* : Komplexe Zahlen werden als geordnete Paare dargestellt

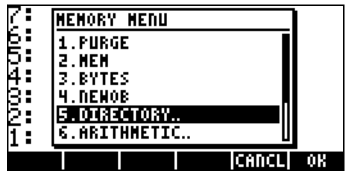
60 [α][α] Datensperren: Die Tastenfolge ALPHA ALPHA stellt die alphabetische Tastatur fest. Drücken Sie OK zweimal, um zur Normalanzeige des Taschenrechners zurückzukehren.

CHOOSE boxes vs. Funktions-MENU (Funktionsmenü)

In einigen Beispielen dieses Kapitels, konnten wir Menü-Befehlslisten auf dem Display angezeigt, sehen. Diese Menülisten werden als *CHOOSE* boxes bezeichnet. Um z. B. mit dem Befehl ORDER die Variablen, die wir verwendet haben, in einem Verzeichnis neu anzuordnen:

← PRG ▼ Zeige Menüliste PROG und wähle MEMORY

OK ▼ ▼ ▼ ▼ Zeige Menüliste MEMORY und wähle DIRECTORY

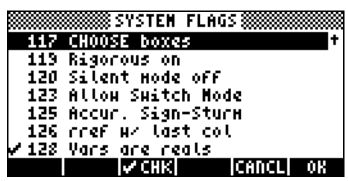


OK ▲ ▲ Zeige Menüliste DIRECTORY und wähle ORDER

OK starten Sie den Befehl ORDER. Es gibt eine Alternative diese Menüs über die Funktionstasten zu erreichen, und zwar durch setzen des Systemflags 117. Um dieses Flag zu setzen versuchen Sie Folgendes:

MODE FLAGS ▲ ▲ ▲ ▲ ▲ ▲ ▲

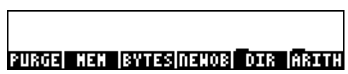
In der Anzeige erscheint Flag 117 nicht gesetzt (CHOOSE boxes), wie nachfolgend zu sehen ist:



Drücken Sie die Funktionstaste ✓CHK um Flag 117 im Funktions-MENU zu setzen. In der Anzeige wird diese Änderung angezeigt:

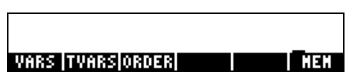
Drücken Sie zweimal, um zur Normalanzeige des Taschenrechners zurückzukehren.
Nun werden wir versuchen den Befehl ORDER mit ähnlicher Tastenfolge wie oben zu finden, d. h. wir starten mit ← PRG .
Beachten Sie, dass wir in diesem Fall anstelle einer Menüliste, Funktionstasten für das Menü mit den verschiedenen Optionen für das Menü PROG erhalten, d. h.

Drücken Sie F2 , um das Funktionsmenü MEMORY (MEM) auszuwählen. In der Anzeige erscheint nun:



Drücken Sie F5 , um das Funktionsmenü DIRECTORY (DIR) auszuwählen.

Der Befehl ORDER wird nicht angezeigt. Um diesen anzuzeigen, benutzen wir die Taste NXT :



Um den Befehl ORDER zu starten drücken wir die Funktionstaste F3 (ORDER). Obwohl nicht auf ein bestimmtes Beispiel angewendet, zeigt diese Übung zwei Optionen für Menüs im Taschenrechner an (CHOOSE boxes und Funktions-MENUs = Funktionsmenüs).

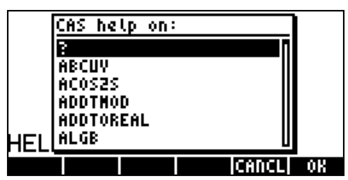
Ausgewählte CHOOSE boxes

Einige Menüs erzeugen nur CHOOSE boxes, so z. B.

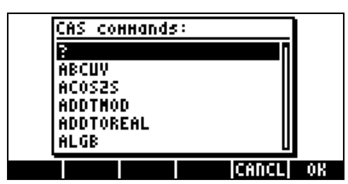
- Das Menü APPS (APPlicationS – Anwendungen), gestartet mit der Taste APPS, erste Taste in der zweiten Reihe von oben:

- Das Menü CAT (CATalog – Katalog), gestartet mit der Taste ↱ CAT , zweite Taste in der vierten Reihe von oben:

- Das Menü HELP, gestartet mit TOOL NXT HELP



- Das Menü CMDS (CoMmanDS – Befehle), welches innerhalb des EquationWriters mit der Tastenfolge ↱ EQW NXT CMDS aktiviert wird



Kapitel 3
Berechnung mit reellen Zahlen

In diesem Kapitel wird die Verwendung des Taschenrechners für Operationen und Funktionen in Zusammenhang mit reellen Zahlen erläutert. Die hier aufgeführten Operationen werden in den meisten Berechnungen in den Bereichen Physik und Technik angewendet. Der Benutzer sollte über Kenntnisse in Zusammenhang mit der Verwendung der Tastatur verfügen, um bestimmte auf der Tastatur befindliche Funktionen aufrufen zu können (z. B. SIN, COS, TAN usw.). Es wird davon ausgegangen, dass dem Benutzer die Bedienungsweise des Taschenrechners bekannt ist, d. h. das Auswählen des Betriebsmodus (siehe Kapitel 1) sowie die Verwendung von Menüs und CHOOSE Boxes (Kapitel 1), und dass er über Erfahrungen im Umgang mit Variablen verfügt (Kapitel 2).

Überprüfen der Einstellungen des Taschenrechners

Um die aktuellen CAS-Einstellungen des Taschenrechners zu überprüfen, beachten Sie die oberste Zeile im Display des Taschenrechners im Normalbetrieb. So könnten Sie z. B. die folgenden Einstellungen sehen: RAD XYZ DEC **R** = 'X'

RADiane steht für winkelförmige Maße, XYZ für rechtwinklige (Kartesische) Koordinaten, DECimal Zahlenbasis, **R**eelle Zahlen bevorzugt = "genaue" Ergebnisse, und 'X' ist der Wert der unabhängigen Standardvariablen.

Eine weitere Möglichkeit, diese Optionen anzuzeigen, lautet

DEG R∠Z HEX C ~ 't'

Hier steht DEGrees (Grade) für Winkelmaß, R∠Z für Polar-Koordinaten, HEXadezimalzahlen, komplexe Zahlen erlaubt, ~ steht für "ungefähre" Ergebnisse und 't' für die unabhängige Standardvariable.

Im Allgemeinen sind in diesem Bereich des Displays sieben Elemente enthalten. Jedes Element wird durch eine Zahl zwischen 1 und 7 gekennzeichnet. Die möglichen Werte für die Elemente werden in Klammern hinter der

Beschreibung des Elementes angegeben. Darüber hinauswird die Erklärung für jeden einzelnen dieser Werte angezeigt:

1. Spezifikation des Winkelmaßes (DEG, RAD, GRD)
   DEG: Grad, 360 Grad bilden einen vollständigen Kreis
   RAD: Radian, 2π Radiane bilden einen vollständigen Kreis
   GRD: Zentesimalgrad, 400 Zentesimalgrad bilden einen vollständigen Kreis
2. Spezifikationen des Koordinatensystems (XYZ, R∠Z, R∠∠). Das Symbol ∠ steht für Winkelkoordinaten
   XYZ: Kartesisch oder rechtwinklig (x,y,z)
   R∠Z: zylindrische Polarkoordinaten (r,θ,z)
   R∠∠: sphärische Koordinaten (ρ,θ,ϕ)
3. Zahlenbasen Spezifikation (HEX, DEC, OCT, BIN)
   HEX: Hexadezimalzahlen (Basis 16)
   DEC: Dezimalzahlen (Basis 10)
   OCT: Oktalzahlen (Basis 8)
   BIN: Binärzahlen (Basis 2)
4. Reeller oder komplexer Modus (**R, C**)
   R: reelle Zahlen
   C: komplexe Zahlen
5. Realmodus oder Annäherungsmodus (=,~)
   = exakter (symbolischer) Modus
   ~ Annäherungsmodus (numerischer)
6. unabhängige Standard CAS-Variable (z. B. 'X', 't' usw.)

Überprüfen des Taschenrechnermodus

Im RPN-Modus werden die verschiedenen Stack-Ebenen auf der linken Seite des Displays angezeigt. Im ALGEBRAISCHEN Modus gibt es keine durchnumerierten Stack-Ebenen, und das Wort ALG wird in der oberen Zeile des Displays rechts angezeigt. Der Unterschied zwischen diesen beiden Operationsmodi wurde detailliert in Kapitel 1 beschrieben.

Berechnungen mit reellen Zahlen

Beim Durchführen von Berechnungen mit reellen Zahlen wird empfohlen, das CAS auf *Real* (im Gegensatz zum *Complex*-Modus) einzustellen. In einigen Fällen könnten Sie ein komplexes Ergebnis erhalten, wobei Sie dann vom

Taschenrechner aufgefordert werden, in den *Complex*-Modus zu wechseln. Der *Exact*-Modus ist der Standardmodus für die meisten Berechnungen. Sie sollten daher mit Ihren Berechnungen in diesem Modus starten. Sollte es erforderlich sein, in den *Approx*-Modus umzuschalten, werden Sie vom Taschenrechner dazu aufgefordert. Es gibt keine bevorzugte Auswahl für das Winkelmaß oder für die Zahlenbasisspezifikation. Berechnungen mit reellen Zahlen werden sowohl im algebraischen (ALG) Modus als auch im Reverse Polish Notation- (RPN) Modus dargestellt.

Änderung des Vorzeichens einer Zahl, einer Variablen oder eines Ausdrucks

Drücken Sie die Taste +/-. Im ALG-Modus können Sie, bevor Sie eine Zahl eingeben, +/- drücken, z. B. +/- 2 • 5 ENTER. Das Ergebnis ist = -2,5. Im RPN-Modus müssen Sie vorerst mindestens einen Teil der Zahl eingeben und dann die Taste +/- verwenden, z. B. 2 • 5 +/-. Das Ergebnis ist = -2,5. Bei Verwenden der Funktion +/-, wenn keine Befehlszeile vorhanden ist, wendet der Taschenrechner die Funktion NEG (Umkehr des Vorzeichens) auf das Objekt in der ersten Stark-ebene an.

Die Umkehrfunktion

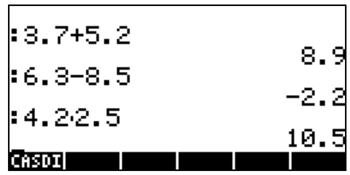
Verwenden Sie die Taste 1/x. Im ALG-Modus drücken Sie zuerst 1/x, gefolgt von einer Zahl oder einem algebraischen Ausdruck, z. B. 1/x 2. Das Ergebnis ist = ½ oder 0,5. Im RPN-Modus geben Sie zuerst die Zahl, dann die Funktion ein, z. B.: 4 ENTER 1/x. Das Ergebnis ist = ¼ oder 0,25.

Addition, Subtraktion, Multiplikation, Division

Verwenden Sie die reinen Operationstasten, also + − × ÷. Im ALG-Modus tippen Sie einen Operanden, dann einen Operator, dann wiederum einen Operanden, gefolgt von einem ENTER ein, um ein Ergebnis zu erzielen. Beispiele:

3 • 7 + 5 • 2 ENTER
6 • 3 − 8 • 5 ENTER
4 • 2 × 2 • 5 ENTER
2 • 3 ÷ 4 • 5 ENTER

Die ersten drei Operationen werden in obiger Anzeige dargestellt:



Im RPN-Modus geben Sie einen Operanden nach dem anderen, jeweils durch ein `ENTER` getrennt, ein. Anschließend drücken Sie die Taste für den Operator. Beispiele:

`3` `.` `7` `ENTER` `5` `.` `2` `+`
`6` `.` `3` `ENTER` `8` `.` `5` `−`
`4` `.` `2` `ENTER` `2` `.` `5` `×`
`2` `.` `3` `ENTER` `4` `.` `5` `÷`

Im RPN-Modus können Sie alternativ dazu die Operanden durch Leerzeichen (`SPC`) trennen, bevor Sie die Befehlstaste drücken. Beispiele:

`3` `.` `7` `SPC` `5` `.` `2` `+`
`6` `.` `3` `SPC` `8` `.` `5` `−`
`4` `.` `2` `SPC` `2` `.` `5` `×`
`2` `.` `3` `SPC` `4` `.` `5` `÷`

Verwendung von Klammern

Klammern können dazu verwendet werden, Operationen zu gruppieren, und darüber hinaus, um Funktions-Argumente einzuschließen. Klammern erhält man über die Tastenkombination `←` `()` . Klammern treten immer paarweise auf, so z. B. zur Berechnung von (5+3.2)/(7-2.2):

Im ALG-Modus:

`←` `()` `5` `+` `3` `.` `2` `▶` `÷` `←` `()` `7` `−` `2` `.` `2` `ENTER`

Im RPN-Modus benötigen Sie keine Klammern, die Berechnung erfolgt direkt im Stack:

`5` `ENTER` `3` `.` `2` `ENTER` `+` `7` `ENTER` `2` `.` `2` `ENTER` `−` `÷`

Wenn Sie im RPN-Modus den Ausdruck in Anführungszeichen schreiben, können Sie diesen wie im algebraischen Modus eingeben.

' ← () 5 + 3 . 2 ▶ ÷
← () 7 − 2 . 2 ENTER EVAL

In beiden Fällen, im ALG- wie auch im RPN-Modus, kann der EquationWriter dazu verwendet werden:

→ EQW 5 + 3 . 2 ▶ ÷ 7 − 2 . 2

Der Ausdruck kann innerhalb des EquationWriters ausgewertet werden, indem Sie nachfolgende Tastenfolgen verwenden

▲ ▲ ▲ ▲ EVAL oder → ▲ EVAL

Funktion Absoluter Wert

Die Funktion absoluter Wert, ABS, kann über die Tastenkombination ← ABS aufgerufen werden. Sollten Sie im Stack im ALG-Modus Berechnungen durchführen, müssen Sie die Funktion vor dem Argument eingeben, z. B. wie folgt: :

← ABS +/- 2 . 3 2 ENTER

Im RPN-Modus geben Sie zuerst die Zahl, dann die Funktion ein, z. B.:

2 . 3 2 +/- ← ABS

Quadrate und Quadratwurzeln

Die Quadratfunktion, SQ, kann über die Tastenkombination ← x^2 aufgerufen werden. Sollten Sie im Stack im ALG-Modus Berechnungen durchführen, müssen Sie die Funktion vor dem Argument eingeben, z. B. wie folgt: :

← x^2 +/- 2 . 3 ENTER

Im RPN-Modus geben Sie zuerst die Zahl, dann die Funktion ein, z. B.:

2 . 3 +/- ← x^2

Die Quadratwurzelfunktion, √, kann über die Taste R aufgerufen werden. Sollten Sie im Stack im ALG-Modus Berechnungen durchführen, müssen Sie die Funktion vor dem Argument eingeben, z. B. wie folgt: :

√x 1 2 3 . 4 ENTER

Im RPN-Modus geben Sie zuerst die Zahl, dann die Funktion ein, z. B.:

1 2 3 . 4 √x

Potenzen und Wurzeln

Die Potenzfunktion, ^, wird über die Taste Y^x aufgerufen. Wenn Sie im Stack im ALG-Modus berechnen, geben Sie die Base (y) gefolgt von der Taste Y^x und anschließend den Exponenten (x) ein, z. B.

5 . 2 Y^x 1 . 2 5

Im RPN-Modus geben Sie zuerst die Zahl, dann die Funktion ein, z. B.:

5 . 2 ENTER 1 . 2 5 ENTER Y^x

Die Wurzelfunktion XROOT*(y,x)* kann über die Tastenkombination → $\sqrt[x]{y}$ erreicht werden. Sollten Sie im Stack im ALG-Modus Berechnungen durchführen, müssen Sie die Funktion XROOT, gefolgt von den Argumenten (*y,x*), durch Komma getrennt, eingeben, z. B. wie folgt: :

→ $\sqrt[x]{y}$ 3 → , 2 7 ENTER

Im RPN-Modus müssen Sie zuerst das Argument *y*, dann *x* und schließlich die Funktion aufrufen, z. B. wie folgt: 2 7 ENTER 3 ENTER → $\sqrt[x]{y}$

Logarithmen mit der Basis 10 und Zehnerpotenzen

Die Berechnung von Logarithmen mit der Basis 10 werden über die Tastenkombination → LOG (Funktion LOG) berechnet, während die Umkehrfunktion (ALOG oder Algorithmus) über die Tastenkombination ← 10^x berechnet wird. Im ALG-Modus wird die Funktion vor dem Argument eingegeben:

→ LOG 2 . 4 5 ENTER
← 10^x +/- 2 . 3 ENTER

Im RPN-Modus wird das Argument vor der Funktion eingegeben:

2 . 4 5 ENTER ↱ LOG
2 . 3 +/- ENTER ↰ 10^x

Verwendung von Zehnerpotenzen bei der Dateneingabe

Zehnerpotenzen, d. h. Zahlen wie -4.5×10^{-2} usw., werden mithilfe der Taste EEX eingegeben. So z. B. im ALG-Modus:

+/- 4 . 5 EEX +/- 2 ENTER

Oder im RPN-Modus:

4 . 5 +/- EEX 2 +/- ENTER

Natürliche Logarithmen und Exponentialfunktionen

Natürliche Logarithmen (d. h. Logarithmen mit der Basis *e = 2,7182818282*) werden über die Tastenkombination ↱ LN (Funktion LN) berechnet, während ihre Umkehrfunktion, die Exponentialfunktion (Funktion EXP) unter Verwendung von ↰ e^x berechnet wird. Im ALG-Modus wird die Funktion vor dem Argument eingegeben:

↱ LN 2 . 4 5 ENTER
↰ e^x +/- 2 . 3 ENTER

Im RPN-Modus wird das Argument vor der Funktion eingegeben:

2 . 4 5 ENTER ↱ LN
2 . 3 +/- ENTER ↰ e^x

Trigonometrische Funktionen

Drei trigonometrische Funktionen sind bereits über die Tastatur abrufbar: Sinus (SIN), Cosinus (COS) und Tangens (TAN). Die Argumente dieser Funktion sind Winkel und können daher in jedes beliebige Winkelsystem (Grade, Radiane, Zentesimalgrade) eingegeben werden. So z. B. können nachfolgende trigonometrische Funktionen mit ausgewählter DEG (Grad) Funktion berechnet werden:

Im ALG-Modus:

SIN 3 0 ENTER
COS 4 5 ENTER
TAN 1 3 5 ENTER

Im RPN-Modus:

3 0 ENTER SIN
4 5 ENTER COS
1 3 5 ENTER TAN

Inverse trigonometrische Funktionen

Die über die Tastatur zur Verfügung stehenden inversen trigonometrischen Funktionen lauten Arcsinus (ASIN), Arccosinus (ACOS) und Arctangens (ATAN) und können über die jeweiligen Tastenkombinationen ← ASIN , ← ACOS und ← ATAN aufgerufen werden. Da die Inversen der trigonometrischen Funktionen Winkel darstellen, werden die Ergebnisse in den ausgewählten Winkelmaßen (DEG, RAD, GRD) ausgegeben. Nachfolgend einige Beispiele:

Im ALG-Modus:

← ASIN 0 • 2 5 ENTER
← ACOS 0 • 8 5 ENTER
← ATAN 1 • 3 5 ENTER

Im RPN-Modus:

0 • 2 5 ENTER ← ASIN
0 • 8 5 ENTER ← ACOS
1 • 3 5 ENTER ← ATAN

Alle oben aufgeführten Funktionen (ABS, SQ, √, ^, XROOT, LOG, ALOG, LN, EXP, SIN, COS, TAN, ASIN, ACOS, ATAN) können mit den Grundrechenarten (+ − × ÷) zur Bildung komplexerer Ausdrücke kombiniert werden. Der EquationWriter, dessen Funktionsweise in Kapitel 2 beschrieben wurde, ist für derartige Ausdrücke optimal geeignet, ungeachtet des Modus, auf den der Taschenrechner eingestellt ist.

Unterschied zwischen Funktionen und Operatoren

Funktionen wie ABS, SQ, √, LOG, ALOG, LN, EXP, SIN, COS, TAN, ASIN, ACOS und ATAN erfordern ein einziges Argument. Ihre Anwendung im ALG-Modus ist daher recht einfach, z. B. ABS(x). Einige Funktionen, wie beispielsweise XROOT, benötigen zwei Argumente, z. B. XROOT(x,y). Diese Funktion besitzt die entsprechende Tastenfolge → $\sqrt[x]{y}$.

Operatoren hingegen werden nach einem einzigen Argument oder zwischen zwei Argumenten eingesetzt. Der faktorielle Operator (!) z. B. wird nach einer Zahl eingesetzt, z. B. 5 ALPHA ↱ 2 ENTER. Da dieser Operator lediglich ein einziges Argument benötigt, wird er als monadisch bezeichnet. Operatoren, welche zwei Argumente benötigen, wie z. B. + − × ÷ y^x, werden als Binäroperatoren bezeichnet, z. B. 3 × 5 oder 4 y^x 2.

Funktionen von reellen Zahlen im Menü MTH

Das Menü MTH (MaTHematics) beinhaltet eine Reihe von mathematischen Funktionen, die hauptsächlich auf reelle Zahlen angewandt werden. Um in dieses Menü zu gelangen, drücken Sie die Tastenkombination ↰ MTH . Mit der Standardeinstellung auf CHOOSE boxes für Systemflag 117 (siehe Kapitel 2) wird das MTH-Menü als Menüliste dargestellt:

Da im Taschenrechner eine Vielzahl mathematischer Funktionen vorhanden sind, wurde das MTH-Menü nach Objekttyp der Funktion, dem diese zugeordnet sind, sortiert. So z. B. sind die Optionen 1. *VECTOR.*, *2. MATRIX.* und *3. LIST.* eben denselben Datentypen zugeordnet (d. h. Vektoren, Matrizen und Listen), die detailliert in einem späteren Kapitel erörtert werden. Die Optionen *4. HYPERBOLIC..* und *5. REAL..* sind reellen Zahlen zugeordnet und werden in diesem Kapitel erläutert. Option *6. BASE..* wird für Umrechnungen von Zahlen in unterschiedlichen Basen verwendet und werden ebenfalls in einem späteren Kapitel erörtert. Option *7. PROBABILITY..* wird für Wahrscheinlichkeits-Anwendungen eingesetzt und in einem der nächsten Kapitel erörtert. Option *8. FFT..* (Fast Fourier Transform) ist eine Anwendung zur Signalbearbeitung und wird in einem anderen Kapitel erörtert. Option *9. COMPLEX..* enthält Funktionen für komplexe Zahlen, welche im nächsten Kapitel erörtert werden. Option *10. CONSTANTS* ermöglicht den Zugriff auf die Konstanten im Taschenrechner. Diese Option wird weiter unten in diesem

Abschnitt erörtert. Schließlich die Option *11. SPECIAL FUNCTIONS..*, ´die Funktionen für höhere Mathematik einschließt, die ebenfalls in diesem Abschnitt erörtert werden.

Im Allgemeinen sollten Sie, um eine dieser Funktionen anzuwenden, Anzahl und Anordnung der für die einzelnen Funktionen erforderlichen Argumente beachten und sich stets vergegenwärtigen, dass im ALG Modus immer zuerst die Funktion und *dann* das Argument eingegeben wird, während im RPN Modus erst das Argument in den Stack eingegeben und anschließend die Funktion ausgewählt wird.

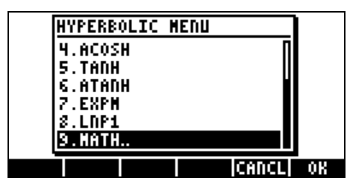
Verwendung der Rechenmenüs:

1. Da die Funktionsweise von MTH-Funktionen (und vieler anderer Taschenrechnermenüs) sehr ähnlich sind, werden wir nur die Funktionsweise der Menü-Option *4. HYPERBOLIC..* in diesem Abschnitt beschreiben, mit der Absicht, eine generelle Funktionsweise der einzelnen Taschenrechnermenüs aufzuzeigen. Achten Sie insbesondere auf die Vorgehensweise bei der Auswahl verschiedener Optionen.

2. Für eine schnelle Auswahl der nummerierten Optionen der Menüliste (oder CHOOSE boxes) geben Sie einfach die entsprechende Nummer der gewünschten Option über die Tastatur ein. Um z. B. die Option *4. HYPERBOLIC..* im Menü MTH auszuwählen, drücken Sie einfach die *4*.

Hyperbolische Funktionen und deren Inverse

Um das hyperbolische Funktionsmenü aufzurufen, wählen Sie im MTH-Menü die Option *4. HYPERBOLIC..* und bestätigen anschließend mit der Taste OK



Die hyperbolischen Funktionen sind:

Hyperbolische Sinusfunktion, SINH, und deren Inverse ASINH oder $\sinh^{-1}$

Hyperbolische Cosinusfunktion, COSH, und deren Inverse ACOSH oder $\cosh^{-1}$
Hyperbolische Tangensfunktion, TANH, und deren Inverse ATANH oder $\tanh^{-1}$

Dieses Menü enthält zusätzlich die nachfolgenden Funktionen:

$$EXPM(x) = \exp(x) - 1,$$
$$LNP1(x) = \ln(x+1).$$

Schließlich Option *9. MATH,* welche den Anwender zurück in das Menü MTH versetzt.

So benötigen Sie z. B. zum Berechnen der Funktion *tanh(2,5)* im ALG-Modus folgende Tastenfolge:

← MTH	Auswahl *MTH* Menü
4 OK	Auswahl Menü *4. HYPERBOLIC..*
5 OK	Auswahl der Funktion *5. TANH*
2 . 5 ENTER	Berechnen von *tanh(2,5)*

In der Anzeige erscheint folgende Ausgabe:

Für die gleiche Kalkulation im RPN-Modus wird nachfolgende Tastenfolge benötigt:

2 . 5 ENTER	Geben Sie das Argument in den Stack ein
← MTH	Auswahl *MTH* Menü
4 OK	Auswahl Menü *4. HYPERBOLIC..*
5 OK	Auswahl der Funktion *5. TANH*

Die Lösung lautet:

Die aufgeführten Operationen setzen voraus, dass Sie die Standard-Einstellungen für Systemflag 117 (*CHOOSE* boxes) verwenden: Haben Sie die Einstellungen dieses Flags auf *SOFT menu* (siehe Kapitel 2) festgelegt, wird das MTH-Menü wie folgt angezeigt (linke Seite ALG-Modus, rechte Seite RPN-Modus):

Wenn Sie die Taste NXT drücken, werden die weiteren noch zur Verfügung stehenden Optionen angezeigt:

Anmerkung: Durch Drücken von ← PREV gelangen Sie zu den ersten Optionen des Menüs *MTH* zurück. Mit der Tastenkombination → ▼ erhalten Sie eine Auflistung der Menüfunktionen in der Anzeige, z. B. wie folgt:

Um z. B. das hyperbolische Funktionsmenü aus diesem Menü aufzurufen, drücken Sie die Taste HYP, um nachfolgende Darstellung zu erhalten:

Um letztendlich die hyperbolische Funktion Tangens (tanh) zu erhalten, drücken Sie einfach die Taste TANH.

Anmerkung: Um zusätzliche Optionen dieses Funktionsmenüs anzuzeigen, drücken Sie entweder die Taste NXT oder die Tastenkombination ← PREV .

Um z. B. dieselbe Funktion *tanh(2,5)* im ALG-Modus über *CHOOSE* boxes zu berechnen, wenn SOFT-Menü aktiviert ist, gehen Sie wie folgt vor:

← MTH	Wählen Sie das *MTH*-Menü
HYP	Wählen Sie das Menü *HYPERBOLIC..*
TANH	Wählen Sie die Funktion *TANH*
2 . 5 ENTER	Berechnen von *tanh(2,5)*

Denselben Wert errechnen Sie im RPN-Modus über nachfolgende Tastenfolge:

2 . 5 ENTER	Geben Sie das Argument in den Stack ein
← MTH	Wählen Sie das *MTH*-Menü
HYP	Wählen Sie das Menü *HYPERBOLIC..*
TANH	Wählen Sie die Funktion *TANH*

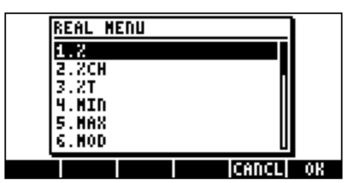
Als Beispiel für Anwendungen von hyperbolischen Funktionen überprüfen Sie die nachfolgenden Werte:

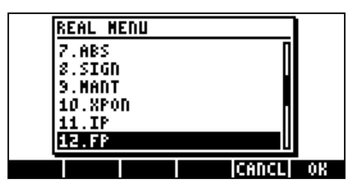
SINH (2,5) = 6,05020..	*ASINH (2,0) = 1,4436…*
COSH (2,5) = 6,13228..	*ACOSH (2,0) = 1,3169…*
TANH (2,5) = 0,98661..	*ATANH (0,2) = 0,2027…*
EXPM (2,0) = 6,38905….	*LNP1(1) = 0,69314….*

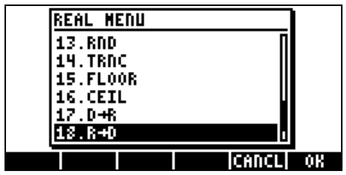
An dieser Stelle sollte noch einmal betont werden, dass die in diesem Abschnitt beschriebene allgemeine Verfahrensweise auf alle Auswahlfunktionen in jedem Menü des Taschenrechners anwendbar ist.

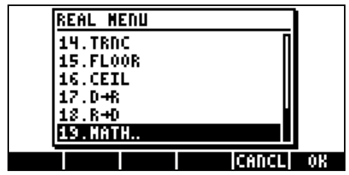
Funktionen zu reellen Zahlen

Bei Auswahl der Option *5. REAL..* aus dem Menü MTH mit auf *CHOOSE* boxes gesetztem Systemflag 117 wird folgende Menüliste erzeugt.



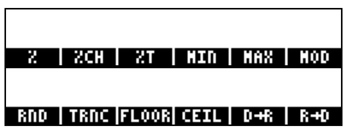


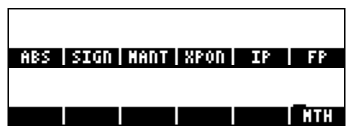




Option *19., MATH,* versetzt den Anwender zurück ins MTH-Menü. Die übrigen Funktionen sind in sechs verschiedene Gruppen zusammengefasst und werden nachfolgend beschrieben.

Wenn das Systemflag 117 auf SOFT-Menüs gesetzt ist, werden die Funktionen *REAL* im ALG-Modus wie folgt dargestellt (der verwendete Modus ist der ALG-Modus, dieselben Funktionstasten stehen jedoch auch im RPN-Modus zur Verfügung):





Die letzte Funktion, MTH, versetzt den Anwender zurück in das Menü *MTH*.

Prozentfunktionen

Diese Funktionen dienen der Prozentrechnung und gleichartiger Werte, wie im Folgenden dargestellt:

% (y,x) : berechnet der x Prozentsatz von y

%CH(y,x) : berechnet 100(y-x)/x, d. h. den Prozentsatz der Änderung bzw. der Differenz der beiden Zahlen.

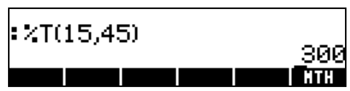
%T(y,x) : berechnet 100 x/y, d. h. den Gesamtprozentsatz bzw.den Prozentsatz für das Verhältnis einer Zahl (x) zu einer anderen Zahl (y).

Diese Funktionen benötigen zwei Argumente, wir veranschaulichen nachfolgend die Berechnung *%T(15,45)*, d.h., Berechnung 15% von 45. Nehmen wir an, der Rechner ist im ALG Modus und Systemflag 117 auf *CHOOSE* boxes gesetzt. Das Verfahren sieht wie folgt aus:

← MTH	Wählen Sie das Menü *MTH*.
5 OK	Wählen Sie Menü *5. REAL*...

3 OK	Wählen Sie die 5, Funktion *%T* .
1 5	Geben Sie das erste Argument ein.
→ ,	Geben Sie ein Komma ein, um die Argumente voneinander zu trennen.
4 5	Geben Sie das zweite Argument ein.
ENTER	Berechnen Sie die Funktion.

Nachfolgend das Ergebnis:



Im RPN-Modus befindet sich Argument *y* in der zweiten Stack-Ebene, während sich Argument *x* in der ersten Stack-Ebene befindet. Dies bedeutet, dass Sie *x* vor *y* nur im ALG-Modus eingeben sollten. Somit erfolgt die Berechnung von *%T(15,45)* im RPN-Modus, mit auf *CHOOSE* boxes gesetztem Systemflag 117, wie folgt:

1 5 ENTER	Geben Sie das erste Argument ein.
4 5 ENTER	Geben Sie das zweite Argument ein.
← MTH	Wählen Sie das Menü *MTH* .
5 OK	Wählen Sie Menü *5. REAL..*
3 OK	Wählen Sie die 5, Funktion *%T*.

Anmerkung: Die Beispiele in diesem Abschnitt veranschaulichen im Allgemeinen den Einsatz von Funktionen mit zwei Argumenten. Funktionen mit 3 oder mehr Argumenten können aus diesen Beispielen generalisiert werden.

Als Beispiel für Anwendungen von prozentbezogenen Funktionen überprüfen Sie die nachfolgenden Werte: *%(5,20) = 1, %CH(22,25) = 13.6363.., %T(500,20) = 4*

Minimum und Maximum

Verwenden Sie diese Funktionen, um den Minimal- oder Maximalwert von zwei Argumenten zu berechnen.

MIN(x,y) : Minimalwert von x und y
MAX(x,y) : Minimalwert von x und y

Als Beispiel überprüfen Sie, ob *MIN(-2,2) = -2* und *MAX(-2,2) = 2* ist.

Modulo:
MOD: y mod x = Rest von y/x, d. h., wenn x und y Integer-Zahlen sind, y/x = d + r/x, wobei d = Quotient und r = Rest darstellt, im diesem Fall r = y mod x.

Beachten Sie, dass MOD keine Funktion darstellt, sondern vielmehr einen Operator, d. h., dass im ALG Modus MOD als `y MOD x` und nicht als `MOD(y,x)` verwendet werden sollte. Somit ist die Vorgehensweise bei MOD ähnlich wie bei den Operatoren [+], [−], [×], [÷].

Als Beispiel überprüfen Sie, ob *15 MOD 4 = 15 mod 4 = Rest von 15/4 = 3* ist.

Absoluter Wert, Vorzeichen, Mantisse, Exponent, Integer und Brüche

ABS(x) : berechnet den absoluten Wert |x|.
SIGN(x): legt das Vorzeichen von x fest, d. h. -1, 0 oder 1.
MANT(x): bestimmt die Mantisse einer Zahl basierend auf $\log_{10}$.
XPON(x): bestimmt die Zehnerpotenz in der Zahl.
IP(x) : bestimmt den Integerteil einer reellen Zahl.
FP(x) : bestimmt den Bruchteil einer reellen Zahl.

Als Übung überprüfen Sie, ob *ABS(-3) = |-3| = 3, SIGN(-5) = -1, MANT(2540) = 2,540, XPON(2540) = 3, IP(2,35) = 2, FP(2,35) = 0,35.*

Aufrunden, Abschneiden, nächst kleinere *(floor)* und nächst höhere (*ceiling)* Grenzzahl Funktion

RND(x,y) : rundet y auf x Dezimalstellen auf.
TRNC(x,y) : schneidet y auf x Dezimalstellen ab.
FLOOR(x) : größtmögliche Integer-Zahl, kleiner oder gleich x.
CEIL(x) : kleinstmögliche Integerzahl, größer oder gleich x.

Als Übung überprüfen sie, ob *RND(1,4567,2) = 1.46, TRNC(1,4567,2) = 1,45, FLOOR(2,3) = 2, CEIL(2,3) = 3* ist.

Radian-in-Grad- und Grad-in-Radian-Funktionen

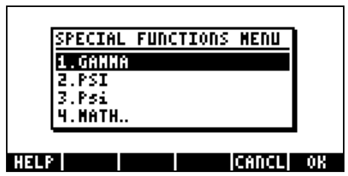
D→R (x) : konvertiert Grade in Radiane.

R→D (x) : konvertiert Radiane in Grade.

Als Übung überprüfen Sie, ob *D→R(45) = 0,78539* (d. h., 45° = $0{,}78539^{rad}$), *R→D(1,5) = 85,943669..* (d. h., $1{,}5^{rad}$ = 85,943669..°) ist.

Sonderfunktionen

Option *11. Special functions…* (Sonderfunktionen) im MTH-Menü beinhaltet folgende Funktionen:



GAMMA: Die Gammafunktion $\Gamma(\alpha)$

PSI: N-te Ableitung der Digamma-Funktion

Psi: Digamma-Funktion, Ableitung des ln (Gamma)

Die Gamma-Funktion wird wie folgt definiert $\Gamma(\alpha) = \int_0^\infty x^{\alpha-1}e^{-x}dx$. Diese Funktion wird in der angewandten Mathematik für Wissenschaft und Technik sowie in Wahrscheinlichkeits– und Statistik-Berechnungen eingesetzt.

Faktorielle einer Zahl

Die Faktorielle einer positiven Integer-Zahl n wird als $n!=n\cdot(n-1)\cdot(n-2) \ldots 3\cdot 2\cdot 1$ mit $0! = 1$ definiert. Über die Tastenfolge [ALPHA] [↱] [2] ist die Funktion Faktorielle im Taschenrechner verfügbar. In beiden Modi, ALG und RPN, geben Sie zuerst die Zahl, gefolgt von der Tastenfolge [ALPHA] [↱] [2], ein. Beispiel: [5] [ALPHA] [↱] [2] [ENTER].

Die oben definierte Gamma-Funktion hat die Eigenschaft, dass Folgendes gilt:

$\Gamma(\alpha) = (\alpha-1)\ \Gamma(\alpha-1)$, für $\alpha > 1$.

Daher kann diese mit der Faktoriellen einer Zahl verglichen werden, d. h.

$\Gamma(\alpha) = (\alpha-1)!$,

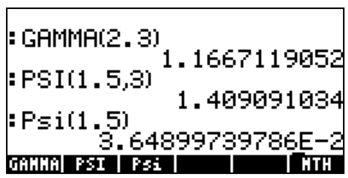
wenn α eine positive Integer-Zahl ist. Sie können die Funktion Faktorielle auch zur Berechnung der Gamma-Funktion und umgekehrt verwenden, z. B. $\Gamma(5) = 4!$ oder [4] [ALPHA] [↱] [2] [ENTER]. Die Funktion Faktorielle steht im MTH-Menü über das Menü *7. PROBABILITY..* (Wahrscheinlichkeit) zur Verfügung.

<u>Die Funktion PSI</u>, $\Psi(x,y)$ stellt die y-te Ableitung der Gamma-Funktion dar, d. h. $\Psi(n,x) = \frac{d^n}{dx^n}\psi(x)$, wobei $\psi(x)$ als die Digamma-Funktion oder Psi-Funktion bekannt ist. Für diese Funktion muss *y* eine positive Integer-Zahl sein.

<u>Die Funktion Psi</u>, $\psi(x)$ oder Digamma-Funktion, wird als $\psi(x) = \ln[\Gamma(x)]$ definiert.

Beispiele dieser Sonderfunktionen werden sowohl im ALG- wie auch im PRN-Modus gezeigt. Als Beispiel überprüfen Sie, ob *GAMMA(2,3) = 1,166711…*, *PSI(1,5,3) = 1,40909.* und *Psi(1,5) = 3,64899739..E-2* ist.

Diese Berechnungen werden in den nachfolgenden Screenshots dargestellt:



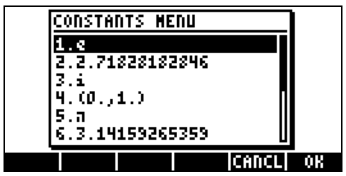
Konstanten des Taschenrechners

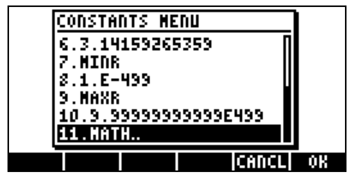
Nachfolgend die von Ihrem Taschenrechner verwendeten mathematischen Konstanten:

- *e*: die Basis eines natürlichen Logarithmus.
- *i*: die imaginäre Einheit $i^2 = -1$.
- π: das Verhältnis zwischen Länge eines Kreises und seinem Durchmesser.
- MINR: die im Taschenrechner zur Verfügung stehende kleinste reelle Zahl.
- MAXR: die im Taschenrechner zur Verfügung stehende größte reelle Zahl.

Um Zugriff auf diese Konstanten zu erhalten, wählen Sie Option *11. CONSTANTS.. (Konstanten)* im Menü MTH.

Die Konstanten werden wie folgt aufgelistet:





Durch Auswahl einer dieser Einträge wird der ausgewählte Wert, entweder ein Symbol (z. B. *e*, *i*, π, *MINR*, oder *MAXR)* oder ein Wert (*2,71.., (0,1), 3,14.., 1E-499, 9,99..E499*), in den Stack ausgegeben.

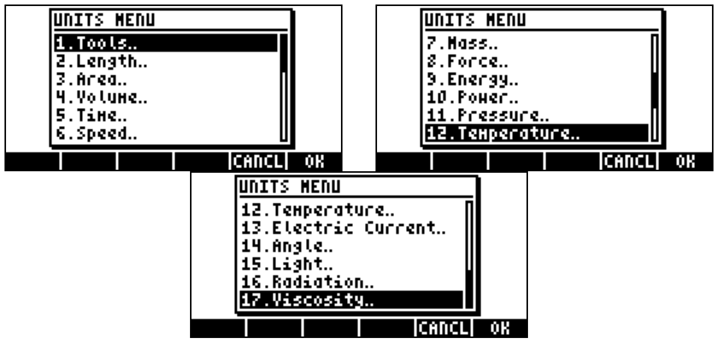
Beachten Sie, dass *e* über die Tastatur als *exp(1)* zur Verfügung steht, d. h. ← e^x 1 ENTER im ALG-Modus oder 1 ENTER ← e^x im RPN-Modus. Auch π ist direkt über die Tastatur verfügbar als ← π . Schließlich ist auch *i* über die Tastatur verfügbar (über die Taste ← *i*).

Operationen mit Einheiten

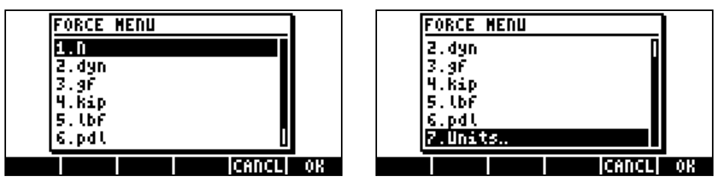
Zahlen im Taschenrechner können unterschiedlichen Einheiten zugeordnet sein. So können Sie Ergebnisse über ein konsistentes Einheiten-System berechnen und die Ergebnisse mit der geeigneten Kombination von Einheiten ausgeben lassen.

Das UNITS-Menü

Das UNITS-Menü wird über die Tastenkombination → UNITS (der Taste 6 zugeordnet) gestartet. Mit auf *CHOOSE* boxes gesetztem Systemflag 117 wird das nachfolgende Menü angezeigt:

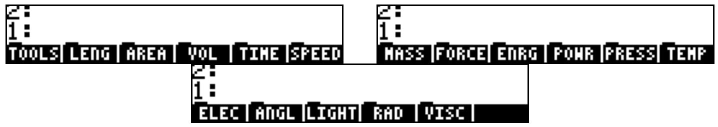


Option *1. Tools..* enthält Funktionen, welche sich auf Einheiten beziehen (zu einem späteren Zeitpunkt diskutiert). Optionen *3. Length..* bis *17.Viscosity..* enthalten Menüs mit einer Reihe von Einheiten für jede der beschriebenen Mengen. Wenn Sie z. B. das Menü *8. Force..* (Kraft) auswählen, erhalten Sie das folgende Menü mit Einheiten:

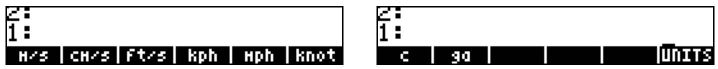


So wird der Benutzer die meisten der Einheiten (einige davon, wie z. B. dyne, werden heutzutage eher selten verwendet) aus dem Physikunterricht wieder erkennen: *N* = Newton, *dyn* = Dyn, *gf* = Gramm – Kraft (um Gramm-Masse oder einfach Gramm als Gewichtseinheiten zu unterscheiden), *kip* = Kilopond (1000 Pfund), *lbf* = Pound-Force (um von Pfund als Gewichtseinheit zu unterscheiden) und *pdl* = Poundal.

Um eine Einheit einer Zahl zuzuordnen, muss auf diese Zahl ein Unterstrich folgen. So wird die Kraft von 5 N als 5_N eingegeben.
Für umfassende Berechnungen mit Einheiten bieten die SOFT-Menüs eine bequemere Möglichkeit, Einheiten zuzuordnen. Ändern Sie das Systemflag 117 in SOFT-Menüs (siehe Kapitel 1), und verwenden Sie die Tastenkombination → *UNITS*, um in folgende Menüs zu gelangen. Drücken Sie die Taste NXT, um auf die nächste Seite des Menüs zu gelangen.

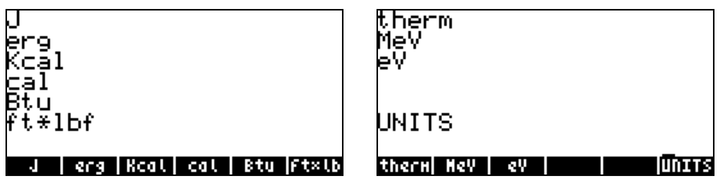


Wenn Sie die entsprechende Funktionstaste drücken, wird ein Untermenü mit Einheiten zu dieser Auswahl angezeigt. So z. B. stehen für das Untermenü SPEED folgende Einheiten zur Verfügung:



Durch erneutes Drücken der Funktionstaste UNITS gelangen Sie zum UNITS-Menü zurück.

Beachten Sie, dass Sie jederzeit die vollständige Liste der Menüeinträge durch Drücken der Tastenfolge ↱▼ anzeigen können. Es werden z. B. für das ENRG-Set von Einheiten nachfolgende Einträge angezeigt:



Anmerkung: Verwenden Sie die Taste NXT oder die Tastenkombination ↰ PREV zur Navigation zwischen den einzelnen Menüs.

Zur Verfügung stehende Einheiten

Nachfolgend finden Sie eine Liste von Einheiten, welche über das UNITS-Menü zur Verfügung stehen. Erst wird das Symbol der Einheit, gefolgt vom Namen der Einheit in Klammern, angezeigt:

LENGTH (LÄNGE)

m (Meter), cm (Zentimeter), mm (Millimeter), yd (Yard), ft (Fuß), in (Zoll), Mpc (Mega Parsec), pc (Parsec), lyr (Lichtjahr), au (astronomische Einheit), km (Kilometer), mi (internationale Meile), nmi (Seemeile), miUS (US gesetzliche englische Meile), chain (Kette), rd (Rute), fath (Kubikfuß), ftUS (Vermessungsfuß), Mil (Mil), μ (Mikron), Å (Angström), fermi (Fermi)

AREA (FÄCHE)

m^2 (Quadratmeter), cm^2 (Quadratzentimeter), b (Barn – Maßeinheit des Wirkungsquerschnittes), yd^2 (Quadratyard), ft^2 (Quadratfuß), in^2 (Quadratzoll), km^2 (Quadratkilometer), ha (Hektar), a (Are), mi^2 (Quadratmeile), miUS^2 (gesetzliche englische Quadratmeile), acre (Acre)

VOLUME (VOLUMEN)

m^3 (Kubikmeter), st (Ster), cm^3 (Kubikzentimeter), yd^3 (Kubikyard), ft^3 (Kubikfuß), in^3 (Kubikzoll), l (Liter), galUK (UK Gallone), galC (Kanadische Gallone), gal (US Gallone), qt (Quart), pt (Pint), ml (Milliliter), cu (US Tasse), ozfl (US fluid ounce), ozUK (UK fluid ounce), tbsp (Löffel), tsp (Teelöffel), bbl (Barrel), bu (Bushel - Trockenhohlmaß), pk (Peck - Trockenmaß), fbm (Board Foot - Holzmaß)

TIME (ZEIT)

yr (Jahr), d (Tag), h (Stunde), min (Minute), s (Sekunde), Hz (Hertz)

SPEED (GESCHWINDIGKEIT)

m/s (Meter pro Sekunde), cm/s (Zentimeter pro Sekunde), ft/s (Fuß pro Sekunde), kph (Kilometer pro Stunde), mph (Meilen pro Stunde), knot (Knoten – nautische Meilen pro Stunde), c (Lichtgeschwindigkeit), ga (Gravitationsbeschleunigung)

MASS (MASSE)

kg (Kilogramm), g (Gramm), Lb (avoirdupois Pound - Handelspfund), oz (Unze), slug (Slug, Gee-pound), lbt (Troy pound), ton (short ton), tonUK (long ton), t (metric ton), ozt (Troy Unze), ct (Karat), grain (Grain), u (atomare Masseeinheit), mol (Mol)

FORCE (KRAFT)

N (Newton), dyn (Dyn), gf (Gramm-Kraft), kip (Kilopond-Kraft), lbf (pound-force), pdl (Poundal)

ENERGY (ENERGIE)
J (Joule), erg (Erg), Kcal (Kilokalorie), Cal (Kalorie), Btu (englische Kalorie - Wärmemenge), ft×lbf (Foot-Pound), therm (EEC (GB) Wärmeeinheit zur Lieferung von Stadtgas), MeV (Megaelektronen Volt), eV (Elektronenvolt)

POWER (KRAFT)
W (Watt), hp (Pferdestärke)

PRESSURE (DRUCK)
Pa (Pascal), atm (Atmosphäre), bar (bar), psi (Pfund pro Quadratzoll), torr (Torr), mmHg (Millimeter Quecksilbersäule), inHg (Zoll Quecksilbersäule), inH20 (Zoll Wassersäule)

TEMPERATUR
° C (Grad Celsius), ° F (Grad Fahrenheit), K (Kelvin), ° R (Grad Rankine)

ELEKTRISCHER STROM (Elektrische Maßeinheiten)
V (Volt), A (Ampere), C (Coulomb), Ω (Ohm), F (Farad), W (Watt), Fdy (Faraday), H (Henry), mho (mho), S (Siemens), T (Tesla), Wb (Weber)

ANGLE (Maßeinheiten von Flächenwinkel und Raumwinkel)
° (sexagesimal Grad), r (Radian), Grad (Zentesimalgrad), arcmin (Bogenminute), arcs (Bogensekunde), sr (Steradiant)

LICHT (Maßeinheiten für das Licht)
fc (Footcandle), flam (Foot-Lambert), lx (Lux), ph (Phot), sb (Stilb), lm (Lumen), cd (Candela), lam (Lambert)

RADIATION (STRAHLUNG)
Gy (Gray), rad (Rad), rem (Rem), Sv (Sievert), Bq (Becquerel), Ci (Curie), R (Röntgen)

VISCOSITY (VISKOSITÄT)
P (Poise), St (Stokes)

Nicht aufgelistete Einheiten

Nicht aufgelistete Einheiten im UNITS-Menü, die dennoch im Taschenrechner vorhanden sind: gmol (Gramm-Mol), lbmol (Pound-Mol), rpm (Umdrehungen pro Minute), dB (Dezibel). Diese Maßeinheiten erreicht man über das Menü 117.02, welches im ALG-Modus über MENU (117.02) oder im RPN-Modus unter MENU 117.02 ENTER gestartet wird. In der Anzeige des Menüs erhalten Sie nachfolgende Einträge (verwenden Sie dazu die Tastenfolge → ▼, um die Beschriftung im Display anzuzeigen):

Auf diese Einheiten kann jedoch auch über den Katalog zugegriffen werden, z. B. folgendermaßen:

gmol:	→ _CAT ALPHA ← G
lbmol:	→ _CAT ALPHA ← L
rpm:	→ _CAT ALPHA ← R
dB:	→ _CAT ALPHA ← D

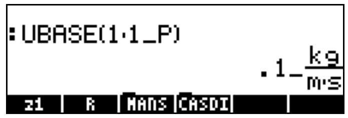
Umrechnung in Grundeinheiten

Verwenden Sie die Funktion UBASE, um jede dieser Einheiten in die Standardeinheiten des SI-Systems zu konvertieren. Um z. B. herauszufinden, welchen Wert 1 Poise (Viskositätseinheit) in SI-Einheiten darstellt, gehen Sie wie folgt vor:

Im ALG-Modus, wobei Systemflag 117 auf *CHOOSE* boxes gesetzt ist:

→ _UNITS	Wählen Sie das Menü UNITS.
OK	Wählen Sie das Menü TOOLS.
▼ OK	Wählen Sie die Funktion UBASE.
1 → _	Tragen Sie 1 und Unterstrich ein.
→ _UNITS	Wählen Sie das Menü UNITS.
▲ OK	Wählen Sie die Option VISCOSITY.
OK	Wählen Sie das Menü UNITS.
ENTER	Konvertieren Sie die Einheiten.

Als Ergebnis erhalten Sie die folgende Anzeige (d. h. *1 Poise = 0,1 kg/(m·s)*):



Im RPN-Modus, wobei Systemflag 117 auf *CHOOSE* boxes gesetzt ist:

1	Tragen Sie 1 (kein Unterstrich) ein.
↱ UNITS	Wählen Sie das Menü UNITS.
▲ OK	Wählen Sie die Option VISCOSITY.
OK	Wählen Sie die Einheit P (Poise.)
↱ UNITS	Wählen Sie das Menü UNITS.
OK	Wählen Sie das Menü TOOLS.
▼ OK	Wählen Sie die Funktion UBASEIm ALG-Modus.

Systemflag 117 ist auf *SOFT-Menüs* gesetzt.

↱ UNITS	Wählen Sie das Menü UNITS.
TOOLS	Wählen Sie das Menü TOOLS.
UBASE	Wählen Sie die Funktion UBASE.
1 ↱ _	Tragen Sie 1 und Unterstrich ein.
↱ UNITS	Wählen Sie das Menü UNITS.
↰ PREV VISC	Wählen Sie die Option VISCOSITY.
P	Wählen Sie die Einheit P (Poise).
ENTER	Konvertieren Sie die Einheiten.

Im RPN-Modus, wobei Systemflag 117 auf *SOFT- Menüs* gesetzt ist.

1	Tragen Sie 1 (kein Unterstrich) ein.
↱ UNITS	Wählen Sie das Menü UNITS.
↰ PREV VISC	Wählen Sie die Option VISCOSITY.
P	Wählen Sie die Einheit P (Poise).
↱ UNITS	Wählen Sie das Menü UNITS.
TOOLS	Wählen Sie das Menü TOOLS.
UBASE	Wählen Sie die Funktion UBASE.

Einheiten den Zahlen zuordnen

Um eine Einheit einer Zahl zuzuordnen, muss ein Unterstrich an diese Zahl angehängt werden (→ _ , Taste(8,5)). So wird die Kraft von 5 N als 5_N eingegeben.

Nachfolgend die Tastenfolge, die im ALG-Modus, mit auf *CHOOSE* boxes gesetztem Systemflag 117, eingegeben werden muss:

5 → _	Tragen Sie die Zahl und den Unterstrich ein.
→ UNITS	Begeben Sie sich in das Menü UNITS.
8 OK	Wählen Sie die Krafteinheiten (*8. Force..*).
OK	Wählen Sie Newton (*N*).
ENTER	Geben Sie die Anzahl der Einheiten in den Stack ein.

Die Anzeige wird wie folgt aus dargestellt:

Anmerkung: Vergessen Sie den Unterstrich, wird das Ergebnis als 5*N ausgegeben, wobei N eine mögliche Variable darstellt, nicht jedoch die Einheit Newton.

Verwenden Sie nachfolgende Tastenfolge, um dieselbe Eingabe im RPN-Modus vorzunehmen:

5	Geben Sie die Zahl ein (geben Sie keinen Unterstrich ein).
→ UNITS	Begeben Sie sich in das Menü UNITS.
8 OK	Wählen Sie die Krafteinheiten (*8. Force..*).
OK	Wählen Sie Newton (*N*).

Beachten Sie dabei, dass der Unterstrich automatisch eingefügt wird, wenn der RPN-Modus aktiviert ist. Das Ergebnis ist die nachfolgende Anzeige:

Wie zuvor angedeutet, wird, wenn das Systemflag 117 auf SOFT-Menüs steht, das Menü UNITS als Bezeichnung für die Funktionstasten angezeigt. Diese Einstellung erweist sich für umfassende Berechnungen als äußerst praktisch.

Nachfolgend die Tastenfolge zur Eingabe von Einheiten mit ausgewählter Option SOFT-Menü im ALG- und im PRN Modus: Um im ALG-Modus den Ausdruck 5_N einzugeben, verwenden Sie die Tastenfolge:

5 → _	Tragen Sie die Zahl und den Unterstrich ein.
→ UNITS	Begeben Sie sich in das Menü UNITS.
NXT FORCE	Wählen Sie die Einheiten für die Kraft.
N	Wählen Sie Newton (*N*).
ENTER	Geben Sie die Anzahl der Einheiten in den Stack ein.

Um denselben Ausdruck im RPN-Modus einzugeben, verwenden Sie die Tastenfolge:

5	Tragen Sie die Zahl (ohne Unterstrich) ein.
→ UNITS	Begeben Sie sich in das Menü UNITS.
NXT FORCE	Wählen Sie die Einheiten für die Kraft.
N	Wählen Sie Newton (*N*).

Anmerkung: Sie können einen Ausdruck mit Einheiten eingeben, indem Sie den Unterstrich und die Einheiten über die ALPHA–Taste eingeben, 5 → _ ALPHA N ergibt z. B. den folgenden Eintrag: 5_N.

Vorzeichen von Einheiten

Für Einheiten können Sie Vorzeichen gemäß der nachfolgenden Tabelle aus dem SI-System eingeben.
Als Erstes wird die Abkürzung des Vorzeichens aufgeführt, anschließend der Name, gefolgt vom Exponenten x im Faktor 10^x, welcher dem jeweiligen Vorzeichen entspricht:

Vorzeichen	Name	x	Vorzeichen	Name	x
Y	Yotta	+24	d	Deci	-1

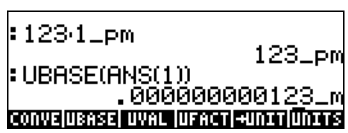
Z	Zetta	+21	c	Centi	-2
E	Exa	+18	m	Milli	-3
P	Peta	+15	µ	Micro	-6
T	Terra	+12	n	Nano	-9
G	Giga	+9	p	Pico	-12
M	Mega	+6	f	Femto	-15
k,K	Kilo	+3	a	Atto	-18
h,H	Hekto	+2	z	Zepto	-21
D(*)	Deka	+1	y	Yocto	-24

(*) Im SI-System wird höchstwahrscheinlich dieses Vorzeichen (*da*) anstelle von *D* stehen. Verwenden Sie dennoch das D für Deka im Taschenrechner.

Um diese Vorzeichen einzugeben, tippen Sie einfach das Vorzeichen über die ALPHA-Tastatur ein. Um z. B. 123 pm (1 Picometer) einzugeben, tippen Sie Folgendes ein:

1 2 3 ⮣ _ ALPHA ⮢ P ALPHA ⮢ M

Verwenden Sie UBASE, um in die Standard-Einheit (1 m) umzuwandeln; das Ergebnis wird wie folgt dargestellt:

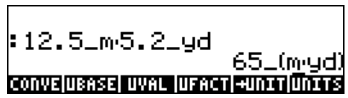


Operationen mit Einheiten

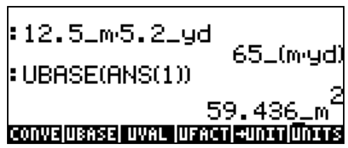
Sobald eine Menge, gefolgt von Einheiten, in den Stack eingegeben wurde, kann diese in Berechnungen, ähnlich reiner Zahlen, verwendet werden, ausgenommen, dass Mengen mit Einheiten nicht als Argumente in Funktionen eingesetzt werden können (z. B. SQ oder SIN). Bei dem Versuch, LN(10_m) zu berechnen, erhalten Sie eine Fehlermeldung: *Error: Bad Argument Type.*

Hier finden Sie einige Berechnungsbeispiele im ALG-Modus. Gehen Sie bei der Multiplikation und Division von Mengen mit Einheiten vorsichtig vor: Sie müssen jede Menge mit der dazugehörigen Einheit in Klammern einschließen.

Um z. B. das Produkt 12,5m × 5,2 yd einzugeben, muss Ihre Eingabe wie folgt aussehen (12,5_m)*(5,2_yd) ENTER:

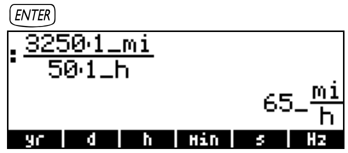


, welche dann als 65_(m·yd) angezeigt wird. Verwenden Sie die Funktion UBASE, um in die Einheiten des SI-Systems zu konvertieren:

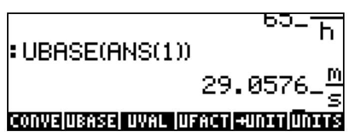


Anmerkung: Beachten Sie zu jedem Zeitpunkt, dass die Variable ANS(1) über die Tastenkombination ← ANS (der Taste ENTER zugeordnet) aufgerufen werden kann.

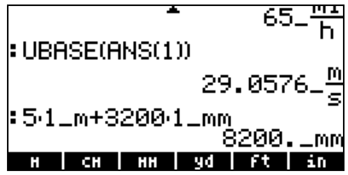
Um eine Division durchzuführen, z. B. 3250 mi / 50 h, geben Sie diese wie folgt ein: (3250_mi)/(50_h) ENTER



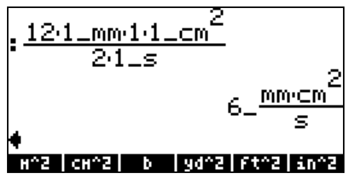
, welche dann mit der UBASE Funktion in SI-Einheiten konvertiert Folgendes ergibt:



Additionen und Subtraktionen können im ALG-Modus ohne Eingabe von Klammern durchgeführt werden. So kann z. B. 5 m + 3200 mm ganz einfach als 5_m + 3200_mm ENTER eingegeben werden.



Kompliziertere Ausdrücke hingegen benötigen Klammern, z. B. wie folgt:
(12_mm)*(1_cm^2)/(2_s) [ENTER] :



Bei Stack-Berechnungen im PRN-Modus werden keine Klammern bei der Eingabe unterschiedlicher Ausdrücke benötigt, z. B. wie folgt:

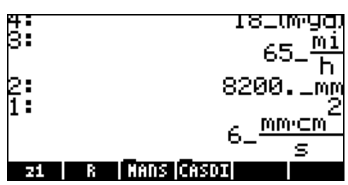
12_m [ENTER] 1,5_yd [ENTER] [×]
3250_mi [ENTER] 50_h [ENTER] [÷]

Diese Operationen ergeben folgende Ausgabe:

Testen Sie auch nachfolgende Operationen:

5_m [ENTER] 3200_mm [ENTER] [+]
12_mm [ENTER] 1_cm^2 [ENTER] [×] 2_s [ENTER] [÷]

Diese letzten beiden Operationen ergeben folgende Ausgabe:



Anmerkung: In Ausdrücke des EquationWriters dürfen keine Einheiten eingegeben werden.

Werkzeuge zur Manipulation von Einheiten

Das Menü UNITS enthält ein Untermenü TOOLS, welches folgende Funktionen zur Verfügung stellt:

CONVERT(x,y):	konvertiert Objekteinheit x in Einheiten des Objektes y.
UBASE(x):	konvertiert Objekteinheit x in SI-Einheiten.
UVAL(x):	extrahiert den Wert aus Objekteinheit x
UFACT(x,y):	multipliziert eine Einheit y aus der Objekteinheit x.
→UNIT(x,y):	kombiniert Werte von x mit Einheiten von y.

Die Funktion UBASE wurde im Detail in einem früheren Abschnitt dieses Kapitels beschrieben. Um auf eine dieser Funktionen zuzugreifen, gehen Sie wie in den vorangegangen gezeigten UBASE Beispielen vor. Beachten Sie, dass, während die Funktion UVAL ein einziges Argument benötigt, für die Funktionen CONVERT, UFACT und →UNIT jeweils zwei Argumente benötigt werden.

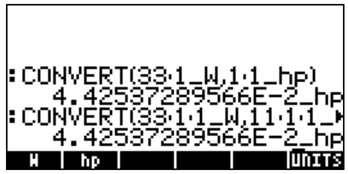
Probieren Sie die nachfolgenden Übungen in der von Ihnen bevorzugten Einstellung Ihres Taschenrechners durch. Die nachfolgende Ausgabe wurde im ALG-Modus, mit auf *SOFT*-Menü eingestelltem Systemflag 117, erzeugt:

CONVERT-Beispiele

Beide Beispiele ergeben das gleiche Ergebnis um 33 Watt in btus zu konvertieren:

CONVERT(33_W,1_hp) ENTER
CONVERT(33_W,1_hp) ENTER

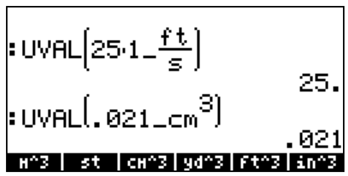
Diese Operationen werden in der Anzeige wie nachfolgend dargestellt:



UVAL-Beispiele:

UVAL(25_ft/s) `ENTER`

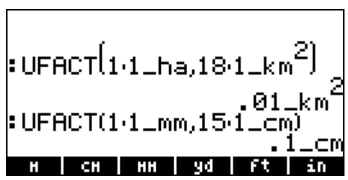
UVAL(0.021_cm^3) `ENTER`



UFACT-Beispiele:

UFACT(1_ha,18_km^2) `ENTER`

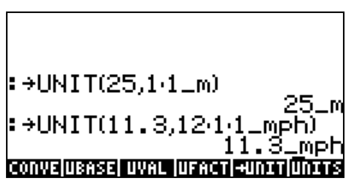
UFACT(1_mm,15,1_cm) `ENTER`



→UNIT-Beispiele

→UNIT(25,1_m) `ENTER`

→UNIT(11,3,1_mph) `ENTER`



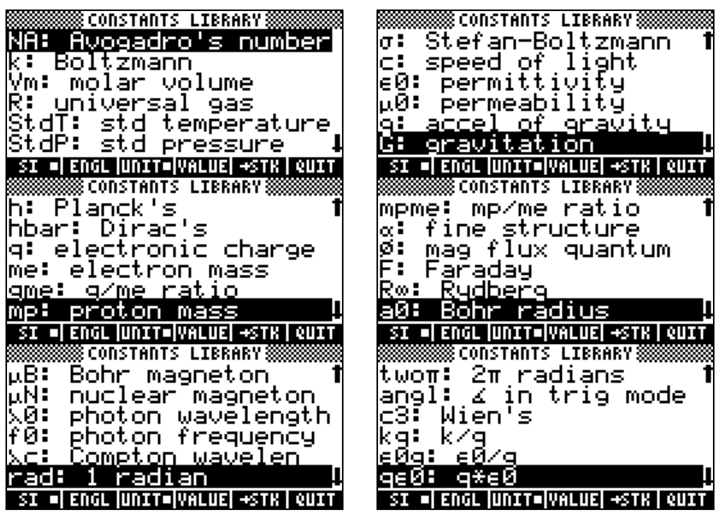
Physikalische Konstanten im Taschenrechner

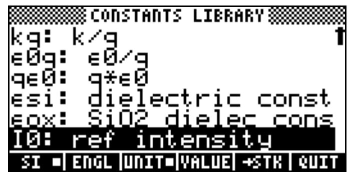
Analog zu der Behandlung von Einheiten erörtern wir ebenfalls die im Taschenrechner zur Verfügung stehenden physikalischen Konstanten. Die physikalischen Konstanten des Taschenrechners befinden sich in einer *constants library* (Konstantenbibliothek), welche mit dem Befehl CONLIB aufgerufen werden kann. Um diesen Befehl zu starten, müssten Sie lediglich folgende Eingabe im Stack vornehmen:

ALPHA ALPHA C O N L I B ALPHA ENTER

Alternativ wählen Sie den Befehl CONLIB aus dem Befehle Katalog wie folgt: Starten Sie zuerst den Katalog ↱ _CAT ALPHA C . Verwenden Sie dann die Pfeiltasten ▲▼, um CONLIB auszuwählen. Anschießend drücken Sie die Funktionstaste F6 (OK). Falls erforderlich, drücken Sie ENTER.

Die Anzeige der Konstantenbibliothek wird wie folgt dargestellt (verwenden die Pfeiltasten zur Navigation in der Bibliothek):



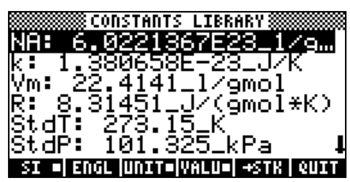


Die dieser Anzeige zugeordneten Funktionstasten der CONSTANTS LIBRARY enthalten folgende Funktionen:

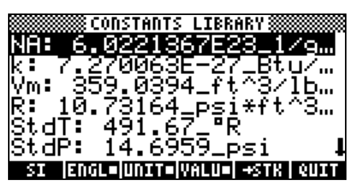
SI	wenn ausgewählt, werden die Werte der Konstanten in SI-Einheiten angezeigt.
ENGL	wenn ausgewählt, werden die Werte der Konstanten in Englischen-Einheiten angezeigt (*).
UNIT	wenn ausgewählt, werden die Konstanten zusammen mit den ihnen zugeordneten Einheiten ausgegeben (*).
VALUE	wenn ausgewählt, werden die Konstanten ohne Einheiten ausgegeben.
→STK	kopiert den Wert (mit oder ohne Einheiten) in den Stack.
QUIT	dient dem Verlassen der Konstantenbibliothek.

(*) Aktive nur, wenn die Funktion VALUE aktiv ist.

So wird die erste Seite der CONSTANTS LIBRARY-Anzeige dargestellt, wenn die Option VALUE ausgewählt wurde (Einheiten im SI-System):



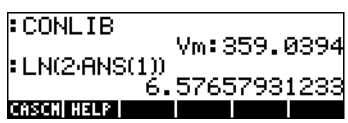
Um die Werte der Konstanten im englischen (bzw. Imperial-) System anzuzeigen, drücken Sie die Option ENGL :



Schalten Sie die UNITS-Option aus (deselektieren Sie sie durch Drücken der Taste UNITS), werden lediglich die Werte angezeigt (in diesem Fall wurden englische Einheiten ausgewählt):

Um den Wert Vm in den Stack zu kopieren, wählen Sie einen Variablen–Namen und drücken Sie erst die Taste →STK, dann QUIT. Ist der Taschenrechner auf ALG-Modus eingestellt, wird die Anzeige wie folgt dargestellt:

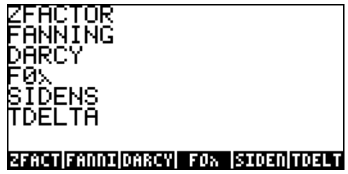
Die Anzeige zeigt einen so genannten *tagged value (*gekennzeichneten Wert) Vm:359,0394. In diesem Fall ist Vm die Kennzeichnung des Ergebnisses. Jede arithmetische Operation mit dieser Nummer ignoriert die Markierung. Probieren Sie z. B. Folgendes aus: → LN 2 × ← ANS ENTER, wodurch folgendes Resultat erzeugt wird:



Bei der gleichen Berechnung im RPN-Modus verwenden Sie folgende Tastenfolgen (nachdem der Wert von Vm aus der Konstantenbibliothek extrahiert wurde): 2 ENTER × → LN .

Spezielle physikalische Funktionen

Menu 117, ausgewählt über MENU (117) im ALG-Modus, oder MENU 117 ENTER im RPN-Modus, erzeugt das nachfolgende Menü (Beschriftungen werden über die Tasten → ▼ angezeigt):



Die Funktionen schließen ein:
ZFACTOR: Gaskompressibilität-Funktion Z Faktor
FANNING: Widerstandsfaktor der Strömungsauffächerung
DARCY: Darcy-Weisbach Widerstandsfaktor der Strömungsauffächerung
F0λ: Ausströmende Kraftfunktion Schwarzer Körper (Planckscher Strahler)
SIDENS: innere Dichte Silizium
TDELTA: Delta-Funktion Temperatur

Auf der zweiten Seite dieses Menüs (drücken Sie NXT) finden Sie nachfolgende Elemente:

Auf dieser Menüseite befinden sich eine Funktion (TINC) und eine Anzahl Maßeinheiten, die in einem früheren Abschnitt beschrieben wurden (siehe oben). Die für uns interessante Funktion ist:

TINC: Befehl Wertzuwachs (Inkrement) für Temperatur

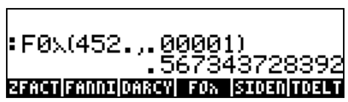
Von allen Funktionen in diesem Menü (UTILITY), d. h. ZFACTOR, FANNING, DARCY, F0λ, SIDENS, TDELTA und TINC, werden die Funktionen FANNING und DARCY in Kapitel 6 in Zusammenhang mit Gleichungen zur Durchflussberechnung in Rohrleitungen beschrieben. Für die verbleibenden Funktionen finden Sie nachfolgend eine kurze Beschreibung.

Funktion ZFACTOR

Die Funktion ZFACTOR berechnet den Berichtigungsfaktor für die Gaskompressibilität bei nicht idealem Verhalten von Kohlewasserstoffgas. Die Funktion wird über ZFACTOR(x_T, y_P) aufgerufen, wobei x_T die verringerte Temperatur, d. h. das Verhältnis der eigentlichen Temperatur zur pseudo-kritischen Temperatur ist und y_P der verringerte Druck, d. h. das Verhältnis des eigentlichen Drucks zum pseudo-kritischen Druck darstellt. Der Wert von x_T muss zwischen 1,05 und 3,0 liegen, während der Wert von y_P zwischen 0 und 30 liegen muss. Beispiel im ALG-Modus:

Funktion F0λ

Die Funktion F0λ (T, λ) berechnet den Bruch (dimensionslos) der gesamten ausströmenden Kraft der schwarzen Körper bei einer Temperatur T zwischen den Wellenlängen 0 und λ. Sind T und λ keine Einheiten zugewiesen, wird angenommen, dass T in K und λ in m enthalten ist. Beispiel im ALG-Modus:



Funktion SIDENS

Die Funktion SIDENS(T) berechnet die innere Dichte von Silizium (in Einheiten von $1/cm^3$) als Funktion der Temperatur T (T in K), für T zwischen 0 und 1685 K, beispielsweise folgendermaßen:

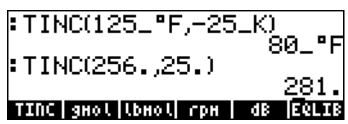
Funktion TDELTA

Die Funktion TDELTA(T_0,T_f) liefert den Temperaturzuwachs $T_f - T_0$. Das Ergebnis wird in der gleichen Maßeinheit, falls überhaupt, als T_0 ausgegeben. Andernfalls erhalten Sie einfach die Zahlendifferenz, beispielsweise folgendermaßen:

Der Zweck dieser Funktion besteht darin, eine Temperaturdifferenzberechnung bei unterschiedlichen Maßeinheiten der Temperatur zu vereinfachen. Andernfalls wird einfach nur die jeweilige Differenz der Zahlen berechnet. z. B.

Funktion TINC

Die Funktion TINC(T_0,ΔT) berechnet T_0+DT. Diese Funktion ist TDELTA insofern ähnlich, als das Ergebnis in der Maßeinheit T_0 ausgegeben wird. Andernfalls ist das Ergebnis eine einfache Addition dieser Werte, wie z. B.:



Definieren und Anwenden von Funktionen

Der Benutzer kann selbst eigene Funktionen definieren, indem er den Befehl DEF über die Tastenfolge ← DEF (der Taste 2 zugeordnet) aufruft. Die Funktion muss im nachfolgenden Format eingegeben werden:

Function_name(arguments) = expression_containing_arguments

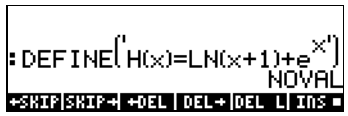
Als Beispiel könnte eine einfache Funktion definiert werden *H(x) = ln(x+1) + exp(-x)*.

Angenommen, Sie müssten diese Funktion für eine Zahl von diskreten Werten auswerten. Sie möchten daher hierfür nur eine einzige Taste verwenden müssen und das gewünschte Ergebnis, ohne umständliches separates Eintippen der einzelnen Werte, auf der rechten Seite anzeigen. Im nachfolgenden Beispiel wird davon ausgegangen, dass sich der Taschenrechner im ALG-Modus befindet. Geben Sie die nachfolgende Tastenfolge ein:

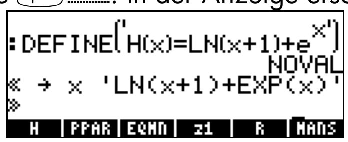
← DEF ' ALPHA H ← () ALPHA ← X ▶ → =

[↱] LN [ALPHA] [↰] [X] [+] [1] [▶] [+] [↰] e^x [ALPHA] [↰] [X] [ENTER]

Die Anzeige wird wie folgt dargestellt:



Drücken Sie die Taste [VAR]. Sie werden feststellen, dass sich eine neue Variable in Ihrer Funktionstaste ([H]) befindet. Um den Inhalt dieser Variablen anzuzeigen, drücken Sie [↱][H]. In der Anzeige erscheint nun Folgendes:



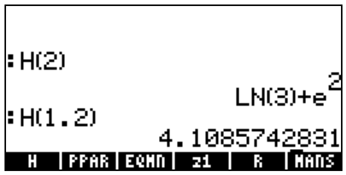
Somit enthält nun die Variable H ein Programm:

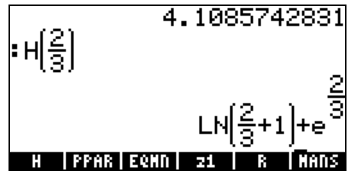
<< → x 'LN(x+1) + EXP(x)' >>

Hierbei handelt es sich um ein einfaches Programm in der Standard-Programmiersprache des HP 48, das darüber hinaus in die neue Reihe HP 49 G integriert wurde. Diese Programmiersprache wird als UserRPL bezeichnet. Das oben aufgeführte Programm ist relativ einfach und besteht aus zwei Bestandteilen, welche sich zwischen den Programm-Containern « » befinden:

- Eingabe: → x → x
- Prozess: 'LN(x+1) + EXP(x) '

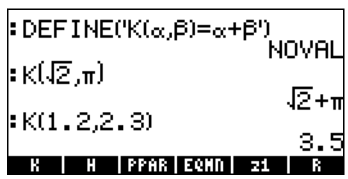
Dies wird so interpretiert: trage einen Wert ein, der temporär dem Namen x (als lokale Variable bezeichnet) zugeordnet wird, werte den Ausdruck zwischen den Anführungszeichen, die lokale Variablen enthalten, aus, und zeige dann den berechneten Ausdruck.
Um die Funktion im ALG-Modus zu starten, geben Sie den Namen der Funktion ein, gefolgt vom Argument in Klammern, z. B.
[H] [↰] () [2] [ENTER]. Nachfolgend einige Beispiele:





Im RPN-Modus müssen Sie zuerst das Argument eingeben und dann die Funktionstaste, welche der Variablen mit dem Namen H entspricht, drücken, bevor die Funktion gestartet wird. Sie könnten z. B. Folgendes ausprobieren: 2 ENTER H. Die weiteren oben aufgeführten Beispiele können wie folgt eingegeben werden: 1 . 2 ENTER H , 2 ÷ 3 ENTER H .

Funktionen können jedoch auch über mehr als zwei Argumente verfügen. So zeigt z. B. die untere Abbildung die Definition der Funktion $K(\alpha,\beta) = \alpha+\beta$ und ihrer Auswertung mit den Argumenten $K(\sqrt{2},\pi)$ und $K(1.2,2.3)$:



Die Inhalte der Variablen K sind: << → α β 'α+β' >>.

Funktionen die über mehr als einen Ausdruck definiert werden

In diesem Abschnitt behandeln wir Funktionen, die von zwei oder mehreren Ausdrücken definiert werden. Ein Beispiel einer solchen Funktionen ist:

$$f(x) = \begin{cases} 2 \cdot x - 1, & x < 0 \\ x^2 - 1, & x > 0 \end{cases}$$

Im Taschenrechner steht die Funktion IFTE (IF-Then-Else) zur Verfügung, welche derartige Funktionen beschreiben würde.

Die Funktion IFTE

Die IFTE-Funktion wird als IFTE (*condition*, *operation_if_true*, *operation_if_false*) geschrieben. Wenn die Bedingung wahr ist, wird die *Bedingung-wenn-wahr* ausgeführt, andernfalls *Bedingung-wenn-falsch*. So können wir z. B., um die obige Funktion zu beschreiben, 'f(x) = IFTE(x>0, x^2-1, 2*x-1)' schreiben. Die Funktion IFTE befindet sich im Befehle-Katalog (CAT). Das Symbol '>' (größer als) ist vorhanden als (der Taste 1/x zugeordnet). Verwenden Sie nachfolgende Befehlsfolge, um diese Funktion im ALG-Modus zu definieren: DEF(f(x) = IFTE(x>0, x^2-1, 2*x-1)).

Drücken Sie anschließend ENTER. Im RPN-Modus geben Sie die Definition der Funktion zwischen Apostrophen ein:

'f(x) = IFTE(x>0, x^2-1, 2*x-1)'

Drücken Sie dann DEF.

Drücken Sie VAR, um ins Variablen-Menü zurückzukehren. In Ihrem Funktionstastenmenü sollte die Variable f zur Verfügung stehen. Drücken Sie nun f, um das resultierende Programm anzuzeigen:

<< → x 'IFTE(x>0, x^2-1, 2*x-1)' >>

Um diese Funktion im ALG-Modus zu berechnen, geben Sie den Funktionsnamen, f, gefolgt von der Zahl ein, in der Sie den Ausdruck auswerten möchten, z. B. f(2), und drücken dann ENTER. Im RPN-Modus geben Sie eine Zahl ein und drücken dann f. Überprüfen Sie z. B., ob *f(2) = 3*, während *f(-2) = -5* ist.

Kombinierte IFTE-Funktionen

Um eine kompliziertere Funktion, wie die nachfolgende Funktion, zu programmieren

$$g(x) = \begin{cases} -x, & x < -2 \\ x+1, & -2 \le x < 0 \\ x-1, & 0 \le x < 2 \\ x^2, & x \ge 2 \end{cases}$$

, können Sie unterschiedliche Stufen der Funktion IFTE kombinieren, d. h.:

‘g(x) = IFTE(x<-2, -x, IFTE(x<0, x+1, IFTE(x<2, x-1, x^2)))’,

Definieren Sie die Funktion über eine der oben vorgestellten Möglichkeiten, und überprüfen Sie, ob g(-3) = 3, g(-1) = 0, g(1) = 0, g(3) = 9 ergibt.

Kapitel 4
Berechnungen mit komplexen Zahlen

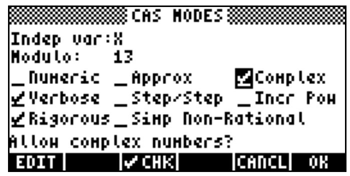
In diesem Kapitel finden Sie Beispiele von Berechnungen und Anwendungen von Funktionen mit komplexen Zahlen.

Definitionen

Eine *komplexe Zahl z* ist eine als *z = x + iy* geschriebene Zahl, wobei *x* und *y* reelle Zahlen sind und *i* die *imaginäre Einheit*, definiert durch $i^2 = -1$ darstellt. Die Zahl *x+iy* hat einen *reellen Teil x = Re(z)* und einen *imaginären Teil y = Im(z).* Wir können uns eine komplexe Zahl als ein Punkt *P(x,y)* in der x-y Ebene vorstellen, wobei die x-Achse als *reelle Achse* und die y-Achse als die *imaginäre Achse* bezeichnet wird. Somit wird eine komplexe Zahl in Form von *x+iy* als deren *Kartesische Darstellung* bezeichnet. Eine alternative Kartesische Darstellung ist das geordnete Paar *z = (x,y)*. Die *polare Darstellung* einer komplexen Zahl lautet $z = re^{i\theta} = r\cdot\cos\theta + i\ r\cdot\sin\theta$, ωοβει $r = |z| = \sqrt{x^2 + y^2}$ die *Magnitude* der komplexen Zahl z ist, und $\theta = Arg(z) = \arctan(y/x)$ das Argument der komplexen Zahl z darstellt. Das Verhältnis zwischen der Kartesischen und der polaren Darstellung einer komplexen Zahl ergibt sich aus der *Euler Formel:* $e^{i\theta} = \cos\theta + i\sin\theta$. Die *konjugiert komplexe Zahl* einer komplexen Zahl $z = x + iy = re^{i\theta}$, ist $\overline{z} = x - iy = re^{-i\theta}$. Die konjugiert komplexe Zahl von *i* kann als Spiegelbild von z auf der reellen *(x)* Achse betrachtet werden. Ähnlich kann die Negative von z, $-z = -x-iy = -re^{i\theta}$, als die Spiegelung von *z* zum Ursprung betrachtet werden.

Einstellen des Taschenrechners auf den COMPLEX-Modus

Bei der Arbeit mit komplexen Zahlen ist es von Vorteil, den Taschenrechner in dem Komplex-Modus umzustellen. Verwenden Sie dazu die Tastenfolge
MODE CAS 2 ▼ ▼ ▶ ✓CHK .
Der COMPLEX Modus wird in der CAS MODES Anzeige ausgewählt, indem die Option _*Complex* mit einem Häkchen versehen wird, d. h.



Drücken Sie OK zweimal, um zum Stack zurückzukehren.

Eingabe von komplexen Zahlen

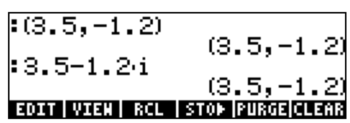
Komplexe Zahlen können in eine der beiden Kartesischen Darstellungsweisen in den Taschenrechner eingegeben werden, entweder *x+iy* oder *(x,y)*. Die Ergebnisse im Taschenrechner werden im Format geordneter Paare, d. h. als *(x,y)* angezeigt. Im ALG-Modus z. B. wird die komplexe Zahl *(3,5,-1.2)* wie folgt eingegeben:

← () 3 • 5 → , +/- 1 • 2 ENTER

Eine komplexe Zahl kann jedoch auch als *x+iy* eingegeben werden. Im ALG-Modus wird *3,5-1,2i* wie folgt eingegeben:

3 • 5 − 1 • 2 × ← i ENTER

Nach Eingabe dieser komplexen Zahlen erhalten Sie folgende Anzeige:



Im RPN-Modus können diese Zahlen über folgende Tastenfolge eingegeben werden:

← () 3 • 5 → , 1 • 2 +/- ENTER

(Beachten Sie, dass die Taste "Vorzeichen ändern" nach der Zahl 1,2 eingegeben wird, und zwar genau gegenteilig zum ALG-Modus-Beispiel) und

' 3 • 5 − 1 • 2 × → i ENTER

(Beachten Sie auch, dass Sie im RPN-Modus einen Apostroph vor der Zahl 3,5-1,2i eingeben müssen.) Im RPN-Modus sieht die Anzeige wie folgt aus:

Beachten Sie, dass die letzte Eingabe eine komplexe Zahl im Format *x+iy* ist, weil die Zahl zwischen Apostrophe eingegeben wurde, und somit einen algebraischen Ausdruck darstellt. Verwenden Sie die Taste EVAL (EVAL), um diese Zahl zu berechnen.

Sobald der algebraische Ausdruck berechnet wurde, stellen Sie die komplexe Zahl wieder her (3,5,1,2).

Polare Darstellung einer komplexen Zahl

Das obige Ergebnis zeigt eine Kartesische (rechtwinklige) Darstellung der komplexen Zahl 3,5-1,2i. Eine polare Darstellung ist möglich, wenn wir das Koordinatensystem über die Funktion CYLIN auf zylindrisch oder polar ändern. Sie finden diese Funktion im Katalog (→ CAT). Ändern wir auf polare Darstellung, erhalten wir folgendes Ergebnis:

Für dieses Ergebnis wurde das rechtwinklige Maß auf Radian umgestellt (Mithilfe der Funktion RAD können Sie jederzeit auf Radian umstellen). Das obige Ergebnis stellt eine Magnitude von 3,7 und einen Winkel von 0,33029… dar. Das Winkelsymbol (∠) wird vor dem Winkelmaß angezeigt.

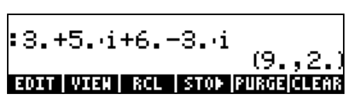
Zur Kartesischen Darstellung oder zu rechtwinkligen Koordinaten kommen Sie mit der Funktion RECT wieder zurück (zu finden im Katalog → CAT). Eine komplexe Zahl in polarer Darstellung wird als $z = r \cdot e^{i\theta}$ eingegeben. Die Eingabe dieser komplexen Zahl kann in den Taschenrechner über ein geordnetes Paar erfolgen, das wie folgt aussieht (r, ∠θ). Das Winkelsymbol (∠) kann als ALPHA ↱ 6 eingegeben werden. So kann z. B. die komplexe Zahl z = 5,2e$^{1.5i}$ wie folgt eingegeben werden (die Zahlen stellen den RPN-Stack vor und nach Eingabe der Zahl dar):

Da das Koordinatensystem auf rechtwinklige (oder Kartesische) Darstellung eingestellt ist, konvertiert der Taschenrechner die eingegebene Zahl in Kartesische Koordinaten, d. h. x = r cos θ, y = r sin θ, in diesem Fall (0,3678…, 5,18…).

Ist hingegen das Koordinatensystem (über die Funktion CYLIN) auf zylindrisch eingestellt, bekommen Sie eine polare Darstellung bei der Eingabe einer komplexen Zahl (x,y), wobei x und y reelle Zahlen sind. Geben Sie z. B. in zylindrische Koordinaten die Zahl (3.,2.) ein. Nachfolgend können Sie den RPN-Stack vor und nach Eingabe dieser Zahl sehen:

Einfache Operationen mit komplexen Zahlen

Komplexe Zahlen können über die vier Grundrechenarten (+ − × ÷) kombiniert werden. Die Ergebnisse befolgen die algebraischen Regeln nach der Formel, dass $i^2= -1$ ist. Operationen mit komplexen Zahlen entsprechen denen mit reellen Zahlen. Versuchen Sie, mit dem Taschenrechner im ALG-Modus und das CAS auf *Complex* eingestellt, folgende Zahlen einzugeben: (3+5i) + (6-3i):



Beachten Sie, dass die reellen Teile (3+6) mit den imaginären Teilen (5-3) kombiniert werden und das erhaltene Ergebnis ein geordnetes Zahlenpaar mit dem reellen Teil 9 und dem imaginären Teil 2 darstellt. Versuchen Sie nachfolgende Berechnungen selbst:

(5-2i) - (3+4i) = (2,-6)
(3-i)·(2-4i) = (2,-14)
(5-2i)/(3+4i) = (0.28,-1.04)
1/(3+4i) = (0.12, -0.16)

Anmerkungen:
Das Produkt zweier Zahlen wird wie nachfolgend dargestellt: $(x_1+iy_1)(x_2+iy_2) = (x_1x_2 - y_1y_2) + i\,(x_1y_2 + x_2y_1)$.
Die Division zweier komplexer Zahlen wird erreicht, wenn man sowohl den Zähler als auch den Nenner mit der konjugiert komplexen Zahl des Nenners multipliziert, d. h.

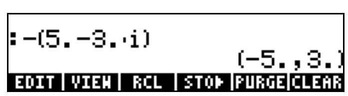
$$\frac{x_1+iy_1}{x_2+iy_2} = \frac{x_1+iy_1}{x_2+iy_2}\cdot\frac{x_2-iy_2}{x_2-iy_2} = \frac{x_1x_2+y_1y_2}{x_2^2+y_2^2} + i\cdot\frac{x_2y_1-x_1y_2}{x_2^2+y_2^2}$$

Somit ist die Inverse INV (wird mit der Taste 1/x aktiviert) definiert als:

$$\frac{1}{x+iy} = \frac{1}{x+iy}\cdot\frac{x-iy}{x-iy} = \frac{x}{x^2+y^2} + i\cdot\frac{y}{x^2+y^2}$$

Änderung des Vorzeichens einer komplexen Zahl

Das Vorzeichen einer komplexen Zahl kann mit der Taste +/- geändert werden, z. B. -(5-3i) = -5 + 3i



Eingabe der Einheit imaginäre Zahl

Um die Zahl des Typs imaginäre Einheit einzugeben, verwenden Sie ← i

Beachten Sie dabei, dass die Zahl *i* als geordnetes Zahlenpaar *(0,1)* eingegeben wird, wenn das CAS im APPROX-Modus steht. Im EXACT-Modus wir die Zahl des Typs imaginäre Einheit als *i* eingegeben.

Weitere Operationen

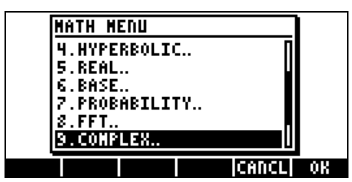
Operationen wie Magnitude, Argumente, reelle und imaginäre Teile, aber auch konjugiert komplexe Zahlen werden weiter unten innerhalb des Menüs CMPLX im Detail erläutert.

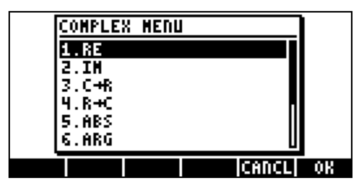
Die CMPLX-Menüs

Im Taschenrechner stehen zwei CMPLX (CoMPLeX – komplex) Menüs zur Verfügung. Eines steht über das Menü MTH (in Kapitel 3 vorgestellt) und eines direkt über die Tastatur (→ CMPLX) zur Verfügung. Nachfolgend werden beide CMPLX-Menüs vorgestellt.

CMPLX-Menü über das Menü MTH

Angenommen, das Systemflag 117 ist auf **CHOOSE boxes** (siehe Kapitel 2) eingestellt, wird das CMPLX-Untermenü innerhalb des MTH-Menüs wie folgt aufgerufen: ← MTH 9 OK. Die nachfolgende Sequenz von Momentaufnahmen veranschaulicht diese Schritte:





Das erste Menü (Optionen 1 bis 6) weist folgende Funktionen auf:

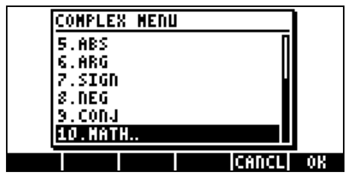
RE(z) : Reeller Teil einer komplexen Zahl

IM(z) : Imaginärer Teil einer komplexen Zahl

C→R(z) : Nimmt eine komplexe Zahl (x,y) und trennt sie in ihre reellen und imaginären Teile

R→C(x,y) : Bildet die komplexe Zahl *(x,y)* aus den reellen Zahlen x und y
ABS(z) : Berechnet die Magnitude einer komplexen Zahl oder den absoluten Wert einer reellen Zahl.
ARG(z) : Berechnet das Argument einer komplexen Zahl.

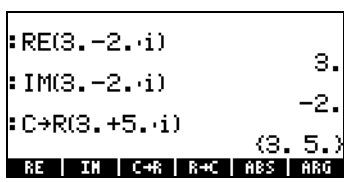
Die noch verbleibenden Optionen (Optionen 7 bis 10) sind nachfolgende:



SIGN(z) : Berechnet eine komplexe Zahl der Einheit Magnitude als z/|z|.
NEG : Ändert das Vorzeichen von z
CONJ(z) : Erzeugt die konjugiert komplexe Zahl von z

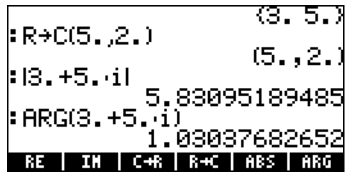
Nachfolgend einige Anwendungsbeispiele dieser Funktionen. Achten Sie darauf, dass im ALG-Modus das Argument der Funktion vorangestellt werden muss, während im RPN-Modus erst das Argument eingegeben und erst dann die Funktion ausgewählt wird. Vergessen Sie auch nicht, dass Sie diese Funktionen als Funktionsmenüs erreichen können, indem Sie das Systemflag 117 setzen (siehe Kapitel 3).

In der ersten Anzeige sind die Funktionen RE, IM und C→R zu sehen. Beachten Sie, dass diese Funktion eine Liste {3. 5.} zurückgibt, welche die reellen und imaginären Komponenten der komplexen Zahl darstellen:

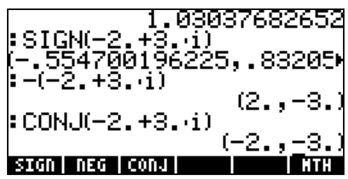


In der nachfolgenden Anzeige sind die Funktionen R→C, ABS und ARG abgebildet. Beachten Sie dabei, dass die Funktion ABS als |3.+5.·i| übersetzt wird, die Notation des absoluten Wertes. Auch das Ergebnis der

Funktion ARG, das einen Winkel darstellt, wird in den zuletzt ausgewählten Winkeleinheiten ausgegeben. In unserem Beispiel wird ARG(3.+5.·i) = 1,0303… in Radian ausgegeben.

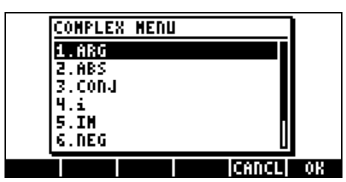


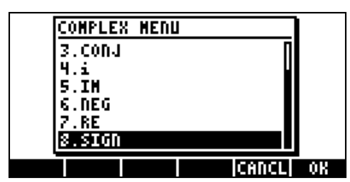
In der nächsten Abbildung stellen wir Beispiele zu den Funktionen SIGN, NEG (welche als das negative Zeichen – angezeigt wird) und CONJ dar.



Das CMPLX-Menü auf der Tastatur

Ein zweites CMPLX-Menü kann über die Tastatur aufgerufen werden, indem Sie die rechte Shift-Taste, der Taste / zugeordnet, d. h. → CMPLX eingeben. Ist das Systemflag 117 auf CHOOSE boxes eingestellt, wird das CMPLX-Menü wie folgt angezeigt:



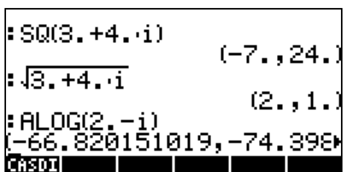


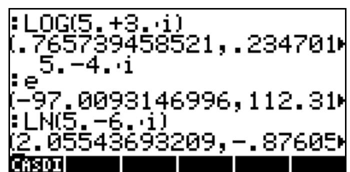
Das so erhaltene Menü enthält einige bereits im vorangegangenen Abschnitt vorgestellte Funktionen, und zwar ARG, ABS, CONJ, IM, NEG, RE und SIGN. Auch die Funktion *i*, die der Tastenkombination ← *i* entspricht, ist enthalten, sie ermöglicht, die imaginäre Einheit *i* in einen Ausdruck einzugeben.

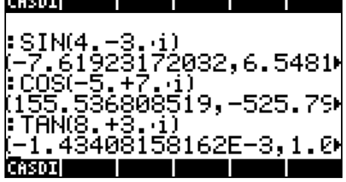
Das tastaturbasierte CMPLX-Menü ist eine Alternative zum MTH-basierten CMPLX-Menü, in dem Grundfunktionen für komplexe Zahlen enthalten sind. Nehmen Sie die zuvor gezeigten Beispiele unter Verwendung des tastaturbezogenen CMPLX-Menüs als Übung.

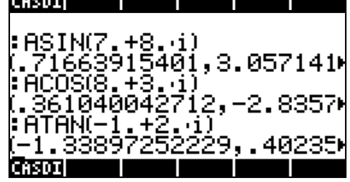
Auf komplexe Zahlen angewandte Funktionen

Viele der tastaturbasierten Funktionen für reelle Zahlen in Kapitel 3, z. B. SQ, LN, e^x, LOG, 10^X, SIN, COS, TAN, ASIN, ACOS oder ATAN können auch auf komplexe Zahlen angewendet werden. Das Ergebnis ist wieder eine komplexe Zahl, wie in nachfolgenden Beispielen zu sehen ist. Um diese Funktionen anzuwenden, gehen Sie genau wie vorher für reelle Zahlen beschrieben (siehe Kapitel 3) vor.





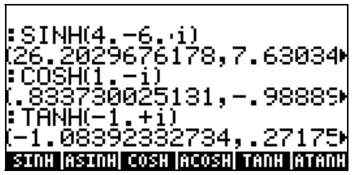


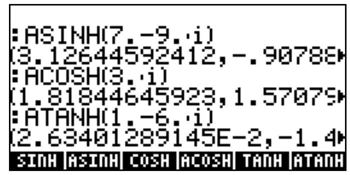


Anmerkung: Wenn Sie trigonometrische Funktionen und deren Inverse zusammen mit komplexen Zahlen verwenden, sind die Argumente keine Winkel mehr. Deshalb hat das für den Taschenrechner ausgewählte Winkelmaß in der Berechnung dieser Funktionen mit komplexen Argumenten keine Auswirkung auf die Berechnung. Informationen zum Verständnis und zur formalen Definition trigonometrischer und anderer Funktionen für komplexe Zahlen erhalten Sie in einem Buch über komplexe Variablen.

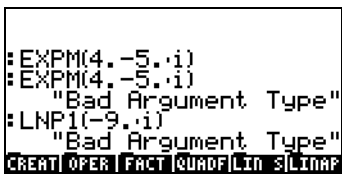
Funktionen aus dem MTH-Menü

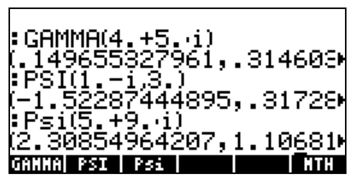
Die Hyperbelfunktionen und deren Inverse, wie auch die Gamma-, PSI- und Psi-Funktionen (Sonderfunktionen) wurden bereits in Kapitel 3 vorgestellt und auf reelle Zahlen angewandt. Diese Funktionen können auf dieselbe Weise wie auf reelle auch auf komplexe Zahlen angewendet werden. Nachfolgend einige Beispiele:





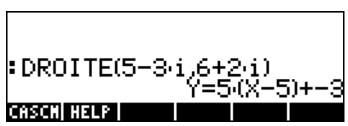
Die nachfolgende Anzeige zeigt, dass die Funktionen EXPM und LNP1 auf komplexe Zahlen nicht angewandt werden können. Hingegen akzeptieren die Funktionen GAMMA, PSI und PSi komplexe Zahlen:





Funktion DROITE: Gleichung einer Geraden

Die Funktion DROITE hat als Argument zwei komplexe Zahlen, beispielsweise x_1+iy_1 und x_2+iy_2 und gibt als Ergebnis die Gleichung einer Geraden, beispielsweise y = a+bx, welche die Punkte (x_1,y_1) und (x_2,y_2) enthält, aus. So z. B. kann die Linie zwischen den Punkten A(5,-3) und B(6,2) wie folgt ermittelt werden (Beispiel im algebraischen Modus):



Die Funktion DROITE finden Sie im Befehls-Katalog (CAT).

Kapitel 5
Algebraische und arithmetische Operationen

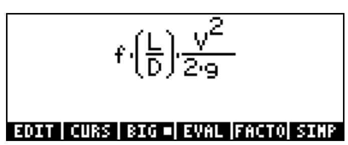
Ein algebraisches Objekt (auch als Algebraik bezeichnet) kann eine beliebige Zahl, Variable oder algebraischer Ausdruck sein, der nach den Regeln der Algebra berechnet, manipuliert oder kombiniert werden kann. Beispiele von algebraischen Objekten sind:

- Eine Zahl: 12,3, 15,2_m, ‘π’, ‘e’, ‘i’
- Der Name einer Variablen: ‘a’, ‘ux’, ‘width’ usw.
- Ein Ausdruck: ‘p*D^2/4’,‘f*(L/D)*(V^2/(2*g))’
- Eine Gleichung: ‘Q=(Cu/n)*A(y)*R(y)^(2/3)*So^0,5’

Eingabe von algebraischen Objekten

Algebraische Objekte können mithilfe von einfachen Anführungszeichen direkt in den Stack Ebene 1 oder über den EquationWriter ⇨ EQW eingegeben werden. Sie können beispielsweise das algebraische Objekt ‘π*D^2/4’ direkt in den Stack, Ebene 1 eingeben: ' ⇦ π × ALPHA D y^x 2 ÷ 4 ENTER. Die resultierende Anzeige wird links für den ALG-Modus und rechts für den RPN-Modus dargestellt:

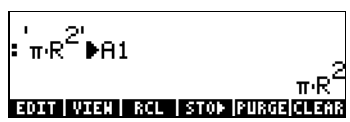
Ein algebraisches Objekt kann auch im EquationWriter erstellt und anschließend an den Stack gesendet werden. Die Bedienung des EquationWriters wurde in Kapitel 2 beschrieben. Zur Übung erstellen Sie das folgende algebraische Objekt im EquationWriter:



Nachdem Sie das Objekt erzeugt haben, drücken Sie, um es im Stack anzuzeigen (nachfolgend im ALG- und RPN-Modus angezeigt):

Einfache Operationen mit algebraischen Objekten

Algebraische Objekte können genau wie jede reelle oder komplexe Zahl addiert, subtrahiert, multipliziert, dividiert (ausgenommen durch Null), potenziert sowie als Argumente für eine Reihe von Standardfunktionen (exponential, logarithmisch, trigonometrisch, hyperbolisch usw.) verwendet werden. Um die Grundoperationen mit algebraischen Objekten zu veranschaulichen, erstellen wir einige Objekte, beispielsweise ‘π*R^2’ und ‘g*t^2/4’, und speichern sie in den Variablen A1 und A2 (siehe Kapitel 2, "Erstellen von Variablen"). Nachfolgend finden Sie die Tastenfolge, um im ALG-Modus die Variable A1 einzugeben: :['] [←] π [×] [ALPHA] [R] [y^x] [2] [▶] [STO▶] [ALPHA] [A] [1] [ENTER], die Anzeige sieht wie folgt aus:

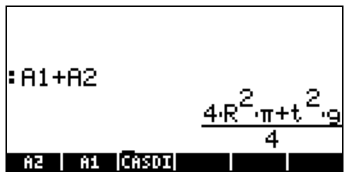


Die Tastenfolge für den RPN-Modus sieht so aus: ['] [←] π [×] [ALPHA] [R] [y^x] [2] [ENTER] [ALPHA] [A] [1] [STO▶]

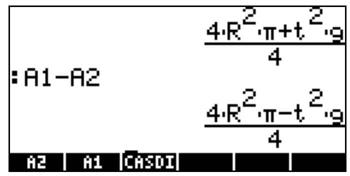
Nachdem Sie nun die Variable A2 gespeichert und die Taste gedrückt haben, erscheinen die Variablen in der Anzeige wie folgt:

Im ALG-Modus zeigen folgende Tastenanschläge eine Anzahl von Operationen mit algebraischen Zahlen, die in den Variablen A1 und A2 enthalten sind (drücken Sie VAR, um zum Variablen-Menü zurückzukehren):

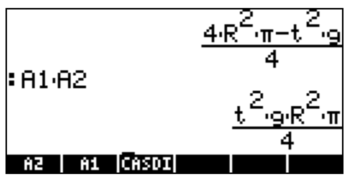
A1 + A2 ENTER



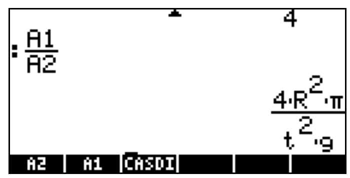
A1 − A2 ENTER



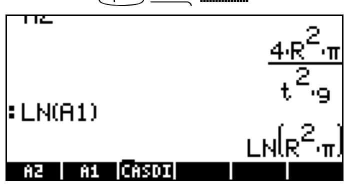
A1 × A2 ENTER



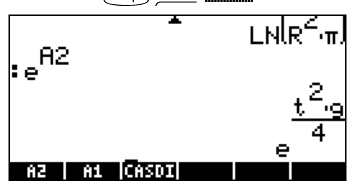
A1 ÷ A2 ENTER



→ LN A1



← e^x A2



Zum gleichen Ergebnis kommen Sie, wenn Sie im RPN-Modus die folgenden Tastenfolgen verwenden:

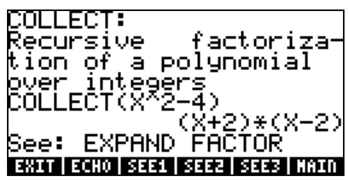
A1 ENTER A2 +   A1 ENTER A2 −
A1 ENTER A2 ×   A1 ENTER A2 ÷
A1 ENTER → LN   A2 ENTER ← e^x

Funktionen im Menü ALG

Das Menü ALG (algebraisch) erreicht man über die Tastenfolge → ALG (der Taste 4 zugeordnet). Ist das Systemflag 117 auf *CHOOSE* boxes gesetzt, zeigt das ALG-Menü folgende Funktionen an:

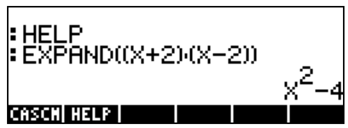
Wir möchten hier keine Beschreibung jeder einzelnen Funktion bringen, sondern den Anwender darauf hinweisen, dass er sie in der Hilfefunktion des Taschenrechners anzeigen lassen kann: TOOL NXT HELP ENTER . Um eine bestimmte Funktion auszuwählen, geben Sie den ersten Buchstaben der Funktion ein. So z. B. geben Sie für die Funktion COLLECT ALPHA C ein und verwenden anschließend die Pfeiltasten ▲▼, um im Hilfefenster zu COLLECT zu wechseln.

Um den Vorgang abzuschließen, drücken Sie OK. Nachfolgend die Hilfeansicht für die Funktion COLLECT:



Die Zeile "See:" (Siehe:"): EXPAND FACTOR am unteren Rand der Anzeige verweist auf andere Hilfefunktionseinträge, die Funktionen EXPAND und FACTOR. Um direkt zu diesen Einträgen zu gelangen, drücken Sie für EXPAND die Funktionstaste SEE1 und für FACTOR SEE2. Wenn Sie z. B. SEE1 drücken, erhalten Sie nachfolgende Informationen für EXPAND:

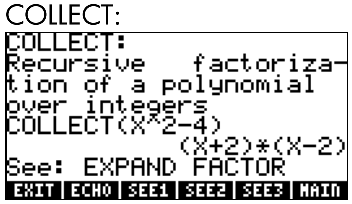
Die Hilfefunktion gibt nicht nur Informationen über die einzelnen Befehle, sondern auch ein Anwendungsbeispiel. Dieses Beispiel kann durch Drücken der Funktionstaste ECHO in Ihren Stack kopiert werden. Um z. B. für den obigen Eintrag zu EXPAND, das Beispiel in den Stack zu kopieren, drücken Sie die Funktionstaste ECHO (drücken Sie ENTER, um den Befehl auszuführen):



Nun überlassen wir es dem Benutzer, die Anwendung dieser Funktionen im ALG (oder ALGB) -Menü selbst zu ergründen. Dies ist eine Liste der Befehle:

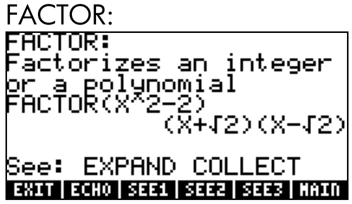
Die Hilfefunktion zeigt folgende Informationen zu den Befehlen:

COLLECT:

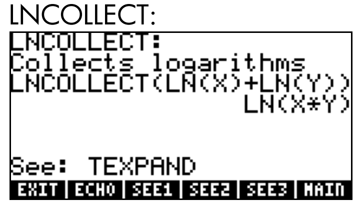


EXPAND:

FACTOR:

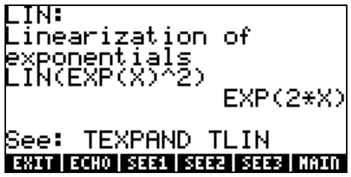


LNCOLLECT:

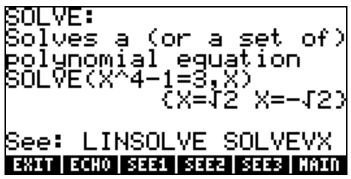


LIN:

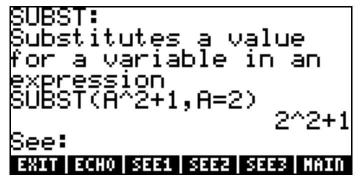
PARTFRAC:



SOLVE:



SUBST:



TEXPAND:

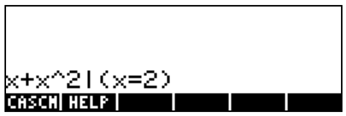
Anmerkung: im PRN-Modus muss das jeweilige Argument der Funktion vorangestellt werden, erst dann wird die Funktion selbst ausgewählt. So z. B. müssen Sie für TEXPAND im RPN-Modus, wie folgt vorgehen:

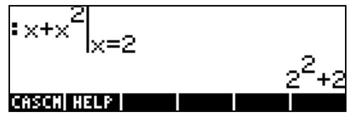
['] [←] e^x [+] [ALPHA] [X] [+] [ALPHA] [Y] [ENTER]

Wählen Sie an dieser Stelle die Funktion TEXPAND aus dem Menü ALG (oder direkt aus dem Katalog [→] CAT), um die Operation abzuschließen.

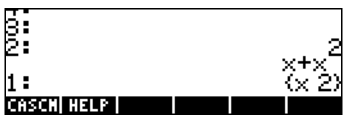
Weitere Möglichkeiten zum Ersetzen in algebraischen Ausdrücken

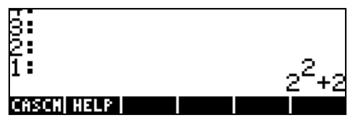
Die oben gezeigte Funktion SUBST wird dazu verwendet, um eine Variable in einem Ausdruck zu ersetzen. Eine zweite Art des Ersetzens kann mit der Tastenfolge [→] | (der Taste I zugeordnet) durchgeführt werden. Als Beispiel wird die nachfolgende Eingabe den Wert $x = 2$ im Ausdruck $x+x^2$ im ALG-Modus ersetzen. Die Zahl auf der linken Seite zeigt, wie dieser Ausdruck einzugeben ist, (der ersetzte Wert, x=2, muss zwischen zwei Klammern stehen) bevor Sie die Taste ENTER betätigen. Nachdem die Taste ENTER gedrückt wurde, wird das Ergebnis in der rechten Abbildung angezeigt:



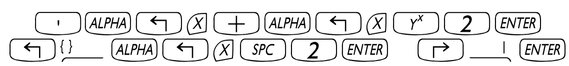


Im RPN-Modus wird dies erreicht, indem Sie zuerst den Ausdruck, in dem der Austausch stattfinden soll ($x+x^2$), gefolgt von einer Liste (siehe Kapitel 8) mit den zu ersetzenden Variablen, einem Leerzeichen und dem Wert, der ausgetauscht werden soll, d. h. {x 2}, eingeben. Der letzte Schritt, um den Vorgang abzuschließen, ist die Tastenkombination: ↱ | .

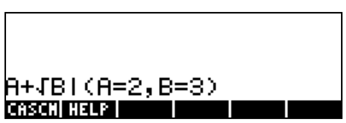


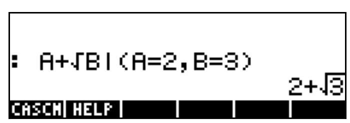


Die dazu erforderlichen Tastenanschläge lauten wie folgt:



Im ALG-Modus kann mehr als eine Variable ersetzt werden, wie dies im nachfolgenden Beispiel gezeigt wird (vor und nachdem Sie die Taste ENTER gedrückt haben).

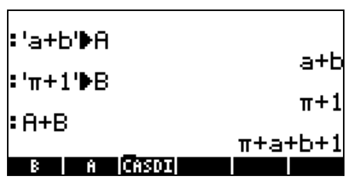




Im RPN-Modus kann ebenfalls mehr als eine Variable gleichzeitig ersetzt werden, wie in nachfolgendem Beispiel angezeigt: Beachten Sie, dass im RPN-Modus eine Liste von Variablen-Namen und Werten für den Austausch verwendet wird.

Ein anderen Ansatz für den Austausch ist, die im Taschenrechner zu ersetzenden Ausdrücke als Variable zu definieren und die Namen der Variablen in den ursprünglichen Ausdruck einzufügen. Im ALG-Modus speichern Sie z. B. folgende Variablen:

Geben Sie anschließend den Ausdruck A+B ein:



Der zuletzt eingefügte Ausdruck wird nach Drücken der Taste ENTER automatisch ausgewertet und bringt das oben gezeigte Ergebnis.

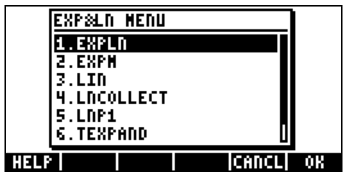
Operationen mit transzendenten Funktionen

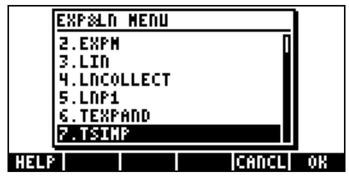
Im Taschenrechner stehen eine ganze Reihe von Funktionen, welche zum Austausch von Ausdrücken mit logarithmischen, Exponential-, trigonometrischen und Hyperbel-Funktionen, als trigonometrische Einheiten oder als Exponentialfunktionen verwendet werden können, zur Verfügung. Die Menüs mit Funktionen zum Austausch von trigonometrischen Funktionen können direkt von der Tastatur aus gestartet werden, indem Sie die rechte Shift-Taste gefolgt von der Taste 8, d. h. ↱ TRIG drücken. Die Kombination dieser Taste mit der linken Shift-Taste, d. h. ↱ EXP&LN startet ein Menü, welches Ihnen erlaubt, Ausdrücke anhand von Exponential oder natürlichen

Logarithmus-Funktionen zu ersetzen. In den folgenden Abschnitten werden diese Menüs im Detail vorgestellt.

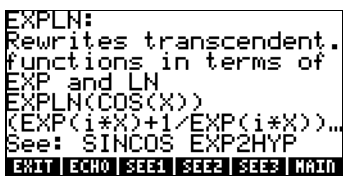
Erweitern und Zusammenfassen mithilfe der log-exp-Funktionen

Mit ← EXP&LN erhalten Sie das folgendes Menü:





Informationen und Beispiele zu diesen Befehlen erhalten Sie über die Hilfefunktion des Taschenrechners. Einige der Befehle aus dem Menü EXP&LN, d. h. LIN, LNCOLLECT und TEXPAND befinden auch im ALG-Menü, welches vorher vorgestellt wurde. Die Funktionen LNP1 und EXPM wurden im Menü HYPERBOLIC unter dem Menü MTH vorgestellt (siehe Kapitel 2). Die einzige noch verbleibende Funktion ist EXPLN. Die Beschreibung dieser Funktion ist auf der linken Abbildung zu sehen, das Beispiel dazu aus der Hilfefunktion finden Sie in der rechten Abbildung:



Erweitern und Zusammenfassen anhand trigonometrischer Funktionen

Das Menü TRIG wird über die Tastenkombination → TRIG aufgerufen und enthält folgende Funktionen:

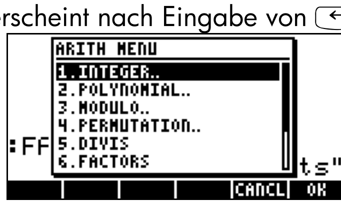
Mithilfe dieser Funktionen können Ausdrücke durch Ersetzen einer bestimmten trigonometrischen Kategorie durch eine andere vereinfacht werden. So z. B. erlaubt die Funktion ACOS2S das Ersetzen der Funktion *arccosine* (acos(x)) durch deren Ausdruck *arcsine* (asin(x)).

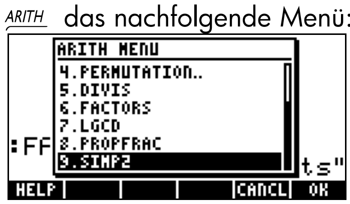
Eine Beschreibung dieser Befehle und Beispiele und ihrer Anwendung finden Sie über die Hilfefunktion des Taschenrechners (TOOL NXT HELP). Der Anwender wird dazu aufgefordert, diese Hilfe nach Informationen zu den Befehlen im Menü TRIG zu durchsuchen.

Beachten Sie, dass der erste Befehl im Menü TRIG das Menü HYPERBOLIC darstellt, dessen Funktionen in Kapitel 2 bereits vorgestellt wurden.

Funktionen im Menü ARITHMETIC

Das Menü ARITHMETIC enthält eine Anzahl von Untermenüs für spezifische Anwendungen in der Zahlentheorie (Integer, Polynome usw.), aber auch eine Reihe von Funktionen für allgemeine arithmetische Operationen. Das ARITHMETIC-Menü wird über die Tastenkombination ← ARITH (der Taste / zugeordnet) gestartet. Ist für das Systemflag 117 das *CHOOSE* boxes gewählt, erscheint nach Eingabe von ← ARITH das nachfolgende Menü:



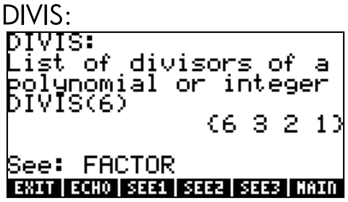


Die Optionen 5 bis 9 (*DIVIS, FACTORS, LGCD, PROPFRAC, SIMP2*) aus dieser Liste entsprechen den allgemeinen Funktionen für Ganzzahlen und Polynome. Die verbliebenen Optionen (*1. INTEGER*, *2. POLYNOMIAL*, *3.*

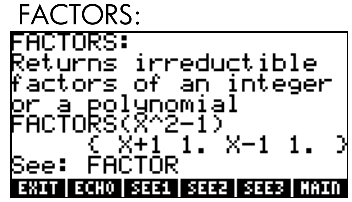
MODULO and *4. PERMUTATION*) sind eigentlich Untermenüs von Funktionen, welche bestimmten mathematischen Objekten zugeordnet sind. Der Unterschied zwischen den Untermenüs (Optionen 1 bis 4) und reinen Funktionen (Optionen 5 bis 9) wird klar, wenn das Systemflag 117 auf *SOFT*-Menüs gesetzt ist. Starten Sie das Menü ARITHMETIC (ARITH) erhalten Sie nun:

Nachfolgend stellen wir Ihnen die Hilfefunktion für die Einträge der Funktionen aus den Optionen 5 bis 9 aus dem Menü ARITHMETIC vor:

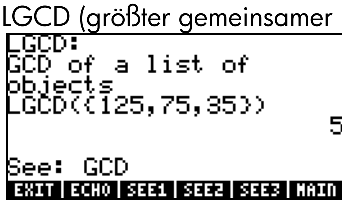
DIVIS:



FACTORS:

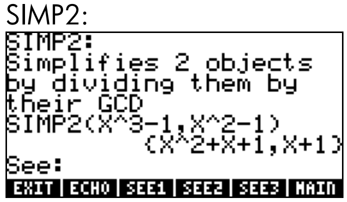


LGCD (größter gemeinsamer Nenner):



PROPFRAC (reiner Bruch):

SIMP2:



Die den ARITHMETIC Untermenüs zugeordneten Funktionen INTEGER, POLYNOMIAL, MODULO und PERMUTATION sind wie folgt:

Menü INTEGER

EULER	Integer-Zahlen < n, die koprim/teilerfremd mit n sind
IABCUV	Löst au + bv = c, wobei a, b, c = Integer-Zahlen sind
IBERNOULLI	n-te Bernoulli Zahl
ICHINREM	Chinesischer Restesatz für Integer-Zahlen
IDIV2	Euklidische Division von zwei Integer-Werten
IEGCD	Gibt als Ergebnis u, v, sodass au + bv = gcd(a,b)
IQUOT	Euklidischer Quotient zweier Integer-Werte
IREMAINDER	Euklidischer Algorithmus für zwei Integer-Werte mit Rest
ISPRIME?	Überprüft, ob eine Integer-Zahl eine Primzahl ist
NEXTPRIME	Nächste Primzahl für eine vorgegebene Integer-Zahl
PA2B2	Primzahl als Quadratnorm einer komplexen Zahl
PREVPRIME	Vorhergehende Primzahl für eine vorgegebene Integer-Zahl

Menü POLYNOMIAL (Polynome)

ABCUV	Polynomengleichung nach Bézout (au+bv=c)
CHINREM	Chinesischer Restesatz für Polynome
CYCLOTOMIC	n-tes Kreisteilungs-Polynom
DIV2	Euklidische Division zweier Polynome
EGDC	Gibt u, v aus au+bv=gcd(a,b) zurück
FACTOR	Zerlegt eine Integer-Zahl oder ein Polynom in Faktoren
FCOEF	Erzeugt einen Bruch bei gegebenen Wurzeln und Mehrwertigkeit
FROOTS	Gibt Wurzel und Mehrwertigkeit eines Bruches zurück
GCD	Größter gemeinsamer Teiler zweier Zahlen eines Polynoms
HERMITE	Hermite-Polynom n-ten Grades
HORNER	Hornersche Berechnung eines Polynoms
LAGRANGE	Lagrange-Interpolation des Polynoms
LCM	Kleinstes gemeinsames Vielfaches zweier Zahlen oder Polynome
LEGENDRE	Legendre-Polynom n-ten Grades
PARTFRAC	Partialbruch-Zerlegung eines gegebenen Bruches
PCOEF	(Hilfefunktionseintrag fehlt)

PTAYL	Gibt Q(x-a) in Q(x-a) = P(x) zurück, Taylor Polynom
QUOT	Euklidischer Quotient zweier Polynome
RESULTANT	Determinante der Sylvester-Matrix zweier Polynome
REMAINDER	Euklidischer Restesatz zweier Polynome
STURM	Sturm-Kette eines Polynoms
STURMAB	Zeichen an unterer Grenze und Anzahl der Nullen zwischen den Grenzen

Menü MODULO

ADDTMOD	Addition zweier Ausdrücke, Modulo aktuelles Modul
DIVMOD	Division zweier Polynome, Modulo aktuelles Modul
DIV2MOD	Euklidische Division zweier Polynome, mit Modulo-Koeffizienten
EXPANDMOD	Erweitert/vereinfacht Polynome, Modulo aktuelles Modul
FACTORMOD	Zerlegt ein Polynom Modulo in Faktoren, aktuelles Modul
GCDMOD GCD	Größter gemeinsamer Teiler zweier Polynome, Modulo aktuelles Modul
INVMOD	Inverse des Integer-Modulo, aktuelles Modul
MOD	(Hilfefunktionseintrag fehlt)
MODSTO	Ändert die Modulo-Einstellung auf den angegebenen Wert
MULTMOD	Multiplikation zweier Polynome, Modulo aktuelles Modul
POWMOD	Potenziert ein Polynom auf einen Exponenten, Modulo aktuelles Modul
SUBTMOD	Subtraktion zweier Polynome, Modulo aktuelles Modul

Anwendungen des Menüs ARITHMETIC

In diesem Abschnitt werden Hintergrundinformationen für die Anwendung der Funktionen des ARITHMETIC-Menüs dargestellt. Gleich anschließend werden Definitionen aus den Themenbereichen von Polynomen, polynomischen Brüchen und zur modularen Arithmetik erläutert. Die vorgestellten Beispiele werden unabhängig von den Einstellungen des Taschenrechners (ALG oder RPN) vorgestellt.

Modulare Arithmetik

Nehmen wir ein Zahlensystem bestehend aus Integer-Zahlen, welche periodisch auf sich selbst zurückgehen und neu starten, wie die Stunden einer Uhr. Ein solches Zählsystem wird als Ring bezeichnet. Da die in einem Ring verwendete Anzahl von Integer-Werten begrenzt ist, wird die Arithmetik in diesem Ring als endliche Arithmetik bezeichnet. Nehmen wir an, unsere endliche Zahl an Integer-Werten besteht aus den Zahlen 0, 1, 2, 3, …, n-1, n. Die Arithmetik in diesem Zählsystem können wir auch als modulare Arithmetik des Moduls n bezeichnen. Im Falle der Stunden einer Uhr, wäre das Modul 12. (Wenn wir jedoch in der modularen Arithmetik mit den Stunden einer Uhr arbeiten, müssten wir die Integer-Zahlen 0, 1, 2, 3, …, 10, 11, und nicht 1, 2, 3,…,11, 12) verwenden.

Operationen in modularer Arithmetik

Addition in modularer Arithmetik mit dem Modul *n*, welches eine positive Integer-Zahl darstellt, wenn *j* und *k* zwei positive Integer-Zahlen, beide kleiner als *n* sind und $j+k \geq n$ gilt, wird $j+k$ als $j+k-n$ definiert. Im Beispiel mit unserer Uhr wäre das, für n = 12, 6+9 "=" 3. Um diese "Gleichwertigkeit" von unendlichen arithmetischen Gleichheiten zu unterscheiden, wird das Symbol ≡ anstelle des Gleichzeichens gesetzt und das Verhältnis zwischen diesen Zahlen als *Kongruenz*und nicht als Gleichwertigkeit bezeichnet. Somit würden wir für das obige Beispiel $6+9 \equiv 3 \pmod{12}$ schreiben und diesen Ausdruck wie folgt lesen "*sechs plus neun ist kongruent zu drei, Modul 12*". Stellen die Zahlen die Stunden seit Mitternacht dar, kann z. B. die Kongruenz 6+9 ≡ 3 (mod 12) als "sechs Stunden nach der neunten Stunde nach Mitternacht, wird drei Stunden nach Mittag sein" interpretieren. Andere Summen, welche in Modul 12-Arithmetik definiert werden können sind beispielsweise 2+5 ≡ 7 (mod 12); 2+10 ≡ 0 (mod 12); 7+5 ≡ 0 (mod 12) usw.

Die Regeln für die Subtraktion lauten wie folgt: wenn $j - k < 0$, dann wird $j-k$ als $j-k+n$ definiert. Somit liest man $8-10 \equiv 2 \pmod{12}$ als "*acht minus zehn ist kongruent zu zwei, Modul zwölf*". Ein weiteres Beispiel einer Subtraktion in

Modul 12-Arithmetik wäre 10-5 ≡ 5 (mod 12); 6-9 ≡ 9 (mod 12); 5 – 8 ≡ 9 (mod 12); 5 –10 ≡ 7 (mod 12) usw.

Die *Multiplikation* erfolgt nach der Regel, dass wenn $j{\cdot}k > n$, wobei $j{\cdot}k = m{\cdot}n + r$, wobei *m* und *r* positive Integer-Zahlen kleiner als *n* darstellen, dann ist $j{\cdot}k \equiv r$ (mod *n*). Das Produkt von *j* mal *k* in Modul *n*-Arithmetik ist im Grunde genommen der Ganzzahlrest von $j{\cdot}k/n$ in der unendlichen Arithmetik, wenn $j{\cdot}k>n$. So gilt z. B. in Modul 12-Arithmetik 7·3 = 21 = 12 + 9, (oder 7·3/12 = 21/12 = 1 + 9/12, d. h., der Ganzzahlrest von 21/12 ist 9). Wir können nun 7·3 ≡ 9 (mod 12) schreiben und das letzte Ergebnis wie folgt lesen: "*sieben mal drei ist kongruent zu neun, Modul zwölf*".

Der *Divisionsvorgang* kann als Multiplikation wie folgt ausgeführt werden $r/k \equiv j$ (mod *n*), wenn $j{\cdot}k \equiv r$ (mod *n*). Das bedeutet, dass *r* den Restwert von $j{\cdot}k/n$ darstellen muss. So gilt z. B. 9/7 ≡ 3 (mod 12), weil 7·3 ≡ 9 (mod 12) darstellt. Einige Divisionen sind in der modularen Arithmetik nicht erlaubt. So können Sie z. B. in Modul 12-Arithmetik 5/6 (mod 12) nicht definieren, weil die Multiplikationstabelle von 6 das Ergebnis 5 in Modul 12-Arithmetik nicht anzeigt. Nachfolgend die Multiplikationstabelle:

6*0 (mod 12)	0	6*6 (mod 12)	0
6*1 (mod 12)	6	6*7 (mod 12)	6
6*2 (mod 12)	0	6*8 (mod 12)	0
6*3 (mod 12)	6	6*9 (mod 12)	6
6*4 (mod 12)	0	6*10 (mod 12)	0
6*5 (mod 12)	6	6*11 (mod 12)	6

Formale Definition eines endlichen arithmetischen Ringes

Der Ausdruck $a \equiv b \ (mod\ n)$ wird als "a ist kongruent zu b, Modulo n" interpretiert und gilt, wenn $(b\text{-}a)$ ein Vielfaches von *n* ist. Anhand dieser Definition werden die arithmetischen Regeln wie folgt vereinfacht:

Wenn $a \equiv b \ (mod\ n)$ und $c \equiv d \ (mod\ n)$,

dann

$$a+c \equiv b+d \ (mod\ n),$$
$$a\text{-}c \equiv b - d \ (mod\ n),$$

$$a \times c \equiv b \times d \; (mod\ n).$$

Für die Division befolgen Sie die zuvor beschriebenen Regeln. Z. B. ist 17 ≡ 5 (mod 6) und 21 ≡ 3 (mod 6). Unter Verwendung dieser Regeln können wir schreiben:

17 + 21 ≡ 5 + 3 (mod 6) => 38 ≡ 8 (mod 6) => 38 ≡ 2 (mod 6)
17 – 21 ≡ 5 -3 (mod 6) => -4 ≡ 2 (mod 6)
17 × 21 ≡ 5 × 3 (mod 6) => 357 ≡ 15 (mod 6) => 357 ≡ 3 (mod 6)

Beachten Sie, dass immer, wenn das Ergebnis auf der rechten Seite der "Kongruenz" größer als das Modulo ist, (in diesem Fall n = 6), können Sie immer ein Vielfaches des Modulo von diesem Ergebnis abziehen und zu einer Zahl, die kleiner als das Modulo ist, vereinfachen. Somit können die Ergebnisse aus dem ersten Fall *8 (mod 6)* auf *2 (mod 6)* vereinfacht werden und die Ergebnisse im dritten Fall, *15 (mod 6)* auf *3 (mod 6).* Sie können mit dieser Funktionalität problemlos umgehen, wenn Sie den Taschenrechner diese Operationen durchführen lassen. Lesen Sie den nächsten Abschnitt für ein besseres Verständnis der Verarbeitung endlicher arithmetischer Ringe in Ihrem Taschenrechner.

Endliche arithmetische Ringe im Taschenrechner

Schon immer haben wir unsere endlichen arithmetischen Operationen so definiert, dass deren Ergebnisse einen positiven Wert ergeben. Das in Ihrem Taschenrechner existierende arithmetische System ist so eingestellt, dass das Modul Ring n die Zahlen *-n/2+1, …,-1, 0, 1,…,n/2-1, n/2*, enthält, wenn n eine gerade Zahl ist und *–(n-1)/2, -(n-3)/2,…,-1,0,1,…,(n-3)/2, (n-1)/2* enthält, wenn n eine ungerade Zahl ist. Für n = 8 (gerade) umfasst der arithmetische Ring Ihres Taschenrechners z. B. die Zahlen (-3,-2,-1,0,1,3,4), während für n = 7 (ungerade) der entsprechende endliche arithmetische Ring die Zahlen (-3,-2,-1,0,1,2,3) umfasst.

Modulare Arithmetik im Taschenrechner

Um das modulare arithmetische Menü im Taschenrechner zu starten, wählen Sie das Untermenü MODULO aus dem Menü ARITHMETIC (↰ ARITH). Das Menü beinhaltet die Funktionen ADDTMOD, DIVMOD, DIV2MOD,

EXPANDMOD, FACTORMOD, GCDMOD, INVMOD, MOD, MODSTO, MULTMOD, POWMOD und SUBTMOD. Eine kurze Beschreibung dieser Funktionen finden Sie in dem entsprechenden Abschnitt oben. Zunächst stellen wir die Anwendung dieser Funktionen vor.

Einstellung des Moduls (oder MODULO)

Im Taschenrechner befindet sich eine Variable mit dem Namen MODULO , welche sich im Verzeichnis {HOME CASDIR} befindet und in welcher die Magnitude für modulare arithmetische Anwendungen gespeichert ist.

Der Standartwert für MODULO lautet 13. Um den Wert von MODULO zu ändern, können Sie den neuen Betrag direkt in die Variable MODULO ins Unterverzeichnis {HOME CASDIR} speichern. Alternativ dazu können Sie einen neuen MODULO-Wert über die Funktion MODSTO speichern.

Modulare arithmetische Operationen mit Zahlen

Verwenden Sie die Funktionen ADDTMOD, SUBTMOD, MULTMOD, DIV2MOD und DIVMOD (für Division) und POWMOD zur Addition, Subtraktion, Multiplikation, Division oder Potenzierung in der modularen Arithmetik. Im RPN-Modus müssen Sie zwei Zahlen für die Operation eingeben, beide durch ein [ENTER] oder ein [SPC] voneinander getrennt, und anschließend die entsprechende modulare arithmetische Funktion drücken. Möchten Sie z. B. Modul von 12 verwenden, versuchen Sie folgende Operationen:

ADDTMOD-Beispiel

$6+5 \equiv -1 \pmod{12}$	$6+6 \equiv 0 \pmod{12}$	$6+7 \equiv 1 \pmod{12}$
$11+5 \equiv 4 \pmod{12}$	$8+10 \equiv -6 \pmod{12}$	

SUBTMOD-Beispiele

$5 - 7 \equiv -2 \pmod{12}$	$8 - 4 \equiv 4 \pmod{12}$	$5 - 10 \equiv -5 \pmod{12}$
$11 - 8 \equiv 3 \pmod{12}$	$8 - 12 \equiv -4 \pmod{12}$	

MULTMOD-Beispiele

$6 \cdot 8 \equiv 0 \pmod{12}$	$9 \cdot 8 \equiv 0 \pmod{12}$	$3 \cdot 2 \equiv 6 \pmod{12}$
$5 \cdot 6 \equiv 6 \pmod{12}$	$11 \cdot 3 \equiv -3 \pmod{12}$	

DIVMOD-Beispiele

12/3 ≡ 4 (mod 12)	12/8 (mod 12) gibt es nicht
25/5 ≡ 5 (mod 12)	64/13 ≡ 4 (mod 12)
66/6 ≡ -1 (mod 12)	

DIV2MOD-Beispiele

2/3 (mod 12) gibt es nicht
26/12 (mod 12) gibt es nicht
125/17 (mod 12) ≡ 1 mit Restwert = 0
68/7 ≡ -4 (mod 12) mit Restwert = 0
7/5 ≡ -1 (mod 12) mit Restwert = 0

Anmerkung: DIVMOD ermittelt den Quotienten der modularen Division j/k (mod n), während DIMV2MOD nicht nur den Quotienten, sondern auch den Restwert der modularen Division j/k (mod n) ermittelt.

POWMOD-Beispiele

2^3≡ -4 (mod 12)	3^5≡ 3 (mod 12)	5^{10}≡ 1 (mod 12)
11^8 ≡ 1 (mod 12)	6^2 ≡ 0 (mod 12)	9^9 ≡ -3 (mod 12)

In den oben gezeigten Beispielen mit modularen Arithmetik-Operationen haben wir Zahlen benutzt, die nicht unbedingt zum Ring gehören, d. h. Zahlen wie 66, 125, 17 usw. Der Taschenrechner konvertiert diese Zahlen erst in Ringwerte und wendet erst dann die Operationen auf sie an. Sie können Zahlen auch selbst mit der Funktion EXPANDMOD in einen Ringwert konvertieren. So zum Beispiel:

EXPANDMOD(125) ≡ 5 (mod 12)
EXPANDMOD(17) ≡ 5 (mod 12)
EXPANDMOD(6) ≡ 6 (mod 12)

Die modulare Inverse einer Zahl

Nehmen wir an eine Zahl *k* gehört einem endlichen arithmetischen Ring des Moduls *n* an, dann ist die modulare Inverse von k, d. h. *1/k (mod n),* eine Zahl *j*, die sich als *j·k ≡ 1 (*mod *n*) verhält. Die modulare Inverse einer Zahl erhält man mit der Funktion INVMOD in MODULO im Untermenü des Menüs ARITHMETIC. In Modul 12-Arithmetik z. B.:

1/6 (mod 12) gibt es nicht
1/5 ≡ 5 (mod 12)
1/7 ≡ -5 (mod 12)
1/3 (mod 12) gibt es nicht
1/11 ≡ -1 (mod 12)

Der MOD-Operator

Der MOD-Operator wird zur Ermittlung der Ringzahl, eines gegebenen Moduls, entsprechend einer gegebenen Integer-Zahl, verwendet. Auf Papier wird diese Operation als m mod $n = p$ geschrieben und wir "m Modul von n ist gleich p" gelesen. So z. B., um 15 mod 8 zu berechnen, geben Sie ein:

- im ALG-Modus: 1 5 MOD 8 ENTER
- im RPN-Modus: 1 5 ENTER 8 ENTER MOD

Das Ergebnis ist 7, d. h. 15 mod 8 = 7. Versuchen Sie folgende Übungen:

18 mod 11 = 7   23 mod 2 = 1   40 mod 13 = 1
23 mod 17 = 6   34 mod 6 = 4

Eine praktische Anwendung der Funktion MOD für Programmierzwecke besteht darin, herauszufinden, ob eine Integer-Zahl gerade oder ungerade ist, da n mod 2 = 0, wenn n gerade ist und n mod 2 = 1, wenn n ungerade ist. Sie kann auch zur Ermittlung, ob m ein Vielfaches einer anderen Integer-Zahl n ist; dies ist der Fall, wenn m mod n = 0 ist, verwendet werden.

Anmerkung: In der Hilfefunktion des Taschenrechners finden Sie eine Beschreibung und Beispiele zu weiterer modularer Arithmetik. Viele dieser Funktionen können auf Polynome angewandt werden. Informationen zur modularen Arithmetik mit Polynomen finden Sie in Büchern zur Zahlentheorien.

Polynome

Polynome sind algebraische Ausdrücke, die aus einem oder mehreren Gliedern, welche abfallende Potenzen einer gegebenen Variable enthalten, bestehen. So ist z. B. der Ausdruck 'X^3+2*X^2-3*X+2' ein Polynom dritten Grades der Variablen X, während 'SIN(X)^2-2' ein Polynome zweiten Grades in SIN(X) darstellt. Eine Aufzählung von Funktionen zu Polynomen im Menü ARITHMETIC wurde vorhin dargestellt. Als Nächstes finden Sie einige allgemeine Definitionen zu Polynomen. In diesen Definitionen stellen *A(X), B(X), C(X), P(X), Q(X), U(X), V(X) usw. Polynome dar.*

- Polynombruch: ein Bruch, dessen Zähler und Nenner Polynome sind, beispielsweise C(X) = A(X)/B(X)
- Wurzeln oder Nullen eines Polynoms: Werte von X, für die P(X) = 0
- Pole eines Bruches: Wurzeln des Nenners
- Mehrwertigkeit der Wurzeln oder Pole: die Anzahl des Auftretens einer Wurzel, z. B. $P(X) = (X+1)^2(X-3)$ hat die Wurzeln {-1, 3} mit den Mehrwertigkeiten {2,1}
- Kreisteilungs-Polynom ($P_n(X)$): ein Polynom EULER(*n*)- Grades, dessen Wurzeln die primitive n-te Wurzel der Einheit ist, z. B. $P_2(X) = X+1$, $P_4(X) = X^2+1$
- Bézouts Polynomgleichung: A(X) U(X) + B(X)V(X) = C(X)

Nachstehend finden Sie spezifische Anwendungsbeispiele von Polynomen.

Modulare Arithmetik mit Polynomen

Auf die gleiche Art, wie wir einen endlichen arithmetischen Ring für Zahlen in einem vorangegangenen Abschnitt definiert haben, können wir auch einen endlichen arithmetischen Ring für Polynome mit einem gegebenen Polynom als Modul definieren. So können wir z. B. ein bestimmtes Polynom *P(X)* als *P(X) = X (mod* X^2*) definieren* oder ein anderes Polynom als *Q(X) = X + 1 (mod X-2).*

Ein Polynom *P(X)* ist Teil eines arithmetischen Ringes von Polynomen des Moduls *M(X),* wenn es ein drittes Polynom *Q(X)* gibt, und zwar so, dass *(P(X) – Q(X))* ein Vielfaches von *M(X)* darstellt. Dann würden wir schreiben: *P(X) ≡ Q(X) (mod M(X)).* Letzterer Ausdruck wird als "*P(X) ist kongruent mit Q(X) modulo M(X)"* interpretiert.

Die Funktion CHINREM

CHINREM steht für CHINese REMainder (Chinesischer Restesatz). Die in diesem Befehl kodierte Operation löst ein System von Kongruenten unter Anwendung des Chinesischen Restesatz-Theorems . Dieser Befehl kann mit Polynomen (wie auch mit Integer-Zahlen, Funktion ICHINREM) verwendet werden. Die Eingabe besteht aus zwei Vektoren *[expression_1, modulo_1] und [expression_2, modulo_2].* Die Ausgabe ist ein Vektor *[expression_3, modulo_3]*, wobei *modulo_3* aus dem Produkt *(modulo_1)·(modulo_2)* ermittelt wird. Beispiel: CHINREM(['X+1', 'X^2-1'],['X+1','X^2']) = ['X+1',-(X^4-X^2)]

Ansatz für das Chinesische Restsatz-Theorem für Integer-Zahlen

Wenn $m_1, m_2, \ldots, m_r$ paarweise teilerfremde natürliche Zahlen und $a_1, a_2, \ldots, a_r$ beliebige Integer-Zahlen sind, dann gibt es genau eine Integer-Zahl x, welche gleichzeitig die folgenden Kongruenzen erfüllt: $x \equiv a_1 \pmod{m_1}$, $x \equiv a_2 \pmod{m_2}$, …, $x \equiv a_r \pmod{m_r}$. Zusätzlich, wenn x = a eine Lösung ist, dann sind alle anderen Lösungen kongruent und gleich dem Produkt von $m_1 \cdot m_2 \cdot \ldots m_r$.

Die Funktion EGCD

EGCD steht für Extended Greatest Common Divisor. Für zwei Polynome A(X) und B(X) erzeugt die Funktion EGCD die Polynome C(X), U(X), and V(X), sodass gilt C(X) = U(X)*A(X) + V(X)*B(X). So z. B. für A(X) = X^2+1, B(X) = X^2-1, EGCD(A(X),B(X)) = {2, 1, -1}, d. h., 2 = 1*(X^2+1')-1*(X^2-1). Auch EGCD('X^3-2*X+5','X') = { 5, '-(X^2-2)', 1}, d. h. 5 = – (X^2-2)*X + 1*(X^3-2*X+5).

Die Funktion GCD

Die Funktion GCD (Greatest Common Denominator – größter gemeinsamer Nenner) dient zur Ermittlung des größten gemeinsamen Nenners zweier Polynome oder zweier Listen von Polynomen der selben Länge. Bevor Sie die Funktion GCD anwenden, müssen die beiden Polynome oder Listen von Polynomen in Stack-Ebene 2 und 1 verschoben werden. Das Ergebnis ist ein Polynom oder eine Liste, die den größten gemeinsamen Nenner der beiden

Polynome oder aller Listen der Polynome darstellt. Es folgen Beispiele im RPN-Modus (Taschenrechner steht im Exact Modus):
'X^3-1' ENTER 'X^2-1' ENTER GCD ergibt: 'X-1'
{'X^2+2*X+1','X^3+X^2'} ENTER {'X^3+1','X^2+1'} ENTER GCD ergibt {'X+1' 1}

Die Funktion HERMITE

Die Funktion HERMITE [HERMI] verwendet als Argument eine Integer-Zahl, k und gibt das Hermite-Polynom k-ten Grades zurück. Ein Hermite-Polynom, $He_k(x)$ wird definiert als

$$He_0 = 1, \quad He_n(x) = (-1)^n e^{x^2/2} \frac{d^n}{dx^n}(e^{-x^2/2}), \quad n = 1,2,...$$

Eine alternative Definition eines Hermite-Polynoms lautet

$$H_0{}^* = 1, \quad H_n{}^*(x) = (-1)^n e^{x^2} \frac{d^n}{dx^n}(e^{-x^2}), \quad n = 1,2,...$$

wobei d^n/dx^n = n-te Ableitungsfunktion zu x darstellt. Dies ist die im Taschenrechner verwendete Definition.

Beispiele: Die Hermite-Polynome dritten und fünften Grades lauten wie folgt:

HERMITE(3) = '8*X^3-12*X' ,

und HERMITE(5) = '32*x^5-160*X^3+120*X'.

Die Funktion HORNER

Die Funktion HORNER erzeugt die Horner- oder synthetische Division eines Polynoms P(X) durch den Faktor (X-*a*). Als Eingabe der Funktion wird das Polynom P(X) und die Zahl a benötigt. Die Funktion gibt – in dieser Reihenfolge – den Quotienten des Polynoms Q(X), welcher aus der Division von P(X) durch (X-*a*) entsteht, den Wert *a* und den Wert von P(*a*) zurück. Mit anderen Worten P(X) = Q(X)(X-a)+P(a). Zum Beispiel: HORNER('X^3+2*X^2-

3*X+1',2) = {'X^2+4*X+5', 2, 11}. Wir könnten somit schreiben, dass $X^3+2X^2-3X+1 = (X^2+4X+5)(X-2)+11$. Ein weiteres Beispiel: HORNER('X^6-1',-5)= {'X^5-5*X^4+25*X^3-125*X^2+625*X-3125',-5, 15624} d. h., $X^6-1 = (X^5-5*X^4+25X^3-125X^2+625X-3125)(X+5)+15624$.

Die Variable VX

Im Verzeichnis {HOME CASDIR} gibt es eine Variable mit dem Namen VX , welche standardmäßig den Wert 'X' annimmt. Dies ist der Name der bevorzugten unabhängigen Variablen für algebraische und Infinitesimalrechnungsanwendungen. Vermeiden Sie, den Variablennamen VX in Ihren Programmen oder Gleichungen zu verwenden, um eine Verwechslung mit der CAS-Variablen VX zu vermeiden. Wenn Sie sich jedoch auf die x-Komponente der Geschwindigkeit beziehen möchten, können Sie dafür entweder vx oder Vx benutzen. Zusätzliche Informationen zu CAS-Variablen finden Sie in Anhang C.

Die Funktion LAGRANGE

Die Funktion LAGRANGE benötigt als Eingabe eine Matrix mit zwei Reihen und *n* Zeilen. Die Matrix speichert Datenpunkte in Form von $[[x_1,x_2, \ldots, x_n] [y_1, y_2, \ldots, y_n]]$. Die Funktion LAGRANGE erzeugt ein erweitertes Polynom aus

$$p_{n-1}(x) = \sum_{j=1}^{n} \frac{\prod_{k=1,k\neq j}^{n}(x - x_k)}{\prod_{k=1,k\neq j}^{n}(x_j - x_k)} \cdot y_j.$$

So können wir z. B. für n = 2 schreiben:

$$p_1(x) = \frac{x - x_2}{x_1 - x_2} \cdot y_1 + \frac{x - x_1}{x_2 - x_1} \cdot y_2 = \frac{(y_1 - y_2)\cdot x + (y_2 \cdot x_1 - y_1 \cdot x_2)}{x_1 - x_2}$$

Überprüfen Sie dieses Ergebnis mit Ihrem Taschenrechner:
LAGRANGE([[x1,x2],[y1,y2]]) = '((y1-y2)*X+(y2*x1-y1*x2))/(x1-x2)'.

Weitere Beispiele: LAGRANGE([[1, 2, 3][2, 8, 15]]) = '(X^2+9*X-6)/2'
LAGRANGE([[0,5;1,5;2,5;3,5;4,5][12,2;13,5;19,2;27,3;32,5]]) =
'-(,1375*X^4+ -,7666666666667*X^3+ - ,74375*X^2 =

1,991666666667*X-12,92265625)'.

Anmerkung: Matrizen werden in Kapitel 10 eingeführt.

Die Funktion LCM

Die Funktion LCM (Least Common Multiple – kleinstes gemeinsames Vielfaches) berechnet das kleinste gemeinsame Vielfache zweier Polynome oder Listen von Polynomen der gleichen Länge. Beispiele:

LCM('2*X^2+4*X+2' ,'X^2-1') = '(2*X^2+4*X+2)*(X-1)'.
LCM('X^3-1','X^2+2*X') = '(X^3-1)*(X^2+2*X)'

Die Funktion LEGENDRE

Ein Legendre-Polynom n-ten Grades ist eine Polynom-Funktion, die die Differentialgleichung $(1-x^2)\cdot\frac{d^2y}{dx^2}-2\cdot x\cdot\frac{dy}{dx}+n\cdot(n+1)\cdot y=0$ löst.

Um den n-ten Grad des Legendre-Polynoms zu erhalten, verwenden Sie LEGENDRE(*n*), wie z. B.

LEGENDRE(3) = '(5*X^3-3*X)/2'
LEGENDRE(5) = '(63*X ^5-70*X^3+15*X)/8'

Die Funktion PCOEF

Wenn wir ein Array mit den Wurzeln des Polynoms haben, erzeugt die Funktion PCOEF ein Array, das die Koeffizienten der entsprechenden Polynome enthält. Die Koeffizienten entsprechen in abfallender Reihenfolge des Grades der unabhängigen Variablen. So zum Beispiel die Gleichung PCOEF([-2,–1,0,1,1,2]) = [1. –1. –5. 5. 4. –4. 0.], welche das Polynom X^6-X^5-5X^4+5X^3+4X^2-4X darstellt.

Die Funktion PROOT

Wenn wir ein Array der Koeffizienten eines Polynoms in abfallender Reihenfolge haben, erzeugt die Funktion PROOT die Wurzeln des Polynoms. Beispielsweise gilt für X^2+5X-6 =0 PROOT([1 –5 6]) = [2. 3.].

Die Funktion PTAYL

Wenn wir ein Polynom P(X) und eine Zahl *a* haben, ergibt die Funktion PTAYL einen Ausdruck Q(X-*a*) = P(X), d. h. es erzeugt ein Polynom als Potenz von (X-*a*). Dies ist auch als Taylor-Polynom bekannt, von welchem auch der Name der Funktion abgeleitet wurde, Polynom & TAYLor, folgen.

So z. B. ergibt PTAYL('X^3-2*X+2',2) = 'X^3+6*X^2+10*X+6'.

Eigentlich sollten Sie dieses Ergebnis so interpretieren

'(X-2) ^3+6*(X-2) ^2+10*(X-2) +6'.

Wir können es überprüfen, indem wir die Substitution benutzen: 'X = x – 2'. Wir stellen das ursprüngliche Polynom wieder her, aber als kleingeschriebenes x anstelle eines großgeschriebenen.

Die Funktionen QUOT und REMAINDER

Die Funktionen QUOT und REMAINDER geben entsprechend den Quotienten (Q(X) und den Restwert R(X), die aus der Division der zwei Polynome $P_1(X)$ und $P_2(X)$ resultieren, wieder. Sie geben also die Werte aus Q(X) und R(X) aus $P_1(X)/P_2(X) = Q(X) + R(X)/P_2(X)$. Beispiel:

QUOT(X^3-2*X+2, X-1) = X^2+X-1
REMAINDER(X^3-2*X+2, X-1) = 1.

Somit können wir schreiben: $(X^3-2X+2)/(X-1) = X^2+X-1 + 1/(X-1)$.

Anmerkung: Sie könnten letzteres Ergebnis auch für PROPFRAC erhalten:
PROPFRAC('(X^3-2*X+2)/(X-1)') = 'X^2+X-1 + 1/(X-1)'.

Die Funktion EPSX0 und die CAS-Variable EPS

Die Variable ε (Epsilon) wird normalerweise in mathematischen Lehrbüchern, zur Darstellung einer sehr kleinen Zahl verwendet. Verwenden Sie die Funktion EPSX0, wird im CAS des Taschenrechners eine Variable EPS mit dem Standardwert 0,0000000001 = 10^{-10} erzeugt. Sobald diese Variable erzeugt wurde, können Sie ihren Wert in einen von Ihnen gewünschten Wert für EPS.

ändern. Wird die Funktion EPSX0 auf ein Polynom angewendet, werden alle Koeffizienten, deren absoluter Wert kleiner als EPS ist, mit Null ersetzt. Die Funktion EPSX0 ist im ARITHMETIC-Menü nicht enthalten, sondern kann nur über die Funktion catalog (N) gestartet werden. Beispiel:

EPSX0('X^3-1,2E-12*X^2+1,2E-6*X+6,2E-11)=
'X^3-0*X^2+,0000012*X+0'.

Mit EVAL: 'X^3+,0000012*X'.

Die Funktion PEVAL

Die Funktion PEVAL (Polynomial EVALuation – Auswertung des Polynoms) kann zur Auswertung eines Polynoms $p(x) = a_n \cdot x^n + a_{n-1} \cdot x^{n-1} + \ldots + a_2 \cdot x^2 + a_1 \cdot x + a_0$, verwendet werden, wenn [a_n, a_{n-1}, … a_2, a_1, a_0] ein Array von Koeffizienten ist und a den Wert von x_0 darstellt. Das Ergebnis ist die Auswertung von $p(x_0)$. Die Funktion PEVAL ist im ARITHMETIC-Menü nicht enthalten, sondern kann nur über die Funktion catalog (N) gestartet werden. Beispiel:

PEVAL([1,5,6,1],5) = 281.

Die Funktion TCHEBYCHEFF

Die Funktion TCHEBYCHEFF(*n*) erzeugt das Tchebycheff-(oder Chebyshev-) Polynom der ersten Art, Grad n, definiert als $T_n(X) = \cos(n \cdot \arccos(X))$. Ist die Integer-Zahl n negativ (n < 0), erzeugt die Funktion TCHEBYCHEFF(*n*) ein Chebyshev-Polynom der zweiten Art, Grad n, definiert als $T_n(X) = \sin(n \cdot \arccos(X))/\sin(\arccos(X))$. Beispiele:

TCHEBYCHEFF(3) = 4*X^3-3*X
TCHEBYCHEFF(-3) = 4*X^2-1

Brüche

Brüche können mit den Funktionen EXPAND und FACTOR, aus dem Menü ALG (‚×) erweitert bzw. faktoriert werden. Beispiel:

EXPAND('(1+X)^3/((X-1)(X+3))') = '(X^3+3*X^2+3*X+1)/(X^2+2*X-3)'
EXPAND('(X^2*(X+Y)/(2*X-X^2)^2') = '(X+Y)/(X^2-4*X+4)'

EXPAND('X*(X+Y)/(X^2-1)') = '(X^2+Y*X)/(X^2-1)'
EXPAND('4+2*(X-1)+3/((X-2)*(X+3))-5/X^2') =
'(2*X^5+4*X^4-10*X^3-14*X^2-5*X)/(X^4+X^3-6*X^2)'

FACTOR('(3*X^3-2*X^2)/(X^2-5*X+6)') = 'X^2*(3*X-2)/((X-2)*(X-3))'
FACTOR('(X^3-9*X)/(X^2-5*X+6)') = 'X*(X+3)/(X-2)'
FACTOR('(X^2-1)/(X^3*Y-Y)') = '(X+1)/((X^2+X+1)*Y)'

Die Funktion SIMP2

Die Funktionen SIMP2 und PROPFRAC werden zur Vereinfachung von Brüchen bzw. Erzeugung eines reinen Bruches verwendet. Die Funktion SIMP2 benötigt als Argument zwei Zahlen oder Polynome, welche den Zähler und Nenner eines rationellen Bruches darstellen und gibt den vereinfachten Zähler und Nenner für sie zurück. Beispiel: SIMP2('X^3-1','X^2-4*X+3') = { 'X^2+X+1','X-3'}.

Die Funktion PROPFRAC

Die Funktion PROPFRAC konvertiert einen rationellen Bruch in einen "reinen" Bruch, d. h. einem Bruchteil wird ein Integer-Wert hinzugefügt, falls eine derartige Zerlegung möglich ist. Beispiel:

PROPFRAC('5/4') = '1+1/4'
PROPFRAC('(x^2+1)/x^2') = '1+1/x^2'

Die Funktion PARTFRAC

Die Funktion PARTFRAC zerlegt einen rationellen Bruch in Teilbrüche, die zusammen den ursprünglichen Bruch ergeben. Beispiel:

PARTFRAC('(2*X^6-14*X^5+29*X^4-37*X^3+41*X^2-16*X+5)/(X^5-7*X^4+11*X^3-7*X^2+10*X)') =
'2*X+(1/2/(X-2)+5/(X-5)+1/2/X+X/(X^2+1))'

Diese Technik ist besonders bei der Berechnung von Integralen (siehe Kapitel über Infinitesimalrechnung) mit rationellen Brüchen von Nutzen.

Wenn Sie den Complex-Modus aktiviert haben, sieht das Ergebnis wie folgt aus:

‘2*X+(1/2/(X+i)+1/2/(X-2)+5/(X-5)+1/2/X+1/2/(X-i))’

Die Funktion FCOEF

Die Funktion FCOEF erzeugt einen rationellen Bruch, wenn die Wurzeln und Pole des Bruches bekannt sind.

Anmerkung: Wenn wir einen rationellen Bruch F(X) = N(X)/D(X) haben, können die Wurzeln dieses Bruches mit der Gleichung N(X) = 0 und die Pole über D(X) = 0 berechnet werden.

Die Eingabe für die Funktion bildet einen Vektor von Wurzeln, gefolgt von deren Mehrwertigkeit (d. h. wie oft kann eine Wurzel wiederholt werden) und die Pole, gefolgt von deren Mehrwertigkeit als negative Zahl dargestellt. So z. B.,. wenn wir einen Bruch erstellen möchten, dessen Wurzeln zwei Mehrwertigkeiten 1,0 mit Mehrwertigkeit 3 und -5 mit Mehrwertigkeit 2, enthalten, die Pole 1 mit Mehrwertigkeit 2 und -3 mit Mehrwertigkeit 5, gehen Sie wie folgt vor:

FCOEF([2 1 0 3 –5 2 1 -2 -3 -5]) = ‘(X--5)^2*X^3*(X-2)/(X--3)^5*(X-1)^2’

Drücken Sie EVAL erhalten Sie:

‘(X^6+8*X^5+5*X^4-50*X^3)/(X^7+13*X^6+61*X^5+105*X^4-45*X^3-297*X^2-81*X+243)’

Die Funktion FROOTS

Die Funktion FROOTS erzeugt die Wurzeln und Pole eines Bruches. Wenn wir z. B. die Funktion FROOTS auf obiges Ergebnis anwenden, erhalten wir [1 –2. –3 –5. 0 3. 2 1. –5 2.]. Das Ergebnis enthält Pole, gefolgt von deren Mehrwertigkeit als negative Zahl und Wurzeln, gefolgt von deren Mehrwertigkeit in Form einer positiven Zahl. In diesem Fall sind die Pole (1, -3) mit entsprechender Mehrwertigkeit (2,5) und die Wurzeln (0, 2, -5) mit der entsprechenden Mehrwertigkeit (3, 1, 2).

Ein weiteres Beispiel lautet: FROOTS('(X^2-5*X+6)/(X^5-X^2)')= [0 –2,1 –1,3 1,2 1]., d. h., Pole = 0 (2), 1(1) und Wurzeln = 3(1), 2(1). Befinden Sie sich im Complex-Modus, sieht ihr Ergebnis wie folgt aus: [0 –2,1 –1 '-((1+i*√3) /2' –1. '-((1-i*√3)/2' –1.].

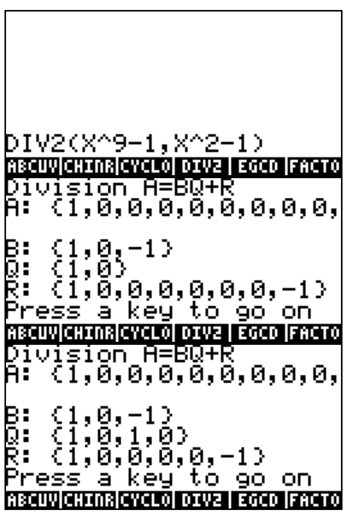
Step-by-Step Operationen mit Polynomen und Brüchen

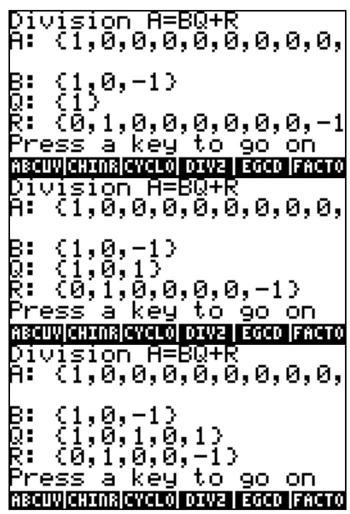
Stellen Sie das CAS auf Step/step, wird der Taschenrechner schrittweise Vereinfachungen von Brüchen und Operationen mit Polynomen anzeigen. Dies ist besonders bei der synthetischen Division nützlich, um die einzelnen Schritte der Division zu sehen. Das Beispiel der Division

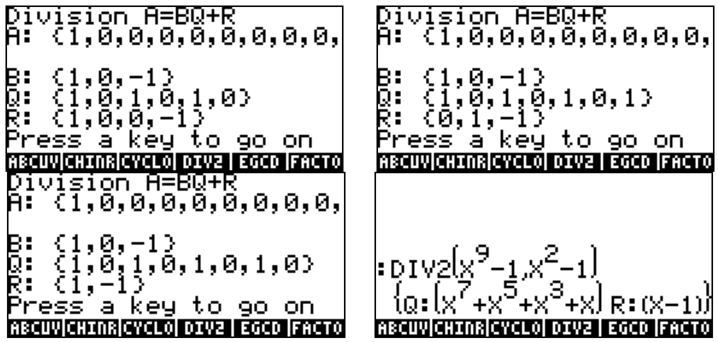
$$\frac{X^3 - 5X^2 + 3X - 2}{X - 2}$$

wird ausführlich in Anhang C erläutert. In nachfolgendem Beispiel wird eine längere synthetische Division angezeigt:

$$\frac{X^9 - 1}{X^2 - 1}$$







Das Menü CONVERT und algebraische Operationen

Das Menü CONVERT wird über die Tasten ← CONVERT (die Taste 6) gestartet. Das Menü zeigt alle Umwandlungs-Menüs im Taschenrechner an. Nachstehend finden Sie eine Abbildung mit der Liste der Menüs:

Die in den einzelnen Untermenüs vorhandenen Funktionen werden nachfolgend besprochen.

Menü Konvertierung von UNITS (Einheiten) (Option 1)

Dieses Menü entspricht dem Menü UNITS unter Verwendung von → UNITS . Die Anwendungen dieses Menüs werden ausführlich in Kapitel 3 erläutert.

Konvertierungs-Menü BASE (Option 2)

Dieses Menü entspricht dem Menü BASE unter Verwendung von → BASE . Die Anwendungen dieses Menüs werden ausführlich in Kapitel 19 erläutert.

Konvertierungs-Menü TRIGONOMETRIC (Option 3)

Dieses Menü entspricht dem Menü TRIG unter Verwendung von → TRIG . Die Anwendungen dieses Menüs werden ausführlich in diesem Kapitel erläutert.

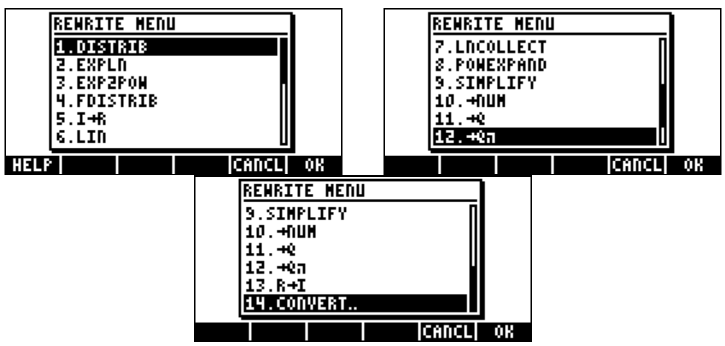
Konvertierungs-Menü MATRIZEN (Option 5)

Dieses Menü enthält zusätzlich die folgenden Funktionen:

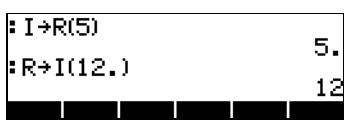
Diese Funktionen werden ausführlich in Kapitel 10 erläutert.

Konvertierungs-Menü REWRITE (Option 4)

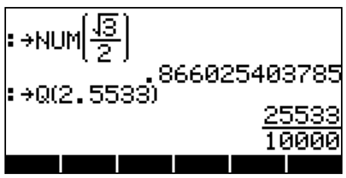
Dieses Menü enthält zusätzlich die folgenden Funktionen:

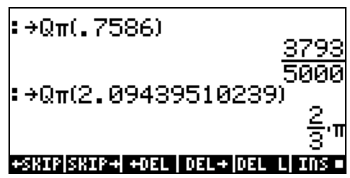


Die Funktionen I→R und R→I werden zur Konvertierung einer Integer- (I) in eine reelle Zahl (R) oder umgekehrt verwendet. Integer-Zahlen werden ohne Dezimalpunkte angegeben, während reelle Zahlen Integer-Werte darstellen und einen Dezimalpunkt am Ende besitzen, z. B.



Die Funktion →NUM hat den gleichen Effekt wie die Tastenkombination →NUM (der Taste ENTER zugeordnet). Die Funktion →NUM konvertiert ein symbolisches Ergebnis in ein Gleitpunkt-Ergebnis. Die Funktion →Q konvertiert einen Gleitpunktwert in einen Bruch. Die Funktion →Qπ konvertiert einen Gleitpunktwert in einen Bruch von π, wenn ein Bruch von π, für die Zahl gefunden werden kann; andernfalls konvertiert sie diese Zahl in einen Bruch. Nachfolgend einige Anwendungsbeispiele dieser Funktionen.

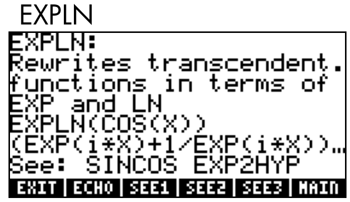




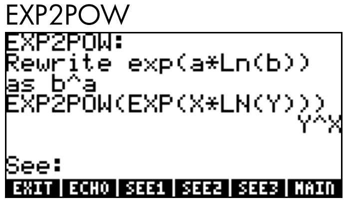
Aus den Funktionen des Menüs REWRITE stehen die Funktionen DISTRIB, EXPLN, EXP2POW, FDISTRIB, LIN, LNCOLLECT, POWEREXPAND und SIMPLIFY für algebraische Ausdrücke zur Verfügung. Viele dieser Funktionen werden in diesem Kapitel vorgestellt. Aus Gründen der Vollständigkeit zeigen wir die Einträge in der Hilfefunktion für diese Funktionen.

DISTRIB

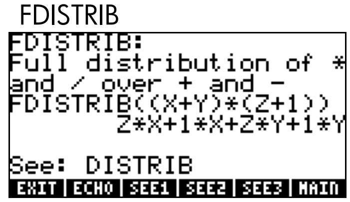
EXPLN



EXP2POW

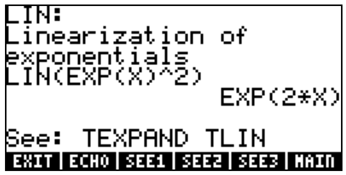


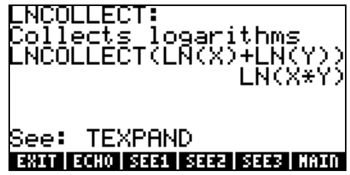
FDISTRIB



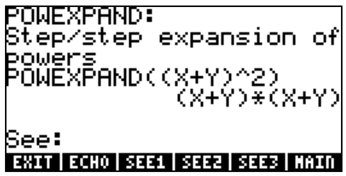
LIN

LNCOLLECT





POWEREXPAND



SIMPLIFY

Kapitel 6
Lösung für Einzelgleichungen

In diesem Kapitel behandeln wir die Funktionen des Taschenrechners zur Lösung von Einzelgleichungen der Form f(X) = 0. Der Taste 7 sind zwei Menüs für die Lösung von Gleichungen zugewiesen, der symbolische SOLVer (Löser) (← S.SLV) und der NUMerische SOLVer (Löser) (→ NUM.SLV). Nachfolgend werden einige Funktionen aus diesen Menüs beschrieben. Ändern Sie für diese Beispiele den CAS-Modus auf Complex (siehe Kapitel 2).

Symbolische Lösung algebraischer Gleichungen

Nachfolgend werden einige Funktionen aus dem Menü Solver beschrieben. Aktivieren Sie das Menü über die Tastenkombination. Mit dem Systemflag 117 auf *CHOOSE* boxes gesetzt, werden folgende Menüeinträge aufgelistet:

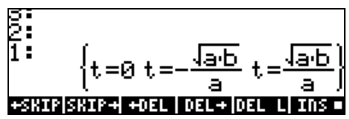
Die Funktionen DESOLVE und LDEC werden zur Lösung von Differentialgleichungen, Thema eines anderen Kapitels, verwendet und deshalb in den folgenden Abschnitten nicht näher erläutert. Ähnlich die Funktion LINSOLVE, die zur Lösung von mehrfachen linearen Gleichungen dient und ebenfalls in einem anderen Kapitel dargelegt wird. Die Funktionen ISOL und SOLVE können zur Lösung der Unbekannten in einer Polynomgleichung verwendet werden. Die Funktion SOLVEX löst eine Polynomgleichung, in der die Standard-CAS-Variable VX (standardmäßig 'X') die Unbekannte ist. Schließlich die Funktion ZEROS, die Nullen, Wurzeln oder ein Polynom bereitstellt. Für alle Funktionen außer ISOL sind Einträge im S.SLV-Menü über die CAS-Hilfefunktion (TOOL NXT HELP) verfügbar.

Funktion ISOL

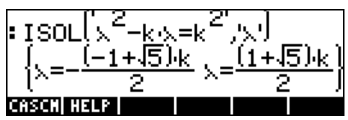
Mit der Funktion ISOL (Gleichung, Variable) erhalten Sie die Lösung(en) für *Gleichung* durch Isolierung der *Variablen*. Um beispielsweise *t* in der

Gleichung at^3-bt = 0 zu ermitteln, wenn der Taschenrechner im ALG-Modus ist, können wir wie folgt vorgehen:

Im RPN-Modus erhalten wir das gleiche Ergebnis, wenn wir die Gleichung, gefolgt von der Variablen, in den Stack schreiben und anschließend die Funktion ISOL eingeben. Bevor Sie die Funktion ISOL ausführen, sollte die Anzeige im RPN-Modus, wie in der Abbildung auf der linken Seite, aussehen. Nachdem Sie die Funktion ISOL ausgeführt haben, sieht ihre Anzeige wie in der rechten Abbildung aus:

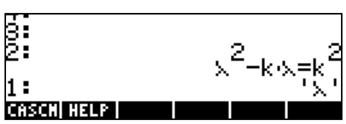


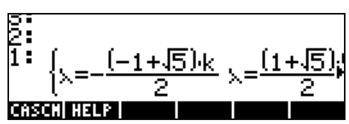
Das erste Argument in ISOL kann ein Ausdruck – wie oben aufgeführt – oder eine Gleichung sein. Versuchen Sie z. B. im ALG-Modus:



Anmerkung: Um das Gleichzeichen (=) in einer Gleichung zu schreiben, verwenden Sie die Tastenfolge ↱ = (der Taste +/- zugeordnet).

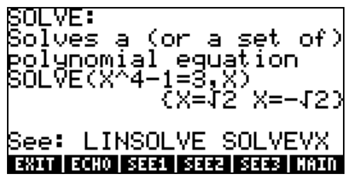
Das gleiche Problem kann wie unten angezeigt im RPN-Modus gelöst werden (Abbildungen zeigen den RPN-Stack vor und nach der Anwendung der Funktion ISOL):



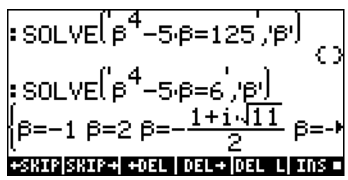


Funktion SOLVE

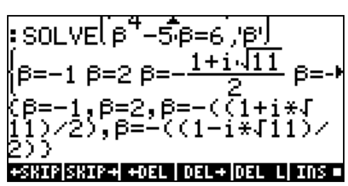
Die Funktion SOLVE hat die gleiche Syntax wie die Funktion ISOL, nur dass SOLVE auch zur Lösung einer Menge von Polynomgleichungen verwendet werden kann. In der Abbildung unten finden Sie den Hilfetext für die Funktion SOLVE, mit der Lösung der Gleichung X^4 – 1 = 3:



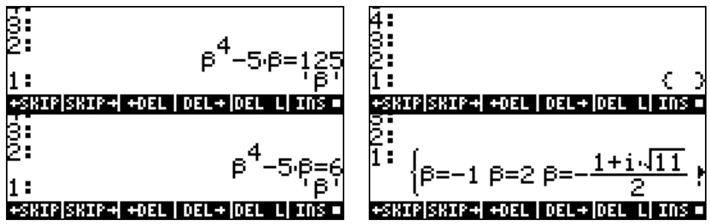
Die folgenden Beispiele zeigen die Funktion SOLVE im ALG- und im RPN-Modus:



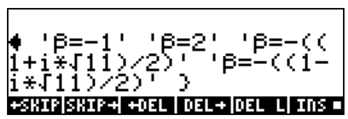
Die Abbildung zeigt zwei Lösungen. In der ersten, β^4-5β =125, findet SOLVE keine Lösungen { }. In der zweiten Abbildung hingegen, β^4 - 5β = 6, findet SOLVE gleich vier Lösungen, die in der letzten Ausgabezeile angezeigt sind. Die letzte Lösung ist nicht sichtbar, weil die Anzahl der Buchstaben der Lösung größer als die Breite der Anzeige des Displays ist. Sie können jedoch mithilfe der Pfeiltaste (▽), welche von einer Zeile des Zeileneditors in die andere umschaltet, alle Lösungen ansehen (dieser Vorgang kann jederzeit benutzt werden, wenn die Ausgabezeile länger als die Breite der Taschenrechner-Displays ist):



Die entsprechende Anzeige für diese beiden Beispiele im RPN-Modus ist nachstehend vor und nach der Anwendung der Funktion SOLVE zu sehen:

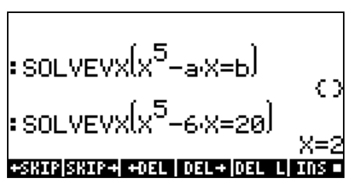


Benutzen Sie in diesem Modus die Pfeiltaste ▽, wird der Zeileneditor gestartet:



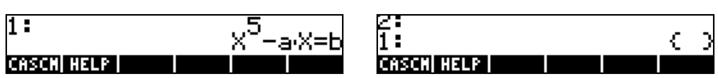
Funktion SOLVEVX

Die Funktion SOLVEVX löst eine Gleichung für die Standard-CAS-Variable in der reservierten Variablen VX. Standardmäßig ist der Wert dieser Variablen 'X'. Nachfolgende Beispiele, im ALG-Modus mit VX = 'X':

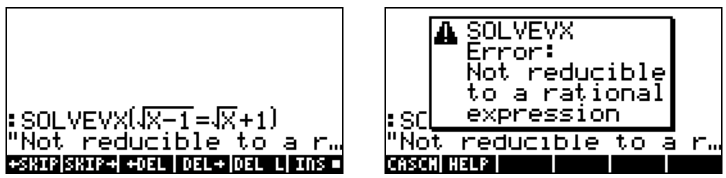


Im ersten Fall konnte SOLVEVX keine Lösung finden. Im zweiten Fall hat SOLVEVX eine einzige Lösung gefunden, X = 2.

Nachfolgend die Anzeige der beiden Beispiele im RPN-Stack (vor und nach Anwendung der Funktion SOLVEVX):

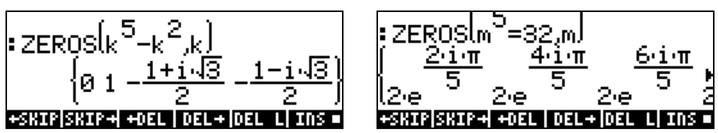


Wird die Gleichung als Argument für die Funktion SOLVEVX benutzt, muss diese auf einen rationalen Ausdruck vereinfacht werden können. So z. B. wird die nachfolgende Gleichung von SOLVEVX nicht verarbeitet:

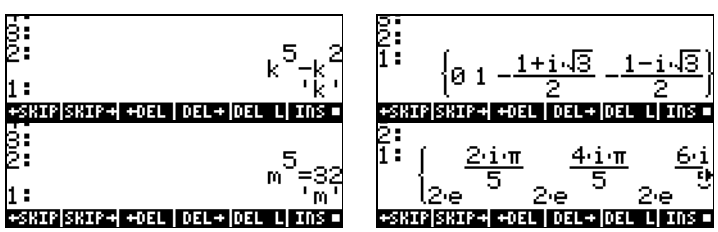


Funktion ZEROS

Die Funktion ZEROS berechnet Lösungen einer Polynomgleichung, ohne deren Mehrwertigkeit anzuzeigen. Als Eingabe für die Funktion wird der Ausdruck für die Gleichung und der Name der Variablen, die zu lösen ist, benötigt. Die folgenden Beispiele werden im ALG-Modus durchgeführt:



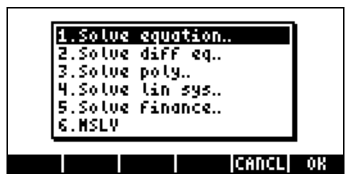
Um die Funktion ZEROS im RPN-Modus zu verwenden, muss zuerst der Polynomausdruck eingegeben werden, dann die zu lösende Variable, anschließend dann die Funktion ZEROS. Die nachfolgenden Abbildungen zeigen den RPN-Stack vor und nach Anwendung der Funktion ZEROS auf die beiden obigen Beispiele.



Die Funktionen des oben aufgeführten symbolischen Lösers ermitteln Lösungen für rationale Gleichungen (hauptsächlich für Polynomgleichungen). Wenn alle Koeffizienten der zu lösenden Gleichung numerisch sind, ist auch eine numerische Lösung über den numerischen Löser des Taschenrechners möglich.

Menü numerischer Löser

Der Taschenrechner bietet eine starke Umgebung zur Lösung von einzelnen algebraischen oder transzendenten Gleichungen. Um auf diese Lösung zuzugreifen, starten sie den numerischen Löser (NUM.SLV) mithilfe von ↱ NUM.SLV . Sie erhalten Ein Drop-Down-Menü mit folgenden Optionen:



Die Position *2. Solve diff eq..* wird in einem späteren Kapitel unter Differentialgleichungen näher behandelt. Position *4. Solve lin sys..* wird in einem späteren Kapitel über Matrizen behandelt. Position *6. MSLV* (Mehrfacher SoLVer (Gleichungslöser)) wird im nächsten Kapitel erläutert. Nachfolgend präsentieren wir Anwendungen zu den Positionen *3. Solve poly..*, *5. Solve finance* und *1. Solve equation..* (in dieser Reihenfolge). In Anhang 1-A am Ende von Kapitel 1 finden Sie Anleitungen zur Benutzung von Eingabeformularen und Beispiele für Anwendungen mit dem numerischen Löser.

Anmerkungen:

1. Wenn Sie eine Lösung in der NUM.SLV Anwendung berechnen, wird der gefundene Wert in den Stack geschrieben. Dies erweist sich als nützlich, wenn sie diesen Wert für spätere Operationen benötigen.
2. Bei jedem Start einer Anwendung im NUM.SLV-Menü werden eine oder mehrere Variablen erzeugt.

Polynomgleichungen

Wenn Sie die Option *Solve poly…* in der SOLVE Umgebung Ihres Taschenrechners benutzen, können Sie:

(1) Lösungen zu einer Polynomgleichung finden,
(2) die Koeffizienten des Polynoms, mit einer bekannten Anzahl von Wurzeln ermitteln, sowie
(3) einen algebraischen Ausdruck für das Polynom als Funktion von X ermitteln.

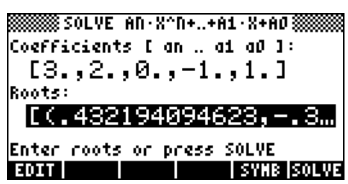
Lösungen zu einer Polynomgleichung berechnen

Eine Polynomgleichung ist eine Gleichung mit folgender Struktur: $a_nx^n + a_{n-1}x^{n-1} + \ldots + a_1x + a_0 = 0$. Der fundamentale Lehrsatz der Algebra besagt, dass es *n* Lösungen zu jeder Polynomgleichung *n*-ten Grades gibt. Dennoch können einige Lösungen aber auch komplexe Zahlen sein. Als Beispiel lösen Sie die Gleichung: $3s^4 + 2s^3 - s + 1 = 0$.

Wir möchten die Koeffizienten der Gleichung in einen Vektor $[a_n, a_{n-1}, a_1\ a_0]$ setzen. Für dieses Beispiel benutzen wir den Vektor [3,2,0,-1,1]. Um diese Polynomgleichung mit dem Taschenrechner zu lösen, versuchen Sie Folgendes:

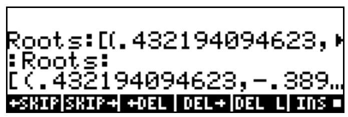
‌→ NUM.SLV ▼ ▼ OK	Wählen Sie Solve poly…
← [] 3 → , 2 → , 0 → , 1 +/- → , 1 OK	Tragen Sie die Koeffizienten in einen Vektor ein
SOLVE	Lösen der Gleichung

In der Anzeige wird die Lösung wie folgt aussehen:



Drücken Sie `ENTER`, um zum Stack zurückzukehren. Der Stack zeigt die folgenden Ergebnisse im ALG-Modus an (die gleichen Ergebnisse werden auch im RPN-Modus angezeigt):

Um alle Lösungen anzuzeigen, drücken Sie die Pfeiltaste (▼) zur Navigation im Zeileneditor:



Alle Lösungen sind komplexe Zahlen: (0,432;-0,389), (0,432;0,389), (-0,766; 0,632) , (-0,766; -0,632).

Anmerkung: Beachten Sie, dass komplexe Zahlen im Taschenrechner als geordnete Paare dargestellt werden, wobei die erste Zahl im Paar den reellen Teil und die zweite den imaginären Teil darstellt. So z. B. wird die Zahl (0,432,-0,389), eine komplexe Zahl, normalerweise als 0,432 – 0,389*i* dargestellt, wobei *i* den imaginären Teil, d. h. $i^2 = -1$ darstellt.

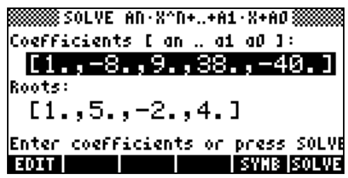
Anmerkung: Der fundamentale Lehrsatz der Algebra besagt, dass es *n* Lösungen zu jeder Polynomgleichung *n*-ten Grades gibt. Es gibt einen weiteren Lehrsatz in der Algebra, der besagt, dass, wenn eine Lösung einer Polynomgleichung mit reellen Koeffizienten eine komplexe Zahl ist, dann ist die konjugierte dieser Zahl auch eine Lösung. Mit anderen Worten, komplexe Lösungen zu einer Polynomgleichung mit reellen Koeffizienten treten paarweise auf. Dies bedeutet, dass Polynomgleichungen mit reellen Koeffizienten von ungeraden Zahlen mindestens eine reelle Lösung haben.

Erzeugen von Polynom-Koeffizienten , wenn die Wurzeln des Polynoms bekannt sind

Angenommen, Sie möchten ein Polynom erstellen, dessen Wurzeln die Zahlen [1, 5, -2, 4] sind. Um den Taschenrechner für diesen Zweck zu nutzen, führen Sie folgende Schritte aus:

↱ NUM.SLV ▼ ▼ OK	Wählen Sie Solve poly…
▼ ↰ [] 1 ↱ , 5 ↱ , 2 +/- ↱ , 4 OK	Tragen Sie die Wurzeln in einen Vektor ein
SOLVE	Lösen der Koeffizienten

Drücken Sie ENTER, um zum Stack zurückzukehren, die Koeffizienten werden im Stack angezeigt.



Drücken Sie ▼, um alle Koeffizienten im Zeileneditor anzuzeigen.

Anmerkung: Möchten Sie ein Polynom mit reellen Koeffizienten erhalten, dies aber komplexe Wurzeln besitzt, müssen Sie die komplexen Wurzeln als Paare von konjugierten Zahlen eingeben. Um dies zu veranschaulichen, erstellen wir ein Polynom mit den Wurzeln [1 (1,2) (1,-2)]. Überprüfen Sie, dass das erhaltene Polynom nur reelle Koeffizienten enthält. Versuchen Sie auch ein Polynom mit den Wurzeln [1 (1,2) (-1,2)] zu erstellen, und überprüfen Sie, dass das daraus resultierende Polynom komplexe Koeffizienten enthält.

Erstellen eines algebraischen Ausdrucks für das Polynom

Sie können bei der Erstellung eines algebraischen Ausdrucks für ein Polynom mit vorgegebenen Wurzelkoeffizienten den Taschenrechner benutzen. Der ermittelte Ausdruck wird als Standard-CAS-Variable X ausgegeben. (Die nachfolgenden Beispiele zeigen, wie Sie X mit einer anderen Variable über die Funktion |) ersetzen können.

Um den algebraischen Ausdruck mithilfe der Koeffizienten zu erstellen, nehmen Sie nachfolgendes Beispiel. Nehmen wir an, die Koeffizienten des Polynoms sind [1,5,-2,4]. Verwenden Sie dazu folgende Tastenfolge:

↱ NUM.SLV ▼ ▼ OK	Wählen Sie Solve poly…
↰ [] 1 ↱ , 5	Tragen Sie die Koeffizienten in einen Vektor ein
↱ , 2 +/- ↱ , 4 OK	
▲ SYMB	Erzeugen Sie den symbolischen Ausdruck
ENTER	Zurück zum Stack

Der so ermittelte Ausdruck wird im Stack wie folgt angezeigt:

'X^3+5*X^2+-2*X+4'.

Um den algebraischen Ausdruck mithilfe der Wurzeln zu erstellen, nehmen Sie folgendes Beispiel. Nehmen wir an, die Wurzeln des Polynoms lauten [1,3,-2,1]. Verwenden Sie dazu folgende Tastenfolge:

↱ NUM.SLV ▼ ▼ OK	Wählen Sie Solve poly…
▼ ↰ [] 1 ↱ , 3	Tragen Sie die Wurzeln in einen Vektor ein
↱ , 2 +/- ↱ , 1 OK	
▼ SYMB	Erzeugen Sie den symbolischen Ausdruck
ENTER	Zurück zum Stack

Der so erzeugte Ausdruck wird im Stack wie folgt angezeigt:

'(X-1)*(X-3)*(X+2)*(X-1)'.

Um die Ergebnisse zu erweitern, können Sie den Befehl EXPAND verwenden. Der daraus resultierende Ausdruck sieht wie folgt aus:

'X^4+-3*X^3+ -3*X^2+11*X-6'.

Ein weiterer Ansatz, einen Ausdruck für das Polynom zu bekommen, besteht darin, zunächst die Koeffizienten zu erzeugen, und anschließend den algebraischen Ausdruck mit hervorgehobenen Koeffizienten. Versuchen Sie in diesem Fall:

NUM.SLV OK	Wählen Sie Solve poly…
[] 1 , 3	Tragen Sie die Wurzeln in einen Vektor ein
, 2 +/- , 1 OK	
SOLVE	Solve (Lösung) für Koeffizienten
SYMB	Erzeugen Sie den symbolischen Ausdruck
ENTER	Zurück zum Stack

Der so ermittelte Ausdruck wird im Stack wie folgt angezeigt: 'X^4+-3*X^3+ -3*X^2+11*X+-6*X^0'. Die Koeffizienten werden in Stack-Ebene 2 angezeigt.

Finanzmathematische Berechnungen

Die Berechnungen in Position *5. Solve finance..* im numerischen Löser (*NUM.SLV*) werden zur Berechnung des Zeitwertes von Geld, von besonderer Bedeutung in der technischen Wirtschaft und anderer Finanzanwendungen verwendet. Diese Anwendung kann mit der Tastenkombination S FINANCE (der Taste 9 zugeordnet) gestartet werden. Bevor wir diese Lösungsumgebung im Detail schildern, präsentieren wir einige Definitionen, welche zum Verständnis finanzmathematischer Operationen im Taschenrechner erforderlich sind.

Definitionen

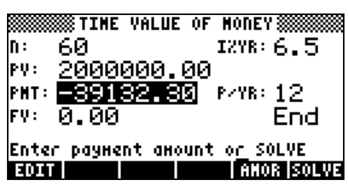
Es kommt häufig vor, dass man zur Entwicklung eines Projektes Geld von einem Finanzinstitut oder aus öffentlichen Mitteln leihen muss. Die geliehene Geldsumme wird als *aktueller Wert* (PV) bezeichnet. Dieser Geldbetrag muss in *n* Raten (Zeitabschnitten) (normalerweise Vielfache oder Untereinheiten eines Monats) zurückgezahlt werden Die Raten unterliegen einer jährlichen *Zinsrate* von I%YR. Die *Anzahl von Zeitabschnitten pro Jahr* (P/YR) ist eine Integer-Zahl von Zeitabschnitten, in welches das Jahr zum Zwecke der Zurückzahlung des Kredits unterteilt wurde. Standardwerte von P/YR sind 12 (eine Zahlung pro Monat), 24 (zwei Zahlungen pro Monat) oder 52 (wöchentliche Zahlungen). Die *payment*(PMT) (monatliche Zahlung) ist die Summe, die der Leiher an den Verleiher am Anfang oder am Ende jedes

Zeitabschnittes *n* der Leihfrist bezahlen muss. Der *zukünftige Wert* des Geldes (FV) ist der Wert der ausgeliehenen Geldsumme am Ende von *n* Zeitschabschnitten. Normalerweise erfolgt die Bezahlung jeweils am Ende eines Zeitabschnittes, sodass der Entleiher am End des ersten Zeitabschnittes mit der Bezahlung beginnt und die gleiche feste Summe am Ende des zweiten, dritten usw. Zeitabschnittes bis hin zum letzten Zeitabschnitt *n* bezahlt.

Beispiel 1 – Berechnung der Rückzahlung eines Kredits
Wenn 2 Millionen Dollar bei einem jährlichen Zinssatz von 6,5% mit einer Leihfrist von 60 Monaten ausgeliehen werden, wie viel beträgt die monatliche Ratenzahlung? Um die Schulden vollständig innerhalb von 60 Monaten zu bezahlen, sollten die zukünftigen Werte des Darlehens Null betragen. Um dies über die finanzmathematischen Merkmale des Taschenrechners durchzuführen, benutzen wir die nachfolgenden Werte: n = 60, I%YR = 6,5, PV = 2000000, FV = 0, P/YR = 12. Um diese Daten einzugeben und die Zahlung, PMT, zu berechnen, gehen Sie wie folgt vor:

← FINANCE	Starten Sie die Eingabemaske für Finanzmathematik
60 OK	Geben Sie n = 60 ein
6,5 OK	Geben Sie I%YR = 6,5 % ein
2000000 OK	Geben Sie PV = 2.000.000 US$ ein
▼	Übergehen Sie PMT, da Sie dies berechnen möchten
0 OK	Geben Sie FV = 0 ein, die Option End wird hervorgehoben
▲ ◀ SOLVE	Heben Sie PMT hervor, um diesen Wert zu ermitteln

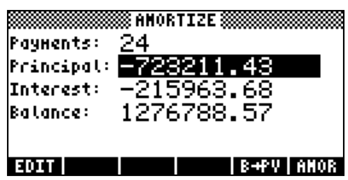
Die Anzeige sieht wie folgt aus:



In der Anzeige erscheint der Wert für PMT als –39.132.30, d. h. der Kreditnehmer wird eine monatliche Rate von US $ 39.132,30 am Ende jedes

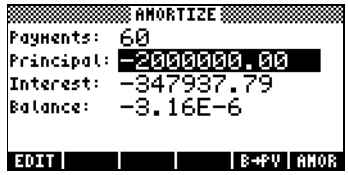
Monats innerhalb der kommenden 60 Monate zahlen, um den Gesamtbetrag zurückzuzahlen. Der Grund, warum der Wert PMT negativ ausgefallen ist, besteht darin, dass der Taschenrechner die Werte aus der Sicht des Kreditnehmers betrachtet. Der Kreditnehmer besitzt ein Plus von US $ 2.000.000,00 in der Zeitspanne t = 0, dann beginnt er mit der Zahlung, sodass jedes Mal – US $ 39.132,30 in den Perioden t = 1, 2, ..., 60 hinzuaddiert werden. Bei t = 60 liegt der tatsächliche Nettowert des Entleihers bei Null. Wenn Sie nun den Betrag von US $ 39.132,30 nehmen und diesen mit 60 Zahlungen multiplizieren, wird die tatsächlich zurückgezahlte Gesamtsumme US $ 2.347.937,79 betragen. Somit macht der Verleiher einen Nettogewinn von $ 347.937,79 in den 5 Jahren, in welchen er das Projekt des Kreditnehmers finanziert.

Beispiel 2 – Berechnung der Amortisation für einen Kredit
Sie erhalten die gleiche Lösung zu dem Problem aus Beispiel 1, wenn Sie die Taste AMOR, welche für AMORTISATION steht, drücken. Diese Option wird benutzt, um zu ermitteln, wie viel am Ende einer bestimmten Leihfrist insgesamt bezahlt (amortisiert) wurde. Nehmen wir an, dass wir 24 Leihfristen in der ersten Zeile der Amortisationsanzeige verwenden, d. h. 2 4 OK. Drücken Sie anschließend AMOR. Sie erhalten folgendes Ergebnis:



Diese Anzeige wird so interpretiert, dass nach 24 Monaten Schuldenrückzahlungen der Kreditnehmer insgesamt US $ 723.211,43 in seinen Gesamtschuldenbetrag eingezahlt und er US $ 215.963,68 an Zinsen bezahlt hat. Der Kreditnehmer muss noch eine Differenz von US $ 1.276.788,57 innerhalb der nächsten 36 Monate zurückzahlen.

Überprüfen Sie, was passiert, wenn Sie 60 bei den Zahlungen eingeben: starten Sie die Amortisationsanzeige drücken Sie dann OK AMOR. Die Anzeige sieht nun wie folgt aus:



Das bedeutet, dass am Ende von 60 Monaten der entliehene Betrag von US $ 2.000.000,00 zusammen mit den Zinsen von US $ 347.937,79 abbezahlt wurde, der Differenzbetrag aber noch US $ 0,000316 beträgt, welche der Kreditnehmer dem Verleiher schuldet, ist. Sicherlich sollte der Differenzbetrag aber Null sein. Der im Display angezeigte Wert ist ein Rundungsfehler aus der numerischen Lösung.

Drücken Sie ON oder ENTER zweimal, um zur Normalansicht des Taschenrechners zurückzukehren.

Beispiel 3 – Berechnen der Zahlung am Anfang der Zeitspanne
Lösen wir das gleiche Problem wie in den Beispielen 1 und 2, jedoch mit der Option, dass die Zahlung am Anfang des gleichen Abschnitts stattfindet. Verwenden Sie:

← FINANCE	Starten Sie die Eingabemaske für Finanzmathematik
60 OK	Geben Sie n = 60 ein
6,5 OK	Geben Sie I%YR = 6,5 % ein
2000000 OK	Geben Sie PV = 2.000.000 US$ ein
▼	Übergehen Sie PMT, da wir diese berechnen
möchten	
0 OK	Geben Sie FV = 0 ein, die Option End wird hervorgehoben
CHOOS ▲ OK	Ändern Sie die Option für die Zahlung auf *Begin (Anfang)*
▲ ◄ SOLVE	Heben Sie PMT hervor, um diesen Wert zu lösen

In der Anzeige erscheint der Wert für PMT als –38.921,47, d. h. der Kreditnehmer wird eine monatliche Rate von US $ 38.921,48 am <u>Anfang</u> jedes Monats während der kommenden 60 Monate zahlen, bis er den

Gesamtbetrag zurückgezahlt hat. Beachten Sie, dass der Betrag den der Kreditnehmer monatlich zu bezahlen hat, wenn er diesen am Anfang jeden Monats bezahlt, geringfügig niedriger als am Ende des gleichen Monats ist. Der Grund dafür ist, dass der Verleiher Zinsguthaben für die Bezahlungen am Anfang des Monats bekommt, und somit die Schuldlast des Kreditnehmers etwas verringert.

Anmerkungen:

1. Die finanzmathematische Umgebung erlaubt es, jeden enthaltenen Wert, d. h. n, I%YR, PV, FV, P/Y, wenn die anderen Werte der Ausleihe bekannt sind, zu berechnen. Heben Sie einfach den Wert, den Sie berechnen möchten, hervor und drücken SOLVE. Das Ergebnis wird im hervorgehobenen Feld angezeigt.

2. Die in der finanzmathematischen Umgebung des Taschenrechners berechneten Werte werden mit ihrer entsprechenden Kennung (Kennzeichen zur Identifizierung) in den Stack kopiert.

Löschen von Variablen

Wenn Sie die finanzmathematische Umgebung Ihres Taschenrechners zum ersten Mal verwenden, werden im HOME-Verzeichnis oder in einem anderen Unterverzeichnis die Variablen n I%YR PV PMT PYR FV erzeugt, um die entsprechenden Faktoren für die verschiedenen Berechnungen zu speichern. Sie können sich den Inhalt dieser Variablen wie folgt anzeigen lassen:

→ n → I%YR → PV → PMT → PYR → FV.

Sie können sie für zukünftige Verwendung im Verzeichnis belassen oder mithilfe der Funktion PURGE aus dem Verzeichnis löschen. <u>Um im ALG-Modus alle Variablen auf einmal zu löschen</u>, versuchen Sie Folgendes:

TOOL PURGE VAR ← { }	Geben Sie PURGE ein und erstellen Sie eine Liste der Variablen
' → n	Geben Sie den Namen der Variablen N ein
▶ → ,	Geben Sie ein Komma ein

' → I%YR	Geben Sie den Namen der Variablen I%YR ein
▶ → ,	Geben Sie ein Komma ein
' → PV	Geben Sie den Namen der Variablen PV ein
▶ → ,	Geben Sie ein Komma ein
' → PMT	Geben Sie den Namen der Variablen PMT ein
▶ → ,	Geben Sie ein Komma ein
' → PYR	Geben Sie den Namen der Variablen PYR ein
▶ → ,	Geben Sie ein Komma ein
' → FV.	Geben Sie den Namen der Variablen FV ein
ENTER	Führen Sie den PURGE-Befehl aus

In den nachfolgenden Abbildungen sehen Sie den PURGE-Befehl zum Löschen aller Variablen im Verzeichnis sowie das Ergebnis, nachdem Sie den Befehl ausgeführt haben.

Denselben Wert berechnen Sie im RPN-Modus wie folgt:

VAR ← { }	Erstellen Sie eine Liste von Variablen, die gelöscht werden sollen
N	Geben Sie den Namen der Variablen N ein
I%YR	Geben Sie den Namen der Variablen I%YR ein
PV	Geben Sie den Namen der Variablen PV ein
PMT	Geben Sie den Namen der Variablen PMT ein
PYR	Geben Sie den Namen der Variablen PYR ein
FV	Geben Sie den Namen der Variablen FV ein

ENTER — Geben Sie die Liste der Variablen in den Stack

TOOL PURGE — Löschen Sie die Variablen in der Liste

Bevor Sie den PURGE-Befehl eingeben, sieht der RPN-Stack wie folgt aus:

Lösen von Gleichungen mit einer Unbekannten über NUM.SLV

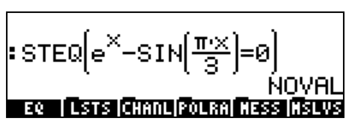
Das Menü NUM.SLV des Taschenrechners bietet in Position *1. Solve equation..* die Lösung verschiedener Typen von Gleichungen in einer einzigen Variablen, einschließlich nicht-linearer algebraischer und transzendenter Gleichungen. Als Beispiel lösen wir die Gleichung e^x-$sin(\pi x/3) = 0$.

Geben Sie den Ausdruck als algebraisches Objekt ein, und speichern Sie es in der Variablen EQ. Die dazu erforderlichen Tastenfolgen im ALG-Modus lauten:

' ← e^x ALPHA ← X ▶ — SIN ← π
× ALPHA ← X ÷ 3 ▶ → = 0 ▶
STO▶ ALPHA E ALPHA Q ENTER

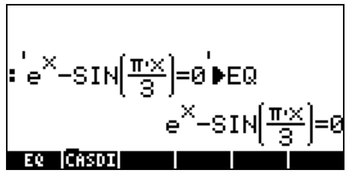
Funktion STEQ

Die Funktion STEQ, die über den Befehls-Katalog (→ CAT) gestartet wird, speichert ein Argument in einer Variablen EQ, z. B. im ALG-Modus:

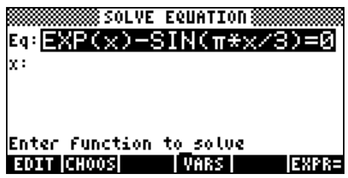


Im RPN-Modus tragen Sie die Gleichung zwischen zwei Apostrophen ein und starten den Befehl STEQ. Auf diese Weise kann die Funktion STEQ als Kürzel zur Speicherung des Ausdrucks in der Variablen EQ verwendet werden.

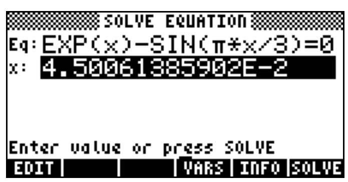
Drücken Sie VAR, um die neu erstellte Variable EQ anzuzeigen:



Wechseln Sie anschließend in die SOLVE-Umgebung, und wählen Sie *Solve equation…* unter Verwendung der Tastenfolge: ↱ NUM.SLV OK. Die entsprechende Anzeige sieht wie folgt aus:



Die Gleichung, die wir gerade in der Variablen EQ gespeichert haben, ist bereits im Feld *Eq* in der Eingabemaske SOLVE EQUATION geladen. Auch ein mit x beschriftetes Feld wird bereitgestellt. Um die Gleichung zu lösen, müssen Sie einfach nur noch das Feld vor dem X markieren, indem Sie die Pfeiltaste ▽ benutzen und dann SOLVE drücken. Die angezeigte Lösung ist X: 4,5006E-2:



Dies ist jedoch nicht die einzig mögliche Lösung für diese Gleichung. Um z. B. eine negative Lösung zu erhalten, tragen Sie, bevor Sie die Gleichung lösen, eine negative Zahl in das Feld X: ein. Versuchen Sie 3 +/- OK ▽ SOLVE. Die Lösung lautet nun -3,045.

Lösungsschema für die *Gleichung Solve…*

Der numerische Löser für Gleichungen mit einer Unbekannten funktioniert wie folgt:

- Erlaubt dem Anwender, die Gleichung einzugeben oder den Typ der zu lösenden Gleichung zu wählen CHOOS.
- Erzeugt eine Eingabemaske mit Eingabefeldern für alle in der Gleichung vorkommenden Variablen, die in der Variablen EQ gespeichert sind.
- Der Anwender muss die Werte aller vorkommenden Variablen eingeben bis auf eine, die ermittelt werden soll.
- Anschließend markiert der Anwender das Feld mit der Unbekannten, für welche die Gleichung gelöst werden soll und drückt dann SOLVE
- Der Anwender kann eine Lösung erzwingen, indem er eine Schätzung der Lösung im entsprechenden Eingabefeld, bevor er die Gleichung löst, vorgibt.

Der Taschenrechner benutzt einen Suchalgorithmus, um ein Intervall festzulegen, für welches die Funktion das Vorzeichen ändert, was darauf hinweist, dass es eine Wurzel oder Lösung für die Gleichung gibt. Er verwendet anschließend eine numerische Methode, um einen Näherungswert für die Lösung zu ermitteln.

Die Lösung, die der Taschenrechner sucht, wird durch den vorhandenen Anfangswert im Feld der Unbekannten bestimmt. Ist kein Wert vorhanden, benutzt der Taschenrechner den Standardwert Null. Somit können Sie mehr als nur eine Lösung zu einer Gleichung suchen, indem Sie diesen Anfangswert im Feld der Unbekannten ändern. Beispiele zu Lösungen für die Gleichung werden nachfolgend gezeigt.

Beispiel 1 – Hookes Gesetz für Dehnung und Spannung

Die zu verwendende Gleichung ist Hookes Gesetz für normale Dehnung in x-Richtung eines Feststoffteilchens, welches einer Spannung, gemäß nachfolgender Abbildung, unterliegt

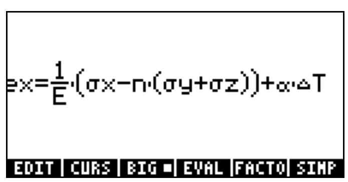
$$\begin{bmatrix} \sigma_{xx} & \sigma_{xy} & \sigma_{xz} \\ \sigma_{yx} & \sigma_{yy} & \sigma_{yz} \\ \sigma_{zx} & \sigma_{zy} & \sigma_{zz} \end{bmatrix}$$

Die Gleichung lautet $e_{xx} = \frac{1}{E}[\sigma_{xx} - n \cdot (\sigma_{yy} + \sigma_{zz})] + \alpha \cdot \Delta T,$ wobei e_{xx} die Einheit der Dehnung in x-Richtung, σ_{xx}, σ_{yy} und σ_{zz}, die normale Spannung auf die Teilchen in Richtung der Achsen x, y und z, *E* die Youngs Elastizitätsmodul des Materials, *n* das Poisson-Verhältnis des Materials, α der thermische Dehnungskoeffizient des Materials und ΔT der Temperaturanstieg ist.

Angenommen, Sie haben folgende Daten: σ_{xx}= 2500 psi, σ_{yy} =1200 psi, und σ_{zz} = 500 psi, E = 1200000 psi, *n* = 0.15, α = 0,00001/°F, ΔT = 60 °F. Um die Dehnung e_{xx} zu berechnen, gehen Sie wie folgt vor:

→ NUM.SLV OK	Starten Sie den numerischen Löser, um die Gleichung zu lösen
→ EQW	Starten Sie den EquationWriter, um die Gleichung einzugeben

An dieser Stelle befolgen Sie die Anweisungen aus Kapitel 2, Verwendung des EquationWriters zur Erstellung einer Gleichung. Die Gleichung, die Sie ins Feld *Eq* eingeben, sollte so aussehen (beachten Sie, dass wir nur einen Unterindex benutzen, um auf die Variablen hinzuweisen, d. h. e_{xx} wird als *ex* übertragen, usw. -- dies wird gemacht, um Zeit beim Eintippen zu sparen):

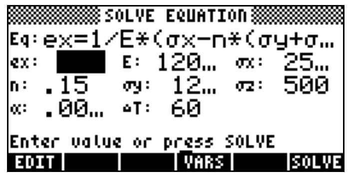


Benutzen Sie folgende Abkürzungen für Sonderzeichen:

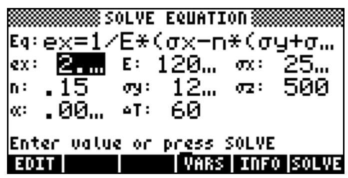
σ: ALPHA → S α: ALPHA → A Δ: ALPHA → C

beachten Sie dabei, dass die Kleinbuchstaben mit ALPHA ← vor dem Buchstaben eingegeben werden, somit wird x als ALPHA ← X eingegeben.

Drücken Sie ENTER, um zum Löser zurückzukehren. Geben Sie die oben vorgeschlagenen Werte in die entsprechenden Felder ein, sodass die Anzeige im Löser wie folgt aussieht:



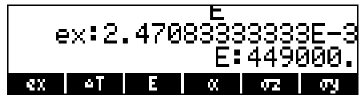
Mit hervorgehobenem *ex:* Feld, drücken Sie SOLVE, um die Lösung für *ex* zu ermitteln.



Drücken Sie EDIT, während das Feld *ex*: markiert ist, ist die Lösung in der SOLVE EQUATION Eingabemaske zu sehen. Das Ergebnis lautet 2,470833333333E-3. Drücken Sie OK, um EDIT (Bearbeitungsmodus) zu verlassen.

Angenommen Sie möchten nun das Young-Modul, welches die Dehnung von e_{xx} = 0,005 unter der gleichen Spannung erzeugt, wobei die thermische Ausdehnung unbeachtet bleibt, ermitteln. In diesem Fall, sollten Sie den Wert 0,005 in das Feld *ex*: eingeben und eine Null in das Feld *ΔT:* (ist *ΔT* = 0, werden keine thermischen Effekte berücksichtigt). Um E zu berechnen, markieren Sie das Feld *E*:, und drücken Sie SOLVE. Das Ergebnis, mit der EDIT Funktion angesehen, ist E ≈449000 psi. Drücken Sie SOLVE ENTER, um zur Normalansicht zurückzukehren.

Beachten Sie, dass die Ergebnisse der Berechnungen, die innerhalb des numerischen Lösers stattfinden, in den Stack kopiert wurden:

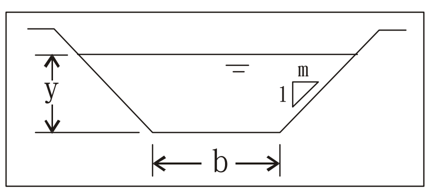


Auch werden Sie in Ihren Funktionstasten die Variablen, die denen, die Sie in der Gleichung EQ gespeichert haben, entsprechen, finden (drücken Sie NXT, um alle Variablen in Ihrem Verzeichnis anzuzeigen), d. h. die Variablen *ex, ΔT, α, σz, σy, n, σx* und *E*.

Beispiel 2 – Spezifische Leistung im offen Kanalsdurchfluss
Spezifische Leistung in einem offenen Kanaldurchfluss wird als Leistung pro gemessene Gewichtseinheit zum Kanalboden bezeichnet. Nehmen wir an E = spezifische Leistung, y = Kanaltiefe, V = Durchflussgeschwindigkeit,
g = Gravitätsbeschleunigung, dann schreiben wir

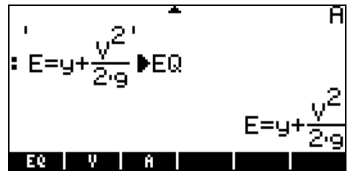
$$E = y + \frac{V^2}{2g}.$$

Die Durchflussgeschwindigkeit ist durch V = Q/A gegeben, wobei Q = Wasserabfluss und A = die Fläche des Querschnitts. Die Fläche ist abhängig vom verwendeten Querschnitt, z. B. für einen trapezförmigen Querschnitt, wie in Abbildung unten gezeigt, A = (b+m·y) ·y, wobei b = die Breite des Kanalbodens und m = Seitenwandneigung des Querschnitts.

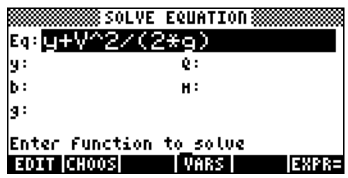


Wir können E wie oben gezeigt und zusätzlich die Variablen A und V in die Gleichung eingeben, sodass die Eingabemaske für die Grundvariablen y, Q, g, m und b wie nachfolgend gezeigt Eingabefelder zur Verfügung stellt:

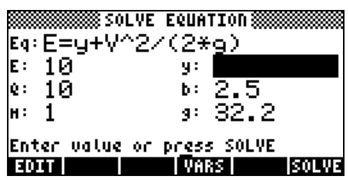
- Erstellen wir zuerst ein Unterverzeichnis SPEN (SPecific ENergy – spezifisiche Leistung) und arbeiten in diesem Unterverzeichnis.
- Als Nächstes definieren Sie folgende Variablen:



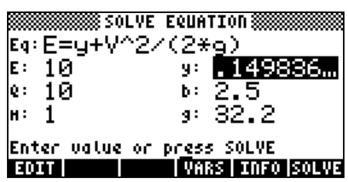
- Starten Sie den numerischen Löser, um die Gleichungen zu lösen: ↱ NUM.SLV OK. Beachten Sie, dass die Eingabemaske bereits Einträge für die Variablen y, Q, b, m und g enthält:
-



- Versuchen Sie folgende Eingabedaten: E = 10 ft, Q = 10 cfs (Kubikfuß pro Sekunde), b = 2,5 ft, m = 1,0, g = 32,2 ft/s^2:



- Lösen Sie die Gleichung für y.

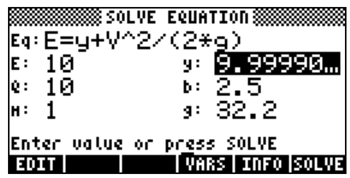


Das Ergebnis ist 0,149836.., d. h. y = 0,149836.

- Es ist jedoch bekannt, dass es für y in dieser Gleichung für spezifische Leistung eigentlich zwei Lösungen gibt. Die Lösung, die wir gerade ermittelt haben, entspricht einer numerischen Lösung mit einem

Ausgangswert von 0 (der voreingestellte Standardwert für y, d. h. wenn das Lösungsfeld leer ist, ist der Ausgangswert 0). Um die andere Lösung zu finden, müssen Sie einen größeren Wert für y eingeben, sagen wir 15, markieren anschließend das Eingabefeld y und lösen die Gleichung für y erneut:

•



Das Ergebnis ist 9,99990, d. h. y = 9,99990 ft.

Dieses Beispiel veranschaulicht die Anwendung von zusätzlichen Variablen zur Erstellung komplizierter Gleichungen. Sobald NUM.SLV aktiviert ist, werden die von den zusätzlichen Variablen implizierten Ersetzungen eingefügt und die Eingabemaske für die Gleichung stellt ein Eingabefeld für primitive oder fundamentale Variablen, die aus der Ersetzung resultieren, zur Verfügung. Das Beispiel veranschaulicht auch eine Gleichung, die mehr als eine Lösung hat und wie man den ersten Schätzwert für die Lösung wählt, welcher verschiedene Lösungen ermitteln kann.

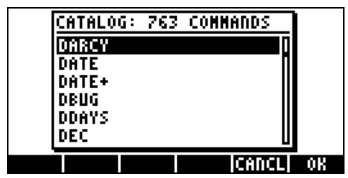
Im nächsten Beispiel werden wir die Funktion DARCY zur Berechnung des Reibungsfaktors in Rohrleitungen verwenden. Zunächst definieren wir die Funktion für unsere Zwecke.

Sonderfunktion für Rohrleitungsdurchfluss: DARCY (ε/D,Re)

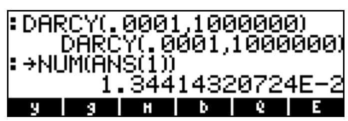
Die Darcy-Weisbach-Gleichung wird zur Berechnung des Leistungsverlustes (pro Gewichtseinheit) h_f bei einem Rohrleitungsdurchmesser D, einer absoluten Rauheit ε und Länge L verwendet, wenn die Durchflussgeschwindigkeit in der Leitung V ist. Die Gleichung lautet $h_f = f \cdot \frac{L}{D} \cdot \frac{V^2}{2g}$. Die Menge f wird als Reibungsfaktor des Durchflusses bezeichnet und wurde aus der Funktion der relativen Rauheit der Rohrleitung, ε/D und einer (dimensionslosen) Reynoldschen Zahl Re, berechnet. Die Reynoldsche Zahl ist als Re = $\rho VD/\mu$ =

VD/ν definiert, wobei ρ und μ die Dichte und die dynamische Viskosität der Flüssigkeit darstellen, während $\nu = \mu/\rho$ die kinematische Viskosität der Flüssigkeit darstellt.

Der Taschenrechner stellt eine Funktion mit dem Namen DARCY zur Verfügung, welche als Eingabe in dieser Reihenfolge die relative Rauheit ε/D und die Reynoldsche Zahl zur Berechnung des Reibungsfaktors f erhält. Die Funktion DARCY kann über den Befehlskatalog aufgerufen werden:



So können Sie beispielsweise für ε/D = 0,0001 und Re = 1000000 den Reibungsfaktor berechnen, indem Sie eingeben DARCY(0,0001,1000000). In der nachfolgenden Abbildung wurde die Funktion →NUM () zur Berechnung des numerischen Wertes der Funktion verwendet:

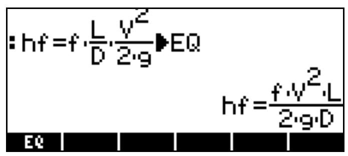


Das Ergebnis ist f = DARCY(0,0001,1000000) = 0,01341…

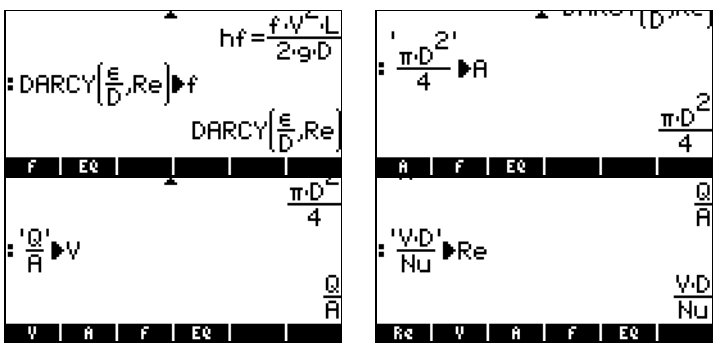
Die Funktion FANNING(ε/D,Re)

In aerodynamischen Anwendungen wird ein anderer Reibungsfaktor verwendet, der sogenannte Fanning-Reibungsfaktor. Der Fanning-Reibungsfaktor f_F, wird als der 4-fache Darcy-Weisbach-Reibungsfaktor *f* bezeichnet. Der Taschenrechner stellt eine Funktion FANNING zur Verfügung, welche die gleichen Eingaben wie die Funktion DARCY benutzt, d. h. ε/D und Re, und stellt den FANNING-Reibungsfaktor als Ergebnis zur Verfügung. Überprüfen Sie, ob FANNING(0,0001,1000000) = 0,0033603589181s, ergibt.

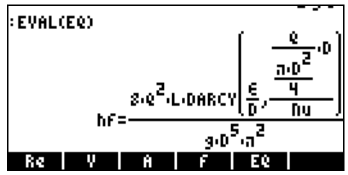
Beispiel 3 – Strömung in einem Rohr
Für die nachfolgenden Beispiele sollten Sie eine separates Unterverzeichnis (PIPES) erstellen. Die Hauptgleichung für die Strömung in einem Rohr, ist selbstverständlich die Darcy-Weisbach-Gleichung. Geben Sie also nachfolgende Gleichung in EQ ein:



Geben Sie genauso die folgenden Variablen (f, A, V, Re) ein:



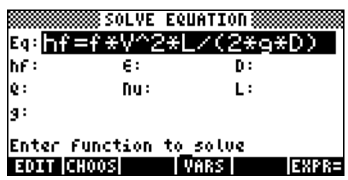
In diesem Fall haben wir die Hauptgleichung (Darcy-Weisbach-Gleichung) in EQ gespeichert und anschließend alle ihre Variablen durch andere Ausdrücke, über die Definition der Variablen f, A, V und Re, ausgetauscht. Um die kombinierte Gleichung zu sehen, benutzen Sie EVAL(EQ). In diesem Beispiel haben wir die Anzeige so geändert, dass die Gleichung vollständig im Display angezeigt wird:



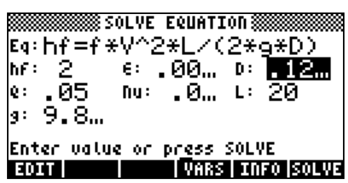
Somit lautet die zu lösende Gleichung, nachdem diese mit den verschiedenen Variablen aus dem Verzeichnis kombiniert wurde, wie folgt:

$$h_f = \frac{8Q^2L}{\pi^2 gD^5} \cdot DARCY\left(\frac{\varepsilon}{D}, \frac{\frac{QD}{\pi D^2/4}}{Nu}\right)$$

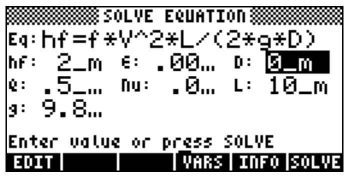
Die kombinierte Gleichung enthält die primitiven Variablen *h_f, Q, L, g, D, ε* und *Nu*.
Starten Sie den numerischen Löser (↱ NUM.SLV OK), um die in der SOLVE EQUATION-Eingabemaske vorhandenen primitiven Variablen anzuzeigen.

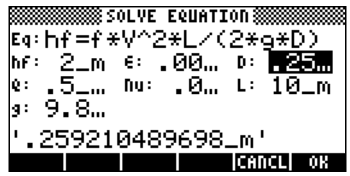


Angenommen, wir verwenden die Werte hf = 2 m, ε = 0,00001 m, Q = 0,05 m^3/s, Nu = 0,000001 m^2/s, L = 20 m und g = 9,806 m/s^2 und möchten den Durchmesser D berechnen. Tragen Sie die Eingabewerte ein und lösen Sie die Gleichung für D. Das Ergebnis lautet 0,12, d. h., D = 0,12 m.



Ist die Gleichung dimensional gesehen konsistent, können Sie Einheiten zu den Eingabewerten, wie in der Abbildung unten gezeigt, hinzufügen. Sie müssen diese Einheiten jedoch zu den ursprünglichen Schätzwerten in der Lösung hinzufügen. Im nachstehenden Beispiel fügen wir vor Lösung des Problems 0_m ins Feld D: ein. Die Lösung ist auf der rechten Abbildung zu sehen:



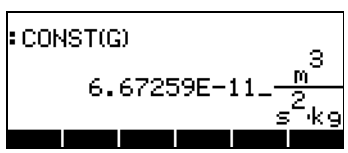


Drücken Sie die Taste ENTER, um zur Normalansicht des Taschenrechners zurückzukehren. Die Lösung für D wird im Stack angezeigt.

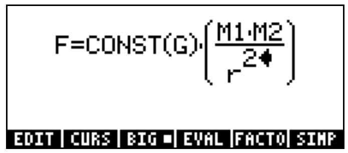
Beispiel 4 – Universelle Gravitation

Newtons Gesetz der universellen Gravitation besagt, dass die Magnitude der Anziehungskraft zweier Körper der Masse m_1 und m_2, die in einem Abstand r voneinander entfernt sind, über die Gleichung $F = G \cdot \frac{M_1 \cdot M_2}{r^2}$. dargestellt werden kann.

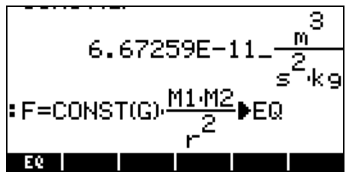
In diesem Fall ist G die universelle Gravitationskonstante, deren Wert man über die Funktion CONST im Taschenrechner erhalten kann:



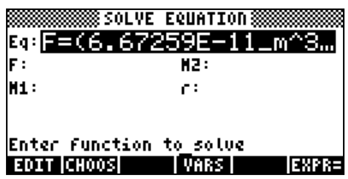
Wir können die Gleichung für jedes Glied, ausgenommen G, lösen, indem wir die Gleichung wie folgt eingeben:



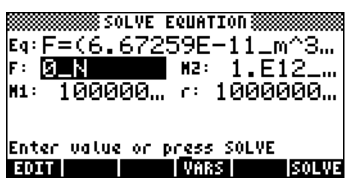
Diese Gleichung speichern wir dann in EQ:



Wenn Sie nun den numerischen Löser für diese Gleichung starten, erhalten Sie eine Eingabemaske mit den Eingabefeldern F, G, m1, m2 und r.



Lösen wir dieses Problem nun, indem wir verschiedene Einheiten für die bekannten Variablen einsetzen: m1 = $1{,}0\times10^{6}$ kg, m2 = $1{,}0\times10^{12}$ kg, r = $1{,}0\times10^{11}$ m. Geben Sie einen Wert von 0_N in Feld F ein, um sicher zu stellen, dass die Lösung mit den Einheiten des Taschenrechners richtig ausgewertet wird:

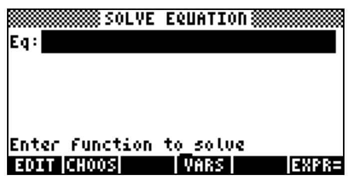


Lösen Sie die Gleichung für F, drücken Sie dann, um zur Normalansicht zurückzukehren. Die Lösung ist F: 6,67259E-15_N, oder F = $6{,}67259\times10^{-15}$ N.

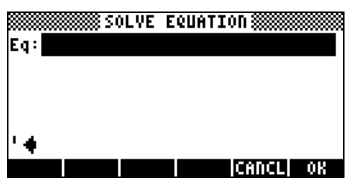
Anmerkung: Wenn Sie Einheiten im numerischen Löser benutzen, stellen Sie sicher, dass alle Variablen die richtigen Einheiten haben, dass diese kompatibel sind und dass die Gleichung dimensional gesehen homogen ist.

Unterschiedliche Wege Gleichungen in EQ einzugeben

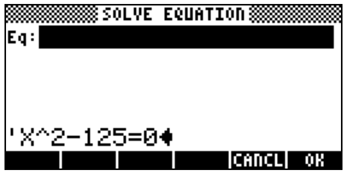
In all den gezeigten Beispielen haben wir die zu lösende Gleichung vor Aktivierung des numerischen Lösers direkt in die Variable EQ eingegeben. Eigentlich kann die zu lösende Gleichung direkt, nachdem Sie ihn gestartet haben, in den Löser eingegeben werden, indem Sie die Inhalte des Feldes EQ in der Eingabemaske des numerischen Lösers bearbeiten. Sollte die Variable EQ beim Start des numerischen Lösers (↱ NUM.SLV OK) nicht bereits definiert sein, wird das Feld EQ hervorgehoben:

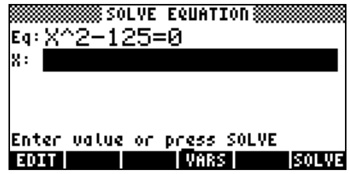


Sie können an dieser Stelle eine neue Gleichung über EDIT eingeben. Sie bekommen ein Apostroph-Paar, sodass Sie den Ausdruck dazwischen eingeben können:



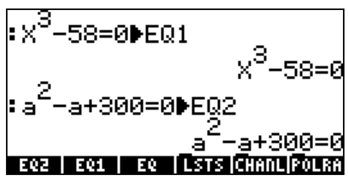
Geben Sie die Gleichung, beispielsweise X^2 - 125 = 0, direkt in den Stack ein, müssen Sie anschließend OK drücken.





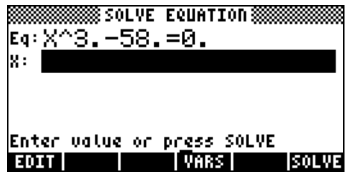
An dieser Stelle ist die Gleichung zur Lösung bereit.
Alternativ dazu können Sie zur Eingabe Ihrer Gleichung den EquationWriter starten, nachdem Sie EDIT gedrückt haben. Drücken Sie ENTER, um zum numerischen Löser zurückzukehren.

Eine weitere Möglichkeit, eine Gleichung in die Variable EQ einzugeben, ist eine bereits bestehende Variable, die in EQ eingegeben werden soll, aus dem Verzeichnis auszuwählen. Das bedeutet, dass Ihre Gleichung bereits in einer Variablen gespeichert sein muss. Nehmen wir z. B. an, dass wir die folgenden Gleichungen in die Variablen EQ1 und EQ2 bereits eingegeben haben:



Starten Sie nun den numerischen Löser (↱ NUM.SLV OK), und heben wir das Feld EQ hervor. Drücken Sie an dieser Stelle die Funktionstaste CHOOS, und benutzen Sie die Pfeiltasten (▲▼), um beispielsweise die Variable EQ1, auszuwählen:

Nachdem Sie die Variable EQ1 ausgewählt haben, drücken Sie OK, um die Variable EQ in den Löser zu laden. Die neue Gleichung steht nun zur Lösung bereit.

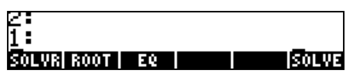


Das Funktionsmenü SOLVE

Über das Menü SOLVE kann über Funktionstasten auf einige der Funktionen des numerischen Lösers zugegriffen werden. Um in dieses Menü im RPN-Modus zu gelangen, verwenden Sie 74-MENU (bzw. im ALG-Modus MENU(74). Alternativ dazu können Sie jedoch auch die Tastenkombination ↱ (halten) 7 zum Starten des Menüs SOLVE benutzen. Die von SOLVE zur Verfügung gestellten Untermenüs lauten wie folgt:

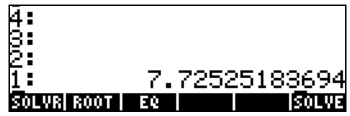
Das Untermenü ROOT

Im Untermenü ROOT sind folgende Funktionen und Untermenüs enthalten:

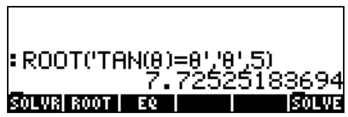


Die Funktion ROOT

Die Funktion ROOT wird zur Lösung einer Gleichung für eine gegebene Variable mit einem geschätzten Anfangswert verwendet. Im RPN-Modus befindet sich die Gleichung in Stack-Ebene 3, während der Variablenname in Ebene 2 zu finden ist und der geschätzte Anfangswert in Ebene 1. Die nachfolgende Abbildung zeigt den RPN-Stack vor und nach Aktivierung der Funktion ROOT:



Im ALG-Modus würden Sie zum Starten der Funktion ROOT wie folgt vorgehen: ROOT('TAN(θ)=θ','θ',5)



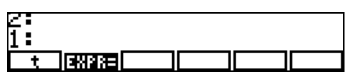
Variable EQ

Die Funktionstaste EQ in diesem Untermenü wird als Referenz auf die Variable EQ verwendet. Das Drücken der Funktionstaste ist gleichwertig mit dem Verwenden der Funktion RCEQ (ReCall EQ).

Das Untermenü SOLVER

Das Untermenü SOLVR startet das Funktionsmenü Löser für die aktuell in EQ gespeicherte Gleichung. Nachfolgend einige Beispiele:

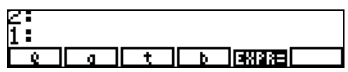
<u>Beispiel 1</u> - Lösen der Gleichung t^2-5t = -4
Wenn Sie beispielsweise die Gleichung 't^2-5*t=-4' in EQ speichern und anschließend SOLVR drücken, wird folgendes Menü gestartet:



Dieses Ergebnis zeigt an, dass Sie die Gleichung am oberen Rand des Displays für einen Wert t lösen können. Wenn Sie z. B. versuchen, ← [t] zu lösen, erhalten Sie den Wert t: 1, nachdem kurz die Meldung "Solving for t" (t lösen) aufblinkt. Es gibt jedoch eine weitere Wurzel für diese Gleichung, welche man durch Änderung des Wertes von t, bevor man dieses erneut berechnet, erhält. Gehen Sie wie folgt vor: 10 [t], und drücken Sie dann ← [t]. Die Lösung lautet nun t: 4,0000000003. Um dieses Ergebnis zu überprüfen, drücken Sie die Funktionstaste EXPR=, welche den Ausdruck in EQ mit dem aktuellen Wert von t berechnet. In diesem Fall lautet das Ergebnis:

Um die SOLVR Umgebung zu verlassen, drücken Sie VAR. An dieser Stelle haben Sie keinen Zugang zum Menü SOLVE, somit Sie müssen dieses erneut wie oben angezeigt, starten, um mit den nachfolgenden Beispielen fortzufahren.

<u>Beispiel 2</u> - Lösen der Gleichung Q = at^2+bt
In EQ kann auch eine Gleichung, die mehr als eine Variable enthält, gespeichert werden, beispielsweise 'Q = at^2 + bt'. Wenn Sie in diesem Fall das Funktionsmenü SOLVE gestartet haben und ROOT SOLVR drücken, erhalten Sie nachfolgende Anzeige:

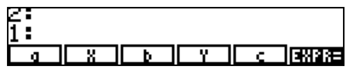


In dieser SOLVR Umgebung können Sie Werte für jede einzelne aufgelistete Variable festlegen, indem Sie diesen Wert einfach in den Stack eingeben und die entsprechende Funktionstaste drücken. Angenommen, wir möchten die Werte Q = 14, a = 2 und b = 3 eingeben. Geben Sie sie wie folgt ein: 14 [Q], 2 [a], 3 [b].

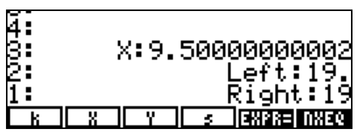
Da den Variablen Q, a und b numerische Werte zugewiesen werden, erscheinen die zugewiesenen Werte in der linken oberen Ecke des Displays. An dieser Stelle können wir t mithilfe von ↰ [t] lösen. Das Ergebnis ist t: 2. Drücken Sie nun EXPR=, erhalten Sie die Ergebnisse:

<u>Beispiel 3</u> - Zwei simultane Gleichungen lösen, jeweils eine auf einmal

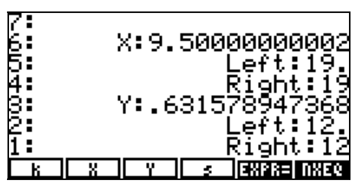
Sie können mehr als eine Gleichung lösen, indem Sie erst eine Gleichung lösen und den gleichen Vorgang solange wiederholen, bis eine Lösung gefunden wurde. So erhalten Sie z. B., wenn Sie nachfolgende Liste von Gleichungen in die Variable EQ eingeben – EQ: { 'a*X+b*Y = c', 'k*X*Y=s'}, mit der Tastenfolge ROOT SOLVE innerhalb des Funktionsmenüs SOLVE die folgende Anzeige:



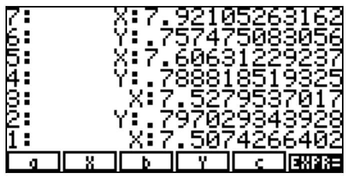
Die erste Gleichung, d. h. a*X + b*Y = c, wird im oberen Teil des Displays angezeigt. Sie können Werte für die Variablen a, b und c eingeben, beispielsweise 2 [a] 5 [b] 19 [c]. Da wir nur eine Gleichung auf einmal lösen können, geben wir einen geschätzten Anfangswert für Y, beispielsweise 0 [Y] ein und lösen die Gleichung für X, mit ← [X]. Dies ergibt den Wert X: 9,4999…. Um den Wert der Gleichung an dieser Stelle zu überprüfen, drücken Sie EXPR=. Die Ergebnisse sind: Left (links): 19, Right (rechts): 19. Um die nächste Gleichung zu lösen, drücken Sie NXT NXEQ. Im Display werden die Funktionstasten wie folgt angezeigt:

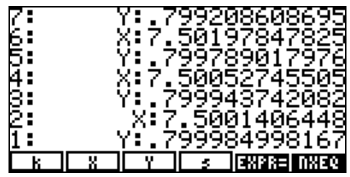


Geben wir z. B. die Werte k = 2, s = 12 ein. Lösen Sie die Gleichung dann für Y, und drücken Sie EXPR=. Die Ergebnisse sind nun wie folgt:



Dann fahren wir fort und navigieren zwischen den beiden Gleichungen vor und zurück, lösen die erste für X und die zweite für Y solange, bis X und Y zu einer Lösung führen. Um sich zwischen den Gleichungen hin und her zu bewegen, verwenden wir die Taste NXEQ. Um X und Y zu lösen, benutzen Sie entsprechend ⬅[X] und ⬅[Y]. Die folgende Sequenz von Lösungen wird erstellt:





Nachdem Sie nun die beiden Gleichungen gelöst haben, jeweils eine auf einmal, bemerken wir, dass X sich bis zur dritten Nachkommastelle dem Wert 7,500, während Y sich dem Wert 0,799 nähert.

Verwenden der Maßeinheiten mit dem SOLVR Unterprogramm

Diese sind einige Richtlinien auf dem Gebrauch von Maßeinheiten mit dem SOLVR Unterprogramm:

- Das Eintragen einer Vermutung mit Maßeinheiten für eine gegebene Variable, stellt den Gebrauch von jenen Maßeinheiten in der Lösung vor.
- Wenn eine neue Vermutung ohne Maßeinheiten gegeben wird, werden die Maßeinheiten, die vorher für die bestimmte Variable gespeichert werden, benutzt.
- Um Maßeinheiten zu entfernen tragen Sie eine Zahl ohne Maßeinheiten in eine Liste als die neue Vermutung ein, d.h. verwenden Sie das Format {Zahl}.
- Eine Liste von Zahlen kann als Vermutung für eine Variable gegeben werden. In diesem Fall nimmt die Maßeinheiten die benutzten Maßeinheiten gehören der letzten Zahl in der Liste. Z.B. zeigt das Hereinkommen {1.41_ft 1_cm 1_m} an, daß Meßinstrumente (M) für diese Variable benutzt werden.
- Der Ausdruck, der in der Lösung verwendet wird, muß konsistente Glieder haben, oder eine Störung resultiert beim Versuchen, für einen Wert zu lösen.

Das Untermenü DIFFE

Das Untermenü DIFFE enthält eine Reihe von Funktionen für die numerische Lösung von Differenzialgleichungen. Die zur Verfügung stehenden Funktionen sind:

Diese Funktionen werden im Detail in Kapitel 16 erläutert.

Das Untermenü POLY

Das Untermenü POLY führt Operationen mit Polynomen durch. Die enthaltenen Funktionen sind:

Funktion PROOT

Diese Funktion wird dazu verwendet, die Wurzeln eines Polynoms mit bekanntem Vektor mit den Koeffizienten des Polynoms in absteigender Reihenfolge der Potenz der unabhängigen Variable zu ermitteln. Mit anderen Worten, wenn das Polynom $a_nx^n + a_{n-1}x^{n-1} + \ldots + a_2x^2 + a_1x + a_0$, ist, sollte der Vektor von Koeffizienten als $[a_n, a_{n-1}, \ldots , a_2, a_1 , a_0]$ eingegeben werden. So sind z. B. die Wurzeln des Polynoms mit den Koeffizienten [1, -5, 6] die Werte [2, 3].

Funktion PCOEF

Diese Funktion erzeugt die Koeffizienten $[a_n, a_{n-1}, \ldots , a_2, a_1 , a_0]$ eines Polynoms $a_nx^n + a_{n-1}x^{n-1} + \ldots + a_2x^2 + a_1x + a_0$, mit bekannten Wurzeln eines Vektors $[r_1, r_2, \ldots, r_n]$. So ergibt z. B. ein Vektor, dessen Wurzeln [-1, 2, 2, 1, 0] sind, folgende Koeffizienten: [1, -4, 3, 4, -4, 0] Das Polynom lautet $x^5 - 4x^4 + 3x^3 + 4x^2 - 4x$.

Funktion PEVAL

Diese Funktion wertet ein Polynom mit bekanntem Vektor seiner Koeffizienten $[a_n, a_{n-1}, \ldots , a_2, a_1 , a_0]$ und einem Wert von x_0 aus, d. h., PEVAL berechnet $a_nx_0^n + a_{n-1}x_0^{n-1} + \ldots + a_2x_0^2 + a_1x_0 + a_0$. So ergibt z. B. die Funktion PEVAL für die Koeffizienten [2, 3, -1, 2] und einen Wert von 2 den Wert 28.

Das Untermenü SYS

Das Untermenü SYS enthält eine Auflistung von Funktionen zur Lösung linearer Systeme. Die in diesem Untermenü aufgelisteten Funktionen sind:

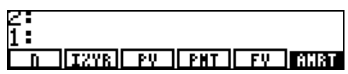
Diese Funktionen werden im Detail in Kapitel 11 erläutert.

Das Untermenü TVM

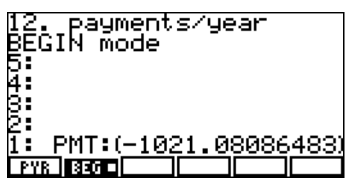
Das Untermenü TVM enthält Funktionen zur Berechnung des Geldzeitwertes (Time Value of Money). Dies ist eine alternative Möglichkeit, finanzmathematische Probleme zu lösen (siehe Kapitel 6). Die Funktionen werden nachfolgend angezeigt:

Das Untermenü SOLVR

Das Untermenü SOLVR aus dem Untermenü TVM startet den Löser zur Lösung von TVM-Problemen. Wenn Sie an dieser Stelle SOLVR drücken, erhalten Sie die nachfolgende Anzeige:



Als Beispiel versuchen wir es mit den Werten n = 10, I%YR = 5,6, PV = 10000 und FV = 0 und geben dann ⬅ [PMT] ein, um PMT = -1021,08…. zu berechnen. Wenn Sie nun NXT drücken, erscheint folgende Anzeige:



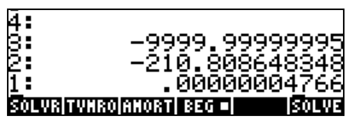
Drücken Sie VAR, um die SOLVR Umgebung zu verlassen. Suchen Sie den Weg zurück zum Untermenü TVM innerhalb des Untermenüs SOLVE, um auch die anderen vorhandenen Funktionen zu testen.

Funktion TVMROOT

Diese Funktion benötigt als Argument den Namen einer der Variablen im TVM-Problem. Die Funktion gibt die Lösung für diese Variable zurück, vorausgesetzt, die anderen Variablen existieren und ihre Werte wurden zuvor gespeichert. So können wir z. B., nachdem wir eines der obigen TVM-Probleme gelöst haben, beispielsweise für 'N' wie folgt lösen: [‘] ALPHA N ENTER TVMRO. Das Ergebnis ist 10.

Funktion AMORT

Diese Funktion nimmt einen Wert, der einen Zahlungszeitraum darstellt (zwischen 0 und n) an und gibt den Hauptwert, die Zinsrate und den Saldo für die zu dem Zeitpunkt gespeicherten TVM-Variablen zurück. Wenn wir z. B. mit den vorhin benutzten Daten die Funktion AMORT für einen Wert 10 starten, erhalten wir:



Funktion BEG

Wenn ausgewählt, werden die Berechnungen innerhalb der Funktion TMV an den Anfang jeder Zahlungsperiode gelegt. Wenn nicht ausgewählt, werden die Berechnungen innerhalb der Funktion TMV ans Ende jeder Zahlungsperiode gelegt.

Kapitel 7
Lösen von Mehrfachgleichungen

Viele wissenschaftliche und technische Probleme benötigen die gleichzeitige Lösung mehrerer Gleichungen. Der Taschenrechner stellt, wie unten gezeigt, mehrere Verfahrensweisen zur Lösung von Mehrfachgleichungen zur Verfügung. Beachten Sie, dass in diesem Kapitel keine Lösungen für Systeme mit linearen Gleichungen vorgestellt werden. Lösungen für lineare Systeme werden in einem späteren Kapitel über Matrizen und lineare Algebra ausführlich erklärt.

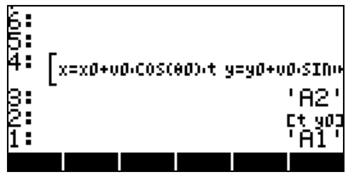
Rationelle Gleichungssysteme

Gleichungen, die sich in Polynome oder rationelle algebraische Ausdrücke umwandeln lassen, können mithilfe der Funktion SOLVE direkt mit dem Taschenrechner gelöst werden. Sie müssen die Liste der Gleichungen als Elemente eines Vektors zur Verfügung stellen. Die Liste der zu lösenden Variablen muss ebenfalls als Vektor zur Verfügung stehen. Bevor Sie versuchen, mit diesem Verfahren eine Lösung zu ermitteln, stellen Sie sicher, dass das CAS im Exakt-Modus ist. Auch sollten Sie beachten, dass das CAS zur Lösung eines angegebenen Systems von Gleichungen um so länger benötigt, je komplizierter ein Ausdruck ist. Beispiele für diese Anwendung finden Sie in den folgenden Abschnitten:

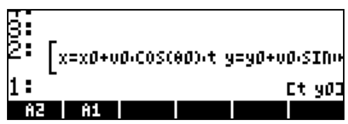
Beispiel 1 – Projektilbewegung

Verwenden Sie die Funktion SOLVE mit nachfolgenden Vektorargumenten, wobei das erste Argument die Liste der Gleichungen ['x = x0 + v0*COS(θ0)*t' 'y =y0+v0*SIN(θ0)*t – g*t^2/2'] ENTER darstellt, das zweite hingegeben die zu lösenden Variablen, beispielsweise t und y0, d. h., ['t' 'y0'].

In diesem Beispiel verwenden wir den RPN-Modus zur Lösungsfindung. Der Grund für dieses Vorgehen besteht darin, dass wir die Lösung Schritt für Schritt berechnen können. Im ALG-Modus ist die Lösung ziemlich ähnlich. Zuerst speichern wir den ersten Vektor (Gleichungen) in eine Variable A2 und den Vektor mit den Variablen in die Variable A1. Im RPN-Stack sieht die Anzeige, bevor Sie die Variablen gespeichert haben, wie folgt aus:



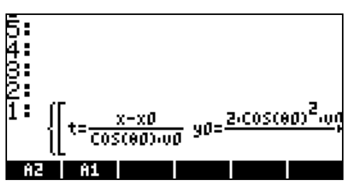
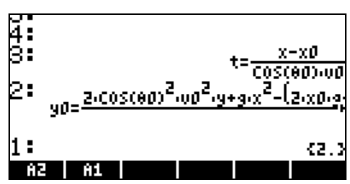
An dieser Stelle müssen wir nur noch STO▶ zweimal drücken, um die Variablen zu speichern.
Für die Lösungsfindung wechseln Sie das CAS in den Exakt-Modus, listen Sie dann die Inhalte der Variablen A2 und A1 – in dieser Reihenfolge – A2 A1 auf.



Verwenden Sie nun den Befehl SOLVE (aus dem Menü S.SLV: ← S.SLV).
Nach etwa 40 Sekunden erhalten Sie als Ergebnis eine Liste:

{ ‘t = (x-x0)/(COS(θ0)*v0)’
‘y0 = (2*COS(θ0)^2*v0^2*y+(g*x^2(2*x0*g+2*SIN(0))*COS(θ0)*v0^2)*x+
(x0^2*g+2*SIN(θ0)*COS(θ0)*v0^2*x0)))/(2*COS(θ0)^2*v0^2)’]}

Drücken Sie EVAL, um den Vektor aus der Liste zu entfernen, und verwenden Sie anschließend den Befehl OBJ→, um die Gleichungen separat im Stack aufzulisten.

Anmerkung: Diese Methode hat in diesem Beispiel einwandfrei funktioniert, weil die Unbekannten t und y0 algebraische Ausdrücke der Gleichungen darstellten. Diese Methode funktioniert jedoch nicht, wenn wir versuchen θ0 zu lösen, weil θ0 Teil eines transzendenten Ausdrucks ist.

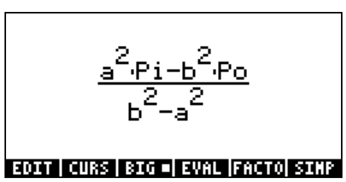
Beispiel 2 – Spannungen in einem dickwandigen Zylinder

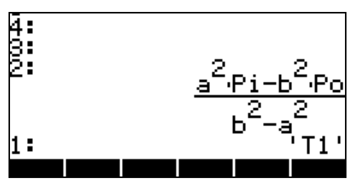
Nehmen wir an, wir haben einen dickwandigen Zylinder mit Innen- und Außendurchmesser *a* und *b*, welcher einem inneren Druck P_i und einem äußeren Druck P_o ausgesetzt ist. An jedem Radialabstand *r* von der Achse des Zylinders aus in Radial- und Querrichtung σ_{rr} und $\sigma_{\theta\theta}$ wird die Spannung durch folgende Formel berechnet:

$$\sigma_{\theta\theta} = \frac{a^2 \cdot P_i - b^2 \cdot P_o}{b^2 - a^2} + \frac{a^2 \cdot b^2 \cdot (P_i - P_o)}{r^2 \cdot (b^2 - a^2)},$$

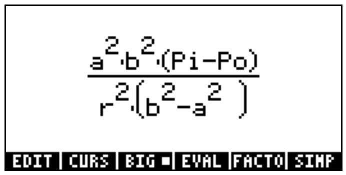
$$\sigma_{rr} = \frac{a^2 \cdot P_i - b^2 \cdot P_o}{b^2 - a^2} - \frac{a^2 \cdot b^2 \cdot (P_i - P_o)}{r^2 \cdot (b^2 - a^2)}.$$

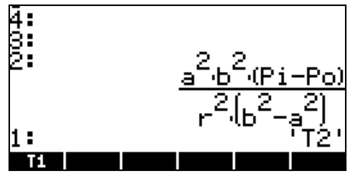
Beachten Sie dabei, dass der einzige Unterschied zwischen den beiden Gleichungen in der rechten Seite der Ausdrücke zu finden ist und allein im Vorzeichen besteht. Um diese Gleichungen in den Taschenrechner einzugeben, schlage ich vor, Sie geben den ersten Ausdruck ein und speichern diesen unter T1, dann den zweiten und speichern diesen unter T2. Eine spätere Eingabe dieser Gleichungen kann dann durch Laden von T1 und T2 in den Stack durchgeführt werden, um diese dann zu addieren und zu subtrahieren. Nachfolgend wird gezeigt, wie Sie dies im EquationWriter tun können.
Geben Sie T1 ein, und speichern Sie den Ausdruck:



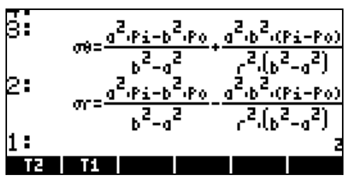


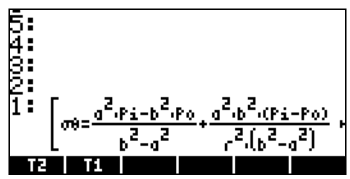
Geben Sie T2 ein, und speichern Sie den Ausdruck:



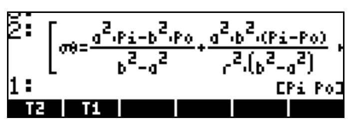


Beachten Sie, dass wir in diesem Beispiel den RPN-Modus verwenden, die Vorgehensweise im ALG-Modus ist jedoch ziemlich ähnlich. Erstellen Sie die Gleichung für $\sigma_{\theta\theta}$: VAR T1 T2 + ALPHA ↱ S ALPHA ↱ T ENTER ▶ ↱ =
Erstellen Sie die Gleichung für σ_{rr}: VAR T1 T2 − ALPHA ↱ S ALPHA ↰ R ENTER ▶ ↱ =
Erstellen Sie aus diesen beiden Gleichungen einen Vektor mit der Funktion →ARRAY (starten Sie diese aus dem Befehls-Katalog mit ↱ CAT), nachdem Sie eine 2 eingegeben haben:



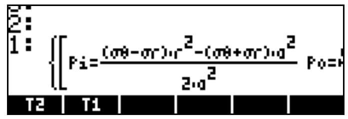


Nehmen wir nun an, dass wir P_i und P_o, mit vorgegebenen *a*, *b*, r, σ_{rr} und $\sigma_{\theta\theta}$ lösen möchten. Wir erstellen einen Vektor mit den Unbekannten:

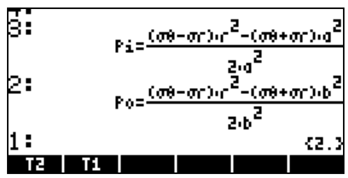


Zur Lösung für P_i und P_o verwenden wir den Befehl SOLVE aus dem Menü S.SLV (↰ S.SLV), es dauert etwa eine Minute bis das Ergebnis im Taschenrechner vorliegt:

{['Pi=-(((σθ-σr)*r^2-(σθ+σr)*a^2)/(2*a^2))'
'Po=-(((σθ-σr)*r^2-(σθ+σr)*b^2)/(2*b^2))'] }, d. h.,



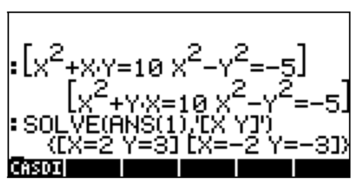
Beachten Sie, dass das Ergebnis als Vektor [ ] innerhalb einer Liste { } dargestellt ist. Benutzen Sie EVAL, um das Symbol für Liste zu entfernen. Verwenden Sie die Funktion OBJ →, um den Vektor zu zerlegen. Die Lösung lautet:



Diese beiden Beispiele stellen Systeme von linearen Gleichungen dar, welche genauso gut mit der Funktion LINSOLVE (siehe Kapitel 11) bearbeitet werden können. Das nachfolgende Beispiel zeigt die Funktion SOLVE, angewendet auf ein System von Polynomgleichungen.

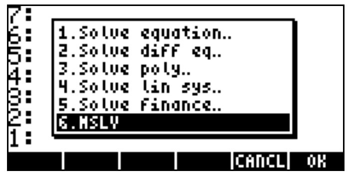
Beispiel 3 – System von Polynomgleichungen

Die nachfolgende Abbildung zeigt die Lösung des Systems $X^2+XY=10$, $X^2-Y^2=-5$ mithilfe der Funktion SOLVE:

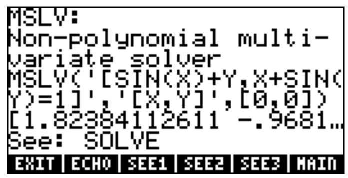


Lösungen zu simultanen Gleichungen mit MSLV

Die Funktion MSLV ist die letzte Option im Menü ↱ NUM.SLV :



Nachfolgend finden Sie den Hilfeeintrag für die Funktion MSLV:

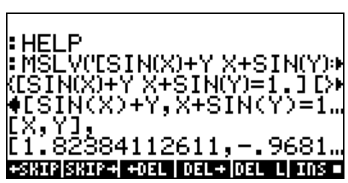


Beispiel 1 – Beispiel aus der Hilfefunktion

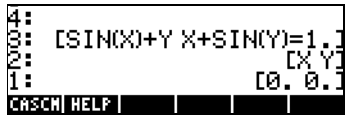
Wie für alle anderen Funktionseinträge gibt es in der Hilfefunktion auch ein Beispiel zum Eintrag MSLV, wie oben gezeigt. Beachten Sie, dass die Funktion MSLV drei Argumente benötigt:

1. Einen Vektor, der die Gleichungen enthält, d. h. '[SIN(X)+Y,X+SIN(Y)=1]'
2. Einen Vektor, der die zu lösenden Variablen enthält, d. h. '[X,Y]'
3. Einen Vektor, der die ursprünglichen Werte für die Lösung beinhaltet, d. h. die ursprünglichen Werte der Variablen X und Y sind Null in diesem Beispiel.

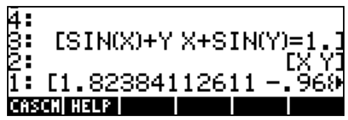
Im ALG-Modus drücken Sie ECHO, um das Beispiel in den Stack zu kopieren, drücken Sie dann ENTER, um das Beispiel auszuführen. Um alle Elemente der Lösung anzusehen, müssen Sie den Zeileneditor mit der Pfeiltaste (▼) aktivieren:



Im RPN-Modus wird die Lösung für dieses Beispiel wie folgt gefunden:



Durch Aktivierung der Funktion MSLV erscheint folgende Anzeige.



Sie haben wahrscheinlich festgestellt, dass während der Berechnung der Lösung in der linken oberen Ecke des Displays Zwischenergebnisse angezeigt werden. Da die von MSLV gelieferte Lösung numerisch ist, zeigen die Informationen in der linken oberen Ecke die Ergebnisse des wiederholten Prozesses auf dem Weg zur Lösung an. Die endgültige Lösung ist *X = 1,8238, Y = -0,9681*.

Beispiel 2 – Eingang aus einem See in einen offenen Kanal

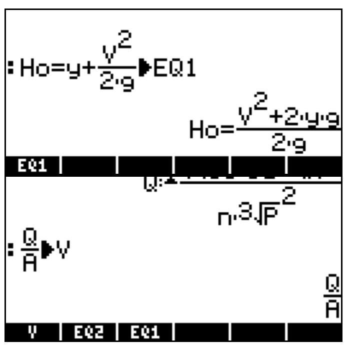
Bei diesem speziellen Problem mit dem Durchfluss in einem offenen Kanal muss die Lösung für zwei Gleichungen simultan erfolgen, nämlich für die Energiegleichung $H_o = y + \frac{V^2}{2g}$ und die Manning-Gleichung:

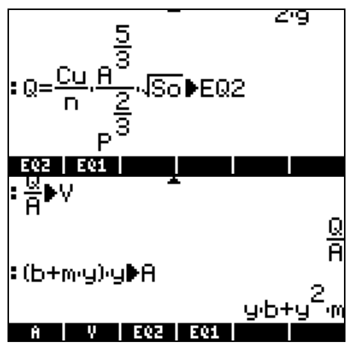
$Q = \frac{Cu}{n} \cdot \frac{A^{5/3}}{P^{2/3}} \cdot \sqrt{S_o}$. In diesen Gleichungen stellt H_o die Druckhöhe (m oder ft) dar, welche beim Durchfluss am Eingang des Kanals vorhanden ist, y die Durchflusstiefe (m oder ft), V = Q/A die Geschwindigkeit (m/s oder ft/s), Q die Wassermenge (m^3/s oder ft^3/s), A den Flächeninhalt des Querschnitts (m^2 oder ft^2), C_u den Koeffizienten, welcher von den Systemeinheiten (C_u = 1,0 für die SI, C_u = 1,486 für das englische Einheitensystem) abhängt, n den Manning-Koeffizienten, eine Maßeinheit der Rauheit der Kanalwandoberfläche (z. B. für Beton, n = 0,012), P der benetzte Umfang des Querschnitts (m oder ft) und S_o die Neigung des Kanalbettes als Bruch. Für einen trapezförmigen Kanal, wie unten gezeigt, wird die Fläche über die Formel $A = (b + my)y$ dargestellt, während der benetzte Umfang mit

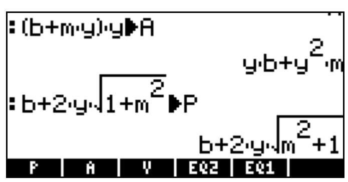
$P = b + 2y\sqrt{1+m^2}$ gegeben ist, wobei b die Breite des Bodens (m oder ft) und m die Seitenwandneigung (1V:mH) des Querschnittes darstellt.

Normalerweise muss man die Energie- wie auch die Manning-Gleichung für y und Q gleichzeitig lösen. Sobald diese Gleichungen als primitive Variablen b, m, y, g, S_o, n, Cu, Q und H_o dargestellt sind, bleibt uns das folgende Gleichungssystem $f_1(y,Q) = 0$, $f_2(y,Q) = 0$. Diese beiden Gleichungen können wir wie folgt erstellen.

Wir nehmen an, daß wir das ALG und die genauen Modi im Rechner verwenden werden, obgleich, die Gleichungen zu definieren und das Lösen sie mit MSLV im RPN Modus sehr ähnlich ist. Erstellen Sie ein Unterverzeichnis, beispielsweise CHANL (für offener CHANneL – Kanal) und innerhalb dieses Unterverzeichnisses die nachfolgenden Variablen.

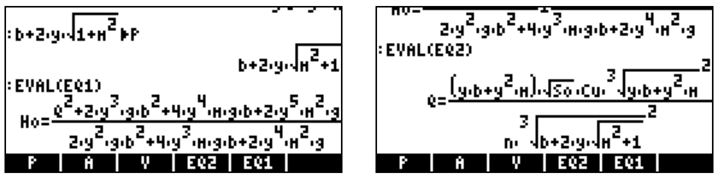






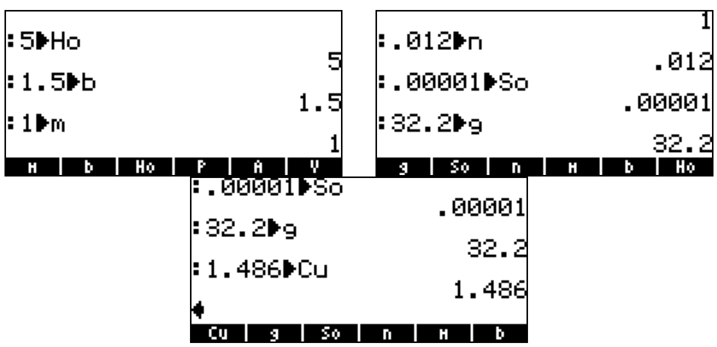
Um die ursprünglichen Gleichungen EQ1 und EQ2 wie oben dargestellt als einfache Variablen zu sehen, können wir die Funktion EVAL nutzen, welche wir auf jede Gleichung anwenden, d. h., EVAL EQ1 EVAL EQ2. Die

Gleichungen werden im Stack wie folgt angezeigt (kleine Schriftart ausgewählt):

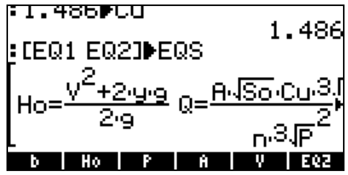


Wir stellen fest, dass diese Gleichungen tatsächlich als einfache Variable b, m, y, g, S_o, n, Cu, Q und H_o dargestellt werden können.

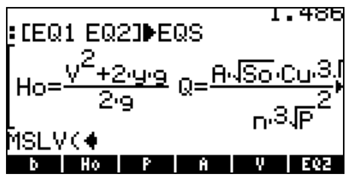
Um y und Q zu lösen, müssen wir den anderen Variablen Werte zuweisen. Angenommen, wir verwenden folgende Werte: H_0 = 5 ft, b = 1,5 ft, m = 1, n = 0,012, S_0 = 0,00001, g = 32,2 und Cu = 1,486. Bevor wir diese Variablen zur Lösung mit MSLV verwenden können, müssen wir die Werte für die entsprechenden Variablennamen eintragen. Dies kann wie folgt erreicht werden:



Nun sind wir bereit die Gleichung zu lösen. Zuvor jedoch müssen wir die beiden Gleichungen zusammen in einen Vektor eingeben. Dies erreichen wir durch Speichern des Vektors in einer Variablen mit dem Namen EQS (EQationS – Gleichungen):



Als Anfangswert für die Variablen y und Q verwenden wir y = 5 (entspricht dem Wert von H_o, welches der Maximalwert ist, den y annehmen kann) und Q = 10 (dies ist nur eine Schätzung/Schätzwert). Um die Lösung zu erhalten, wählen wir die Funktion MSLV aus dem Menü NUM.SLV, z. B. ⟶ NUM.SLV 6 OK, zur Eingabe des Befehls im Display:



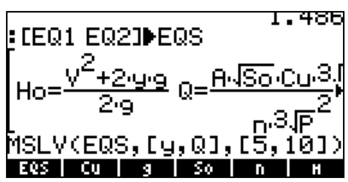
Als Nächstes geben wir die Variable EQS NXT NXT EQS, gefolgt vom Vektor [y,Q] ein:

⟶ , ← [] ALPHA ← Y ⟶ , ALPHA Q ▶

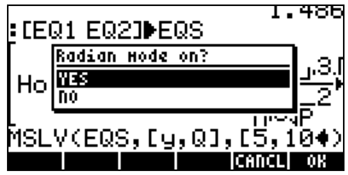
zusammen mit unseren anfänglichen Schätzwerten

⟶ , ← [] 5 ⟶ , 1 0 ein.

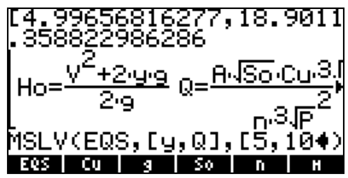
Vor dem Drücken der Taste ENTER sieht die Anzeige wie folgt aus:



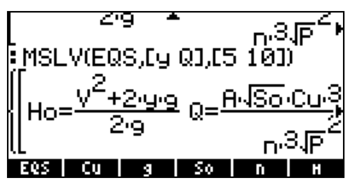
Drücken Sie ENTER, um das System von Gleichungen zu lösen. Wenn Sie das Winkelmaß nicht auf Radian gestellt haben, könnten Sie folgende Anfrage erhalten:



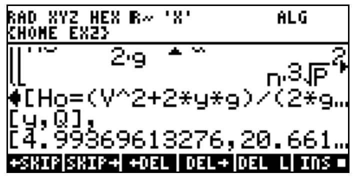
Drücken Sie OK, und fahren Sie mit der Lösung fort. Ein Zwischenergebnis könnte wie folgt aussehen:



Der Vektor im oberen Teil, welcher, während der Lösungsprozess fortschreitet, die aktuellen Werte von [y,Q] und den Wert ,358822986286, welches die Konvergenzkriterien der numerischen Methode, die für die Lösungsfindung benutzt wurde, anzeigt. Wenn das System gut eingestellt ist, wird sich dieser Wert an Null annähern. An dieser Stelle sollte eine numerische Lösung gefunden worden sein. Nachdem MSLV eine Lösung gefunden hat, sollte die Anzeige wie folgt aussehen:



Das Ergebnis ist eine Liste von drei Vektoren. Der erste Vektor stellt die gelösten Gleichungen dar. Der zweite Vektor ist die Liste der Unbekannten. Der dritte Vektor stellt die Lösung dar. Um sich diese Vektoren anzusehen, drücken Sie die Pfeiltaste ▼ zum Starten des Zeileneditors. Das Ergebnis wird wie folgt angezeigt:



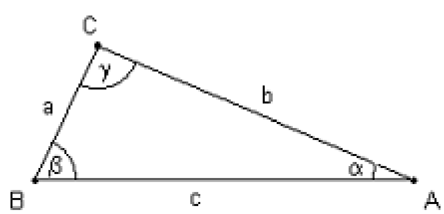
Die vorgeschlagene Lösung ist [4,9936.. ; 20,661…]. Das bedeutet, y = 4,99 ft und Q = 20,66 ft^3/s. Um die Lösung im Detail anzuzeigen, benutzen Sie die Pfeiltasten (◀ ▶ ▲ ▼).

Der Multiple Equation Solver (MES) (Mehrfachgleichungslöser)

Der Mehrfachgleichungslöser ist eine Umgebung, in der Systeme von Mehrfachgleichungen durch Lösen jeweils einer Unbekannten aus einer Gleichung gelöst werden können. Es ist eigentlich nicht ein Löser für Simultanlösungen, sondern eher ein schrittweiser Löser für eine Reihe von ähnlichen Gleichungen. Um die Anwendung des MES bei der Lösung von Mehrfachgleichungen zu veranschaulichen, stellen wir im nächsten Abschnitt eine trigonometrische Anwendung vor. Die nachfolgenden Beispiele werden im RPN-Modus erzeugt:

Anwendung 1 – Lösung von Dreiecken

In diesem Abschnitt verwenden wir eine wichtige Anwendung von trigonometrischen Funktionen: die Berechnung der Maße eines Dreieckes. Die Lösung ist im Taschenrechner unter Verwendung des Mehrfachgleichungslösers oder kurz MES implementiert.
Nehmen wir das Dreieck ABC aus der Abbildung unten.



Die Summe der Innenwinkel eines Dreiecks ist immer 180°, d. h. $\alpha + \beta + \gamma$ = 180°. Der Sinussatz besagt dass:

$$\frac{\sin\alpha}{a} = \frac{\sin\beta}{b} = \frac{\sin\gamma}{c}.$$

Der Kosinussatz besagt dass:

$$a^2 = b^2 + c^2 - 2 \cdot b \cdot c \cdot \cos\ \alpha,$$
$$b^2 = a^2 + c^2 - 2 \cdot a \cdot c \cdot \cos\ \beta,$$
$$c^2 = a^2 + b^2 - 2 \cdot a \cdot b \cdot \cos\ \gamma.$$

Um ein Dreieck lösen zu können, müssen Sie mindestens 3 der folgenden sechs Variablen kennen: *a, b, c,* α, β, γ. Dann können Sie die Gleichungen des Sinussatzes, des Kosinussatzes und der Summe der Innenwinkel anwenden, um die anderen drei Variablen zu berechnen.

Sind drei Seiten bekannt, kann die Fläche des Dreiecks mit der Heronschen Formel $A = \sqrt{s \cdot (s-a) \cdot (s-b) \cdot (s-c)}$, berechnet werden, wobei s als der Halbumfang des Dreiecks bekannt ist, d. h. $s = \frac{a+b+c}{2}$.

Lösung eines Dreiecks anhand des Multiple Equation Solvers (MES) (Mehrfachgleichungslösers)

Der Mehrfachgleichungslöser (MES) ist ein Merkmal des Taschenrechners, das zur Lösung zweier oder mehrerer gekoppelter Gleichungen verwendet wird. Es muss noch darauf hingewiesen werden, dass der MES die Gleichungen nicht simultan löst. Die Vorgehensweise ist die, dass er die bekannten Variablen nimmt, und dann in einer Liste von Gleichungen sucht, bis eine gefunden wird, die für die unbekannten Variablen gelöst werden kann. Dann wird nach einer weiteren Gleichung gesucht, die für die nächsten Unbekannten gelöst werden kann, usw., bis alle Unbekannten gelöst sind.

Erstellen eines Arbeitsverzeichnisses

Wir werden den MES zur Lösung von Aufgabenstellungen in Zusammenhang mit Dreiecken verwenden, unter Verwendung einer Liste von Gleichungen, die dem Sinus- und Kosinus-Satz, der Regel der Summe der Innenwinkel eines Dreiecks und der Heronschen Formel für die Fläche entsprechen. Erstellen Sie als Erstes im HOME-Verzeichnis ein Unterverzeichnis mit dem Namen TRIANG, und wechseln Sie in dieses Verzeichnis. Anweisungen zur Erstellung von Unterverzeichnissen finden Sie in Kapitel 2.

Eingabe der Liste von Gleichungen

Geben Sie entweder durch Direkteingabe in den Stack oder mithilfe des EquationWriters im Verzeichnis TRIANG die Liste der Gleichungen ein. (Beachten Sie dabei, dass ALPHA ↱ A das Zeichen α und ALPHA ↱ B das Zeichen β erzeugt. Das Zeichen γ muss über ↱ CHARS mit ECHO erzeugt werden):

'SIN(α)/a = SIN(β)/b'
'SIN(α)/a = SIN(γ)/c'
'SIN(β)/b = SIN(γ)/c'
'c^2 = a^2+b^2-2*a*b*COS(γ)'
'b^2 = a^2+c^2-2*a*b*COS(β)'
'a^2 = b^2+c^2-2*b*c*COS(α)'
'α+β+γ = 180'
's = (a+b+c)/2'
'A = √ (s*(s-a)*(s-b)*(s-c))'

Geben Sie anschließend die Zahl 9 ein, und erstellen Sie eine Liste von Gleichungen mit der Funktion →LIST (verwenden Sie dazu den Befehls-Katalog ↱ CAT). Speichern Sie die Liste in der Variablen EQ.

Die Variable EQ enthält nun eine Liste von Gleichungen, welche während des Lösungsvorgangs mit dem MES zur Lösungsfindung gescannt (durchsucht) werden.

Titel für ein Fenster eingeben

Als Nächstes erstellen wir eine String-Variable mit dem Namen TITLE, welche den String zur "Lösung des Dreiecks" enthält, wie folgt:

[↱] [" "]	Öffnen Sie die Anführungszeichen im Stack
[ALPHA] [ALPHA] [↰] [ALPHA]	Sperrt die Tastatur für die Eingabe von Kleinbuchstaben.
[↰] [T] [R] [I] [A] [N] [G] [L] [E] [SPC]	Geben Sie den Text ein: Triangle_ (Dreieck_)
[↰] [S] [O] [L] [U] [T] [I] [O] [N]	Geben Sie den Text ein: Lösung
[ENTER]	Geben Sie den String "Triangle Solution" (Lösung des Dreiecks) in den Stack ein
[']	Öffnen Sie die einfachen (') Anführungszeichen im Stack
[ALPHA] [ALPHA] [T] [I] [T] [L] [E] [ENTER]	Geben Sie den Variablennamen 'TITLE' (Titel) ein
[STO▸]	Speichern Sie den String in 'TITLE'

Erstellen einer Liste von Variablen

Erstellen Sie anschließend eine Liste von Variablennamen im Stack, wie nachfolgend gezeigt:

{ a b c α β γ A s }

und speichern Sie diese in der Variablen LVARI (Liste von VARIablen). Die Liste von Variablen stellt die Reihenfolge, in der die Variablen im MES aufgelistet werden, dar, sobald dieser gestartet wird. In dieser Liste müssen alle Variablen für die Gleichungen enthalten sein, da ansonsten die Funktion MTM (siehe unten) nicht ordnungsgemäß ausgeführt werden kann. Nachfolgend ist die Reihenfolge der Tastenanschläge, die dazu erforderlich sind, die Liste zu erstellen und zu speichern:

Drücken Sie [VAR], falls nötig, um ins Variablenmenü zu gelangen. In Ihrem Menü sollten die Variablen LVARI TITLE EQ angezeigt werden.

Vorbereitungen den MES auszuführen

Als nächster Schritt wird der MES gestartet und eine Beispiellösung zu gesucht. Bevor wir dies aber tun, setzen wir die Winkelmaße auf DEGrees (Grade), falls diese nicht bereits auf Grad umgestellt sind. Dazu geben wir ein: [ALPHA] [ALPHA] [D] [E] [G] [ENTER].

Weiterhin möchten wir die Inhalte der Variablen TITLE und LVARI im Stack behalten; dies geschieht durch Verwendung von:

TITLE LVARI

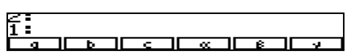
Wir werden die nachfolgenden MES-Funktionen verwenden

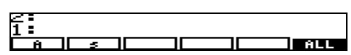
- MINIT: MES INITialization: (Initialisierung des MES) initialisiert die in EQ gespeicherten Variablen der Gleichungen
- MITM: MES' Menu Item: (Menüpunkt) Nimmt einen "title" aus Stack-Ebene 2 und die Liste der Variablen aus Stack-Ebene 1 und setzt diese in der über die Liste angegebenen Reihenfolge über das MES-Fenster und die Liste der Variablen als Funktionstasten. Im vorliegenden Beispiel haben wir bereits einen Titel ("Triangle Solution") und eine Variablenliste ({ a b c α β γ A s }) in Stack-Ebene 2 und 1 bereit, um den MITM zu starten.
- MSOLV: MES SOLVEr (MES Löser) startet den Mehrfachgleichungslöser (MES) und wartet auf eine Anwendereingabe.

MES interaktiv verwenden

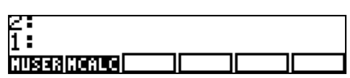
Um den MES mit den im Stack aufgeführten Variablen LVARI und TITLE zu starten, aktivieren Sie den Befehl MINIT, anschließend MITM und schließlich MSOLV (diese Funktionen sind im Befehls-Katalog ↱ CAT zu finden).

Der MES wird mit der folgenden vorhandenen Liste von Variablen gestartet (Drücken Sie NXT, um die vollständige Liste anzusehen):





Drücken Sie NXT ein weiteres Mal, um auch die dritte Variablenliste anzuzeigen. Sie sollten folgendes sehen:



Drücken Sie NXT noch einmal, um die erste Menüseite von Variablen anzuzeigen.

Versuchen wir eine einfache Lösung des Falles I unter Verwendung von a = 5, b = 3, c = 5. Benutzen Sie dazu folgende Einträge:

5 [a]	a:5 wird in der oberen linken Ecke des Displays angezeigt.
3 [b]	b:3 wird in der oberen linken Ecke des Displays angezeigt.
5 [c]	c:5 wird in der oberen linken Ecke des Displays angezeigt.

Um die Winkel zu ermitteln, verwenden Sie:

↰ [α]	Der Rechner meldet Solving for α, (löse für α) und zeigt das Ergebnis α: 72.5423968763.

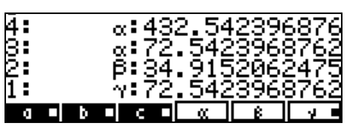
Anmerkung: Ist der Wert, den Sie erhalten größer als 180, versuchen Sie folgend

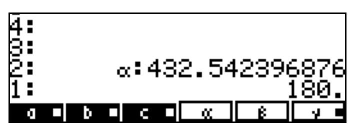
1 0 [α]	Re-initialize a to a smaller value.
↰ [α]	Calculator reports Solving for α

Anschließend berechnen wir die beiden anderen Werte:

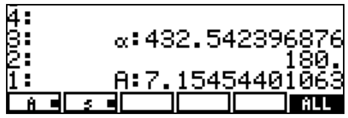
↰ [β]	Die Lösung ist β: 34.9152062474.
↰ [γ]	Die Lösung ist γ: 72.5423968763.

Sie sollten nun die Werte der drei aufgeführten Winkel in Stack-Ebene 3 bis 1 sehen. Drücken Sie + zweimal, um zu überprüfen, ob die Summe aller drei zusammen tatsächlich 180° beträgt.



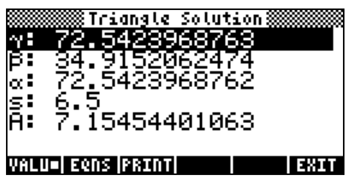


Drücken Sie die Taste NXT, um zur nächsten Seite des Menüs zu gelangen. Um nun die Fläche zu berechnen, verwenden Sie ↰[A]. Der Taschenrechner wird zuerst die Lösung aller anderen Variablen ermitteln und anschließend die Fläche A berechnen: 7.15454401063.



Anmerkung: Sobald eine Lösung gefunden wurde, meldet der Taschenrechner die Bedingungen für die Lösung entweder als Null oder Sign Reversal (Vorzeichenänderung). Möglicherweise werden weitere Meldungen angezeigt, sobald der Taschenrechner Schwierigkeiten bei der Lösungsfindung begegnet.

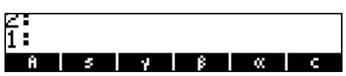
Drücken Sie nun [←] ALL werden alle Variablen gelöst, zeitweise werden Zwischenergebnisse angezeigt. Drücken Sie nun [→] ALL, um alle Lösungen zu sehen:



Wenn Sie den Vorgang abgeschlossen haben, drücken Sie [ON], um zur MES-Umgebung zurückzukehren. Drücken Sie [VAR], um die MES-Umgebung zu verlassen und zur Normalansicht des Taschenrechners zurückzukehren.

Variablen in Unterverzeichnissen organisieren

Ihr Variablenmenü wird nun die folgenden Variablen enthalten (drücken Sie [NXT], um auch die zweite Seite der Variablen anzuzeigen):



Es wurden Variablen, die allen Variablen in den Gleichungen aus EQ entsprechen, erstellt. Nun gibt es da noch eine neue Variable *Mpar* (MES Parameter), welche Informationen zu den aktuellen Einstellungen des MES für diesen einen Satz von Gleichungen enthält. Mithilfe von [→] Mpar können Sie den Inhalt der Variablen *Mpar* anzeigen. Sie erhalten folgendes kryptische Ergebnis: Library Data. (Bibliotheksdaten). Dies bedeutet, dass die MES-Parameter in einer Binärdatei kodiert sind und der Anwender auf diese nicht zugreifen kann.

Als Nächstes möchten wir die Reihenfolge der Parameter im Menü ändern, was wir unter Verwendung der folgenden Schritte durchführen können:

1. Erstellen Sie eine Liste, welche { EQ Mpar LVARI TITLE } enthält, unter Verwendung von:

   ← {} EQ Mpar LVARI TITLE ENTER

2. Setzen Sie im Stack die Inhalte von LVARI, unter Verwendung von: LVARI.
3. Verknüpften Sie beide Listen, indem Sie + drücken.

Verwenden Sie die Funktion ORDER (benutzen Sie dazu den Befehls-Katalog → CAT), um die Reihenfolge der Variablen, die in Stack-Ebene 1 angezeigt werden, festzulegen.

4. Drücken Sie VAR, um Ihre Variablenliste wiederherzustellen. Die Liste sollte wie folgt aussehen:

5. Drücken Sie NXT, um das erste Variablenmenü wiederherzustellen.

Programmieren der MES Dreiecks-Lösung über User RPL

Um das Starten des MES für spätere Anwendungen zu vereinfachen, erstellen wir ein Programm, das den MES mit einer einzigen Taste aufruft. Das Programm sollte wie folgt aussehen: << DEG MINIT TITLE LVARI MITM MSOLVR >> und kann wie folgt eingegeben werden:

→ « »	Öffnet das Programmsymbol
ALPHA ALPHA	Sperrt die alphanumerische Tastatur
D E G SPC	Geben Sie DEG (Winkelmaß wird auf DEGress (Grad) gesetzt) ein
M I N I T SPC	Geben Sie MINIT_ ein
ALPHA	Entsperrt die alphanumerische Tastatur

TITLE	Listen Sie den Namen TITLE im Programm auf
LVARI	Listen Sie den Namen LVARI im Programm auf
ALPHA ALPHA	Sperrt die alphanumerische Tastatur
M I T M SPC	Geben Sie MITM_ ein
M S O L V R	Geben Sie MSOLVR
ENTER	Geben Sie das Programm in den Stack ein

Speichern Sie das Programm in einer Variablen mit dem Namen TRISOL (für TRIangle SOLution – Lösung für das Dreieck) unter Verwendung von:
' ALPHA ALPHA T R I S O L ENTER STO▶

Drücken Sie VAR, falls nötig, um Ihre Variablenliste wieder herzustellen. In Ihrem Menü sollte jetzt die Funktionstaste TRISO zur Verfügung stehen.

Das Programm ausführen - Lösungsbeispiele

Um das Programm auszuführen, drücken Sie die Funktionstaste TRISO. Sie erhalten nun das MES-Menü, das der Lösung des Dreiecks entspricht. Versuchen wir nun die Beispiele der drei weiter oben aufgeführten Fälle zur Lösung des Dreiecks.

Beispiel 1 – rechtwinkliges Dreieck
Verwenden Sie a = 3, b = 4, c = 5. Hier ist die Lösungsfolge:

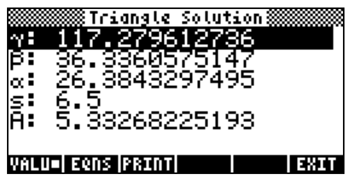
3 [a] 4 [b] 5 [c]	zur Eingabe der Daten
↰ [α]	Das Ergebnis lautet für α: 36.8698976458
↰ [β]	Das Ergebnis lautet für β: 53.1301023541.
↰ [γ]	Das Ergebnis lautet für γ: 90.
NXT	Um zum nächsten Variablenmenü zu gelangen.
[↰][A]	Die Lösung lautet für A: 6.
NXT NXT	Um zum nächsten Variablenmenü zu gelangen.

Beispiel 2 – Irgend ein Dreieck

Verwenden Sie a = 3, b = 4, c = 6. Das hier verwendete Lösungsverfahren besteht darin, alle Variablen gleichzeitig zu lösen und anschließend die Lösung in den Stack zu laden.

VAR TRISO	Um die Daten zu löschen und den MES neu zu starten
3 [a] 4 [b] 6 [c]	Um Daten einzugeben
NXT	Um zum nächsten Variablenmenü zu gelangen.
← ALL	Lösen aller Unbekannten.
→ ALL	Zeige die Lösung:

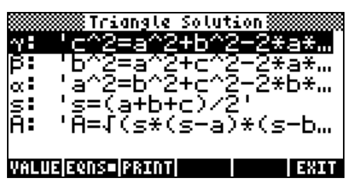
Die Lösung lautet:



Am unteren Rand der Anzeige haben Sie die Funktionstasten:

VALU▪ EQNS PRINT EXIT

Das kleine schwarze Rechteck neben dem Namen der Funktion VALU▪ zeigt an, dass die Werte und nicht die Gleichungen, aus denen diese berechnet wurden, angezeigt werden. Um sich die für die Lösung verwendeten Gleichungen anzusehen, drücken Sie die Funktionstaste EQNS. Die Anzeige sieht wie folgt aus:



Die Funktionstaste PRINT wird dazu benutzt, den Bildschirminhalt bei Bedarf auf einem Drucker auszugeben. EXIT bringt Sie zur MES-Umgebung zurück, um ggf. eine neue Lösung zu ermitteln. Um zur Normalansicht zurückzukehren, drücken Sie VAR.

Die nachfolgende Tabelle von Lösungen für Dreiecke zeigt die Dateneingabe fettgedruckt und die Lösungen in Kursivschrift. Um die Lösungen zu überprüfen, versuchen Sie, das Programm mit diesen Eingaben auszuführen. Vergessen Sie nicht, am Ende jeder Lösung VAR TRISO einzugeben, um die Variableninhalte zu löschen, und starten Sie dann die MES-Lösung erneut. Andernfalls bleiben möglicherweise Informationen aus vorangegangenen Lösungen erhalten, welches Ihre folgenden Berechnungen negativ beeinflussen könnte.

a	***b***	***c***	$\alpha(^o)$	$\beta(^o)$	$\gamma(^o)$	***A***
2.5	*6.9837*	**7.2**	*20.299*	**75**	*84.771*	*8.6933*
7.2	**8.5**	*14.26*	*22.616*	**27**	*130.38*	*23.309*
21.92	**17.5**	**13.2**	**90**	*52.97*	**37.03**	*115.5*
41.92	**23**	*29.6*	**75**	**32**	*73*	*328.81*
10.27	*3.26*	**10.5**	*77*	**18**	**85**	*16.66*
17	**25**	**32**	*31.79*	*50.78*	*97.44*	*210.71*

Hinzufügen einer INFO Schaltfläche zu Ihrem Verzeichnis

Eine Informationsschaltfläche hilft Ihnen, sich die Operationen und Funktionen des Verzeichnisses zu merken. Das Einzige, was wir in diesem Verzeichnis wissen sollten, ist, dass man durch Drücken von TRISO die Lösung des Dreiecks startet. Sie sollten folgendes Programm eingeben: <<"Drücken Sie [TRISO], um " MSGBOX >> zu starten, und speichern Sie dies in einer Variablen INFO. Als Ergebnis wird die erste Variable in Ihrem Verzeichnis nun eine INFO Schaltfläche sein.

Anwendung 2 – Geschwindigkeit und Beschleunigung in Polarkoordinaten

Zweidimensionale Partikelbewegung in Polarkoordinaten beinhaltet oft die Ermittlung der Radial- und Transversalkomponenten der Geschwindigkeit und die Beschleunigung der gegebenen Partikel r, r' = dr/dt, r" = d^2r/dt^2, θ, θ' = d θ /dt und, θ" = $d^2\theta/dt^2$. Nachfolgende Gleichungen werden verwendet:

$$v_r = \dot{r} \qquad a_r = \ddot{r} - r\dot{\theta}^2$$
$$v_\theta = r\dot{\theta} \qquad a_\theta = r\ddot{\theta} + 2\dot{r}\dot{\theta}$$

Erstellen Sie ein Unterverzeichnis POLC (POLar Coordinates - Polarkoordinaten), das wir bei der Berechnung der Geschwindigkeit und Beschleunigung in Polarkoordinaten verwenden. Geben Sie die folgenden Variablen in dieses Unterverzeichnis ein:

Programm oder Wert	zu speichern in Variable:
<< PEQ STEQ MINIT NAME LIST MITM MSOLVR >>	**SOLVEP**
"vel. & acc. polar coord." (Geschw. & entspr. Polarkoordinaten)	**NAME**
{ r rD rDD θD θDD vr vθ v ar aθ a }	**LIST**
{ 'vr = rD' 'vθ = r*θD' 'v = √(vr^2 + vθ^2)' 'ar = rDD – r*θD^2' 'aθ = r*θDD + 2*rD*θD' 'a = √(ar^2 + aθ^2)' }	**PEQ**

Nachfolgend Erläuterungen zu den Variablen:

SOLVEP = ein Programm, welches den Mehrfachgleichungslöser (MES) für die in der Variablen **PEQ** gespeicherten Gleichungen startet;

NAME = eine Variable, in welcher der Name des Mehrfachgleichungslösers (MES), und zwar*, "vel. & acc. polar coord."* Speichert;

LIST = eine Liste der in den Berechnungen verwendeten Variablen, in der Reihenfolge eingegeben, in der wir diese in der Mehrfachgleichungslöser-Umgebung anzeigen lassen möchten;

PEQ = eine Liste von Gleichungen, die gelöst werden sollen, entsprechend der Radial- und Transversalkomponenten der Geschwindigkeit (**vr**, **v**θ) und der Beschleunigung (**ar**, **a**θ) in

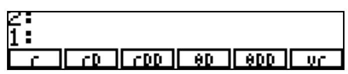
Polarkoordinaten, sowie die Gleichungen zur Berechnung der Magnitude der Geschwindigkeit (**v**) und der Beschleunigung (**a**), wenn die Polarkomponenten bekannt sind.

r, **rD**, **rDD** = r (Radialkoordinaten), r-dot (erste Ableitungsfunktion von r), r-double dot (zweite Ableitungsfunktion von r).

θ**D**, θ**DD** = θ-dot (erste Ableitungsfunktion von θ), θ-double dot (zweite Ableitungsfunktion von θ).

Angenommen, Sie haben folgende Informationen: r = 2,5, rD = 0,5, rDD = -1,5, θD = 2,3, θDD = -6,5, und Sie müssen die Werte für vr, vθ, ar, aθ, v und a ermitteln.

Starten Sie den Mehrfachgleichungslöser durch Drücken von VAR SOLVE. Im Display erscheint eine Anzeige mit der Beschriftung *"vel. & acc. polar coord."*, die wie folgt aussieht:

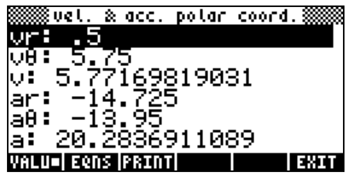


Um die Werte der bekannten Variablen einzutragen, geben Sie einfach den Wert ein und drücken Sie die Taste, die dieser Variablen entspricht. Verwenden Sie dazu folgende Tastenfolge: 2,5 [r] 0,5 [rD] 1,5 +/- [rDD] 2,3 [θD] 6,5 +/- [θDD].

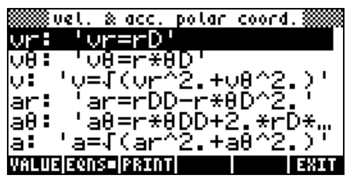
Beachten Sie dabei, dass wenn Sie einen bestimmten Wert eingegeben haben, die Variable mit dem entsprechenden Wert in der linken oberen Ecke des Displays erscheint. Nun haben wir die bekannten Variablen eingegeben. Die Ermittlung der Unbekannten kann auf zwei verschiedene Arten erfolgen:

a). Lösen Sie für einzelne Variablen, z. B. ← [vr] ergibt vr: 0,500. Drücken Sie NXT ← [vθ], um vθ : 5,750 zu erhalten usw. Die noch verbleibenden Ergebnisse sind - v: 5,77169819031; *a*r: -14,725; *a*θ: -13,95 und *a*: 20,2836911089.

b). Lösen Sie alle Variablen auf einmal, indem Sie ← ALL drücken. Im Display werden die Werte, sobald diese ermittelt wurden, angezeigt. Ist die Berechnung beendet, können Sie → ALL drücken, um alle Ergebnisse anzuzeigen. In diesem Fall haben wir:

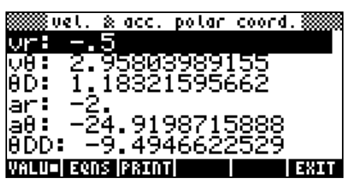


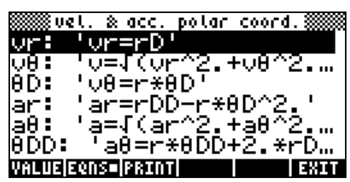
Drücken Sie die Funktionstaste EQNS, erhalten Sie die Gleichungen für jeden einzelnen Wert in der Anzeige, die zur Lösung benutzt wurden:



Um einen neuen Satz von Werten zu verwenden, drücken Sie entweder EXIT ALL NXT NXT oder VAR SOLVE.

Versuchen wir in weiteres Beispiel mit den Werten r = 2,5, vr = rD = -0,5, rDD = 1,5, v = 3,0, a = 25,0. Ermitteln Sie θD, θDD, vθ, ar und aθ. Sie sollten folgendes Ergebnis erhalten:





Kapitel 8
Operationen mit Listen

Listen sind Objekte des Taschenrechners, die besonders bei der Datenverarbeitung und in der Programmierung hilfreich sein können. In diesem Kapitel werden Beispiele von Operation mit Listen vorgestellt.

Definitionen

Im Kontext des Taschenrechners wird eine Liste als eine Reihe von Objekten, eingeschlossen in ein Klammerpaar, getrennt durch Leerschritte (SPC) im RPN-Modus oder Kommas (→ ,) in beiden Modi definiert. Objekte, die in eine Liste eingefügt werden können sind Zahlen, Buchstaben, Strings, Namen von Variablen und/oder Operatoren. Listen sind besonders bei der Manipulation von Daten-Sets und in einigen Programmieranwendungen von Bedeutung. Beispiele von Listen sind:

```
{ t 1 }, {"BETA" h2 4},{1 1.5 2.0},
{a a a a},{ {1 2 3} {3 2 1} {1 2 3}}
```

In den unten aufgeführten Beispielen beschränken wir uns auf numerische Listen.

Erstellen und Speichern von Listen

Um eine Liste im ALG-Modus zu erstellen, drücken Sie zunächst die Klammertaste ← {} (der Taste + zugeordnet), geben Sie anschließend die Elemente der Liste ein, und trennen Sie diese durch ein Komma voneinander (→ ,). Mit den nachfolgenden Tastenanschlägen geben Sie die Liste {1 2 3 4} ein und speichern diese in der Variablen L1.

← {} 1 → , 2 → , 3 → , 4
▶ STO▶ ALPHA L 1 ENTER

In der Anzeige erscheint Folgendes:

Die linke Abbildung zeigt die Anzeige vor dem Drücken der Taste ENTER, während auf der rechten Seite die Anzeige nach dem Speichern der Liste in L1 angezeigt wird. Beachten Sie vor dem Drücken der Taste ENTER, dass in der Liste die Elemente durch ein Komma getrennt dargestellt werden. Nachdem Sie nun die Taste ENTER drücken, verschwinden die Kommas und die Elemente sind durch Leerschritte voneinander getrennt.

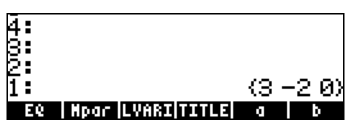
Um dieselbe Liste im RPN-Modus einzugeben, verwenden Sie die Tastenfolge:

← {} 1 SPC 2 SPC 3 SPC 4 ENTER
ALPHA L 1 ENTER ▶ STO▶

Die Abbildung zeigt den RPN-Stack vor dem Drücken der Taste STO▶:

Erstellen und Zerlegen von Listen

Erstellen und Zerlegen von Listen kann nur im RPN-Modus sinnvoll verwendet werden. In diesem Operationsmodus benutzen wir die Funktion OBJ→ zum Zerlegen der Liste. Mit dieser Funktion wird im RPN-Stack eine Liste in ihre Elemente zerlegt, wobei in Stack-Ebene 1 die Anzahl der Elemente in der Liste angezeigt wird. Die nächsten beiden Abbildungen zeigen den Stack mit einer kleinen Liste vor und nach Anwendung der Funktion OBJ→:

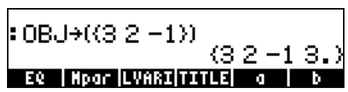


Beachten Sie, dass nach Anwenden der Funktion OBJ→ die Elemente der Liste in den Stack-Ebenen 4 bis 2 angezeigt werden, während Ebene 1 die Anzahl der Elemente in der Liste enthält.

Um im RPN-Modus eine Liste zu erstellen, geben Sie erst die Elemente in den Stack ein, dann Größe der Liste und starten Sie die Funktion →LIST (wählen

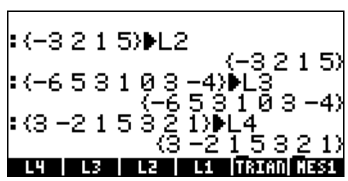
Sie diese mit folgender Tastenkombination aus dem Katalog ↱ _CAT_ ↱ ⟶ , und suchen Sie sie anschließend unter Verwendung der Pfeiltasten (▲▼) und wählen diese aus). In den nachfolgenden Abbildungen sehen Sie eine Liste der Größe 4, vor und nach Anwenden der Funktion →LIST:

Anmerkung: Wird die Funktion OBJ→ im ALG-Modus angewendet, gibt einfach die Liste wieder, und fügt dieser die Listengröße hinzu:



Operationen mit Zahlenlisten

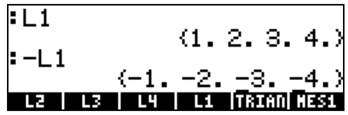
Um Operationen mit Zahlenlisten zu veranschaulichen, werden wir zusätzlich zu der oben erstellten L1 einige weitere Listen erzeugen: L2={-3,2,1,5}, L3={-6,5,3,1,0,3,-4}, L4={3,-2,1,5,3,2,1}. Nach Eingabe der Listen L2, L3 und L4 sieht die Anzeige im ALG-Modus wie folgt aus:



Im RPN-Modus werden drei Listen und deren Namen, bereit zum Speichern, angezeigt. Um die Listen in diesem Fall zu speichern, drücken Sie dreimal die Taste STO▶.

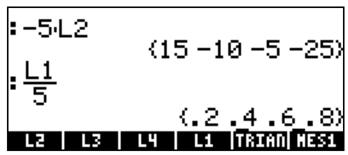
Änderung des Vorzeichens

Wenn auf eine Liste von Zahlen angewandt, ändert die Taste "Vorzeichen ändern" (+/-) das Vorzeichen aller Elemente in der Liste. So zum Beispiel:

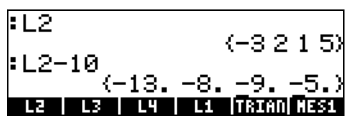


Addition, Subtraktion, Multiplikation, Division

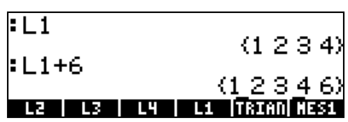
Multiplikation und Division einer Liste durch eine einzelne Zahl, wird über die gesamte Liste angewandt, so z. B.:



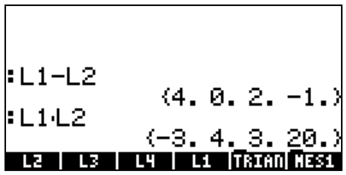
Subtraktion einer einzelnen Zahl von einer Liste, wird die Zahl von jedem Element der Liste abziehen, so z. B.:

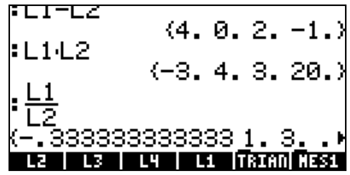


Addition einer einzelnen Zahl zu einer Liste, erzeugt eine um diese Zahl erhöhte Liste, aber nicht eine Addition zu jedem einzelnen Element der Liste. So zum Beispiel:

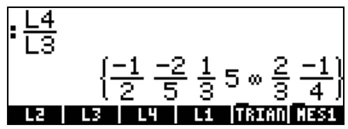


Subtraktion, Multiplikation und Division von Listen mit einer Zahlenreihe der gleichen Länge, erzeugt eine Liste der gleichen Länge mit Glied für Glied Operationen. Beispiele:



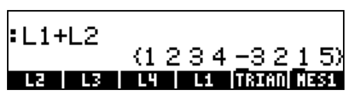


Die Division L4/L3 ergibt eine Unendliche, weil eines der Elemente in L3 eine Null ist:

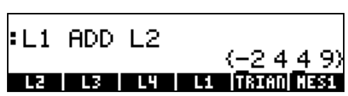


Haben die Listen für Rechenoperation verschiedene Längen, wird eine Fehlermeldung (Error: Invalid Dimensions – Fehler: ungültige Dimensionen) ausgegeben.

Wird das Pluszeichen (+) auf Listen angewandt, verhält sich dieses als *Verkettungs*operator, indem es zwei Listen zusammenfügt und nicht Glied für Glied addiert. So zum Beispiel:



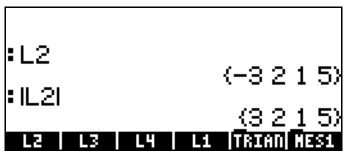
Um eine gliedweise Addition zweier Listen gleicher Länge zu erzielen, muss der Operator ADD verwendet werden. Dieser Operator kann mithilfe der Funktion Katalog (↱ _CAT_) geladen werden. In der nachfolgenden Abbildung wird eine ADD Anwendung gezeigt, um die Listen L1 und L2, gliedweise zu addieren:



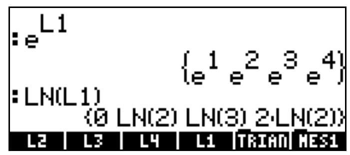
Funktionen mit reellen Zahlen von der Tastatur aus

In Listen können auch Funktionen mit reellen Zahlen (ABS, e^x, LN, 10^x, LOG, SIN, x^2, √, COS, TAN, ASIN, ACOS, ATAN, y^x) von der Tastatur aus verwendet werden. Hier einige Beispiele:

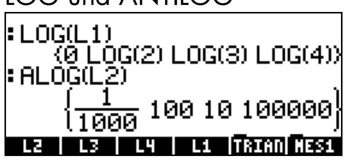
ABS



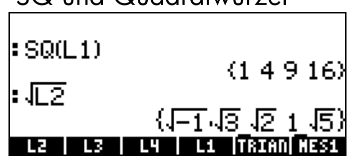
EXP und LN



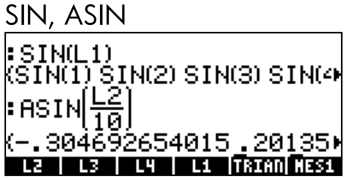
LOG und ANTILOG



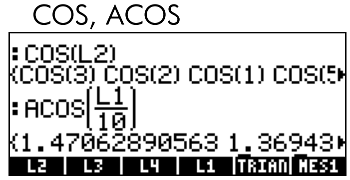
SQ und Quadratwurzel



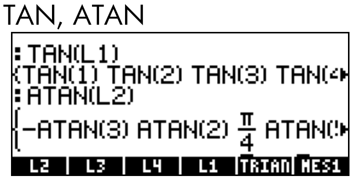
SIN, ASIN



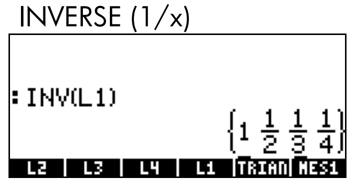
COS, ACOS



TAN, ATAN



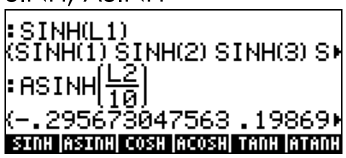
INVERSE (1/x)



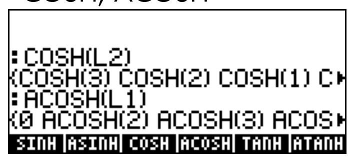
Funktionen von reellen Zahlen aus dem Menü MTH

Funktionen des MTH-Menüs die von Interesse sind – aus dem Menü HYPERBOLIC: SINH, ASINH, COSH, ACOSH, TANH, ATANH und aus dem Menü REAL: %, %CH, %T, MIN, MAX, MOD, SIGN, MANT, XPON, IP, FP, RND, TRNC, FLOOR, CEIL, D→R, R→D. Nachfolgend einige der Funktionen die nur ein Argument benötigen und auf Listen von reellen Zahlen angewendet werden können:

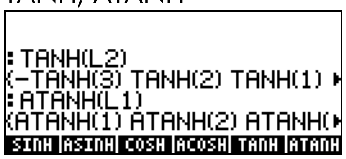
SINH, ASINH



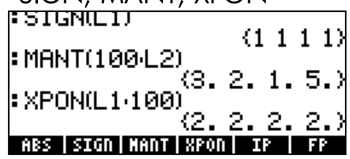
COSH, ACOSH



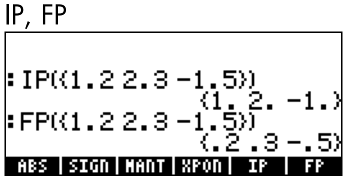
TANH, ATANH



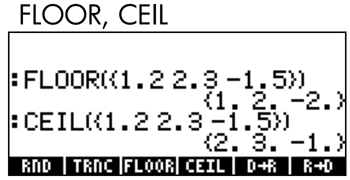
SIGN, MANT, XPON



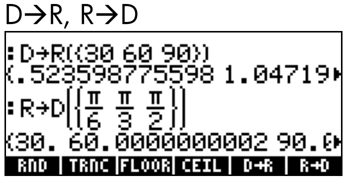
IP, FP



FLOOR, CEIL

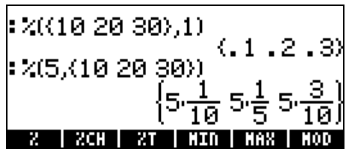


D→R, R→D



Beispiele von Funktionen, die zwei Argumente verwenden

In den nachfolgenden Abbildungen finden Sie Anwendungen der Funktion % zur Auflistung von Argumenten. Die Funktion % benötigt zwei Argumente. Das erste der beiden Beispiele zeigt Fälle, in denen nur eines der beiden Argumente eine Liste darstellt.



Die Ergebnisse sind Listen, die entsprechend der Liste der Argumente mit der Funktion % aufgeteilt sind. So zum Beispiel,

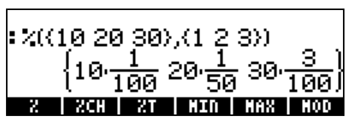
%({10, 20, 30},1) = {%(10,1),%(20,1),%(30,1)},

und

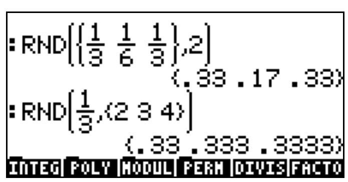
%(5,{10,20,30}) = {%(5,10),%(5,20),%(5,30)}

Im nachfolgenden Beispiel sind beide Argumente der Funktion % Listen derselben Größe. In diesem Fall, wird eine gliedweise Verteilung der Argumente durchgeführt, d. h.,

%({10,20,30},{1,2,3}) = {%(10,1),%(20,2),%(30,3)}

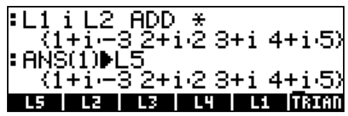


Diese Beschreibung der Funktion % für die Auflistung von Argumenten zeigt ein allgemeines Muster für die Auswertung von Funktionen mit zwei Argumenten, wobei entweder eines der Argumente oder beides Listen sind. Nachfolgend einige Anwendungsbeispiele der Funktion RND:

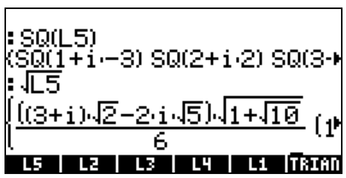


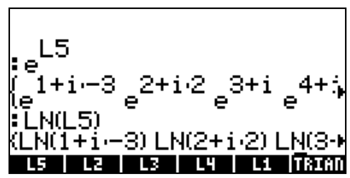
Listen von komplexen Zahlen

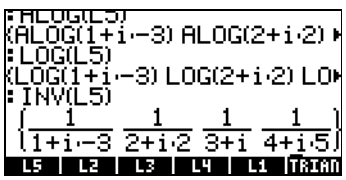
Das nachfolgende Beispiel zeigt, wie man eine Liste von komplexen Zahlen erstellt, vorausgesetzt, beide Listen sind gleich lang, wobei eine Liste die reellen Teile, während die zweite die imaginären Teile der komplexen Zahlen darstellt. Verwenden Sie L1 ADD i*L2. Die Anzeige zeigt auch, dass die daraus resultierende komplexe Zahlenliste in der Variablen L5 gespeichert wird:

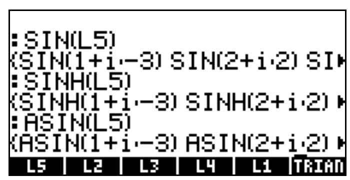


Auch Funktionen wie LN, EXP, SQ, usw. können auf Listen von komplexen Zahlen angewandt werden, z. B.:

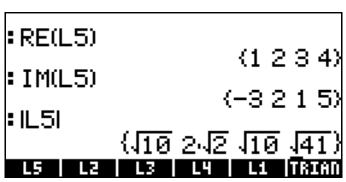


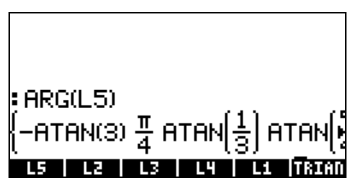






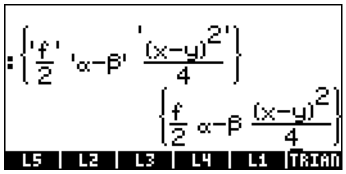
Das nachfolgende Beispiel zeigt Anwendungen der Funktion RE(reeller Teil), IM(imaginärer Teil), ABS(Magnitude) und ARG(Argument) für komplexe Zahlen. Das Ergebnis ist eine Liste von reellen Zahlen:

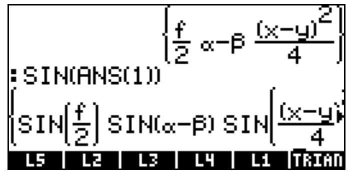




Listen von algebraischen Objekten

Nachfolgend Beispiele von Listen mit algebraischen Objekten, auf welche die Funktion SIN angewendet wurde:





Das MTH/LIST-Menü

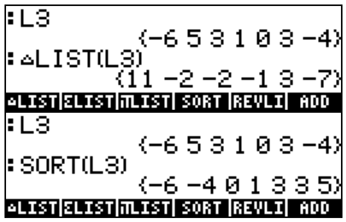
Das Menü MTH stellt eine Reihe von Funktionen, die ausschließlich auf Listen angewendet werden können, zur Verfügung. Mit dem Systemflag 117 auf CHOOSE boxes gesetzt:

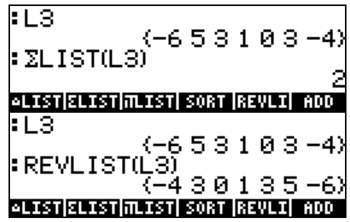
Als nächstes Systemflag auf 117 auf SOFT-Menüs gesetzt.

In diesem Menü finden wir die nachfolgenden Funktionen:

ΔLIST : Berechnet das Inkrement zwischen aufeinanderfolgenden Elementen in der Liste
ΣLIST : Berechnet die Summe der Elemente in der Liste
ΠLIST : Berechnet das Produkt der Elemente in der Liste
SORT : Sortiert die Elemente aufsteigend
REVLIST : Kehrt die Reihenfolge in der Liste um
ADD : Operator für die gliedweise Addition zweier Listen der gleichen Länge (Beispiele für diesen Operator wurden oben gezeigt)

Nachfolgend einige Anwendungsbeispiele dieser Funktionen im ALG-Modus.

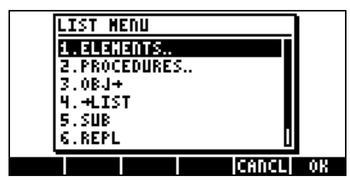




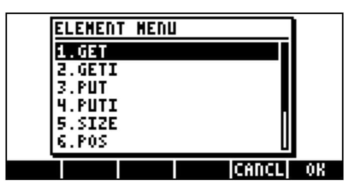
SORT und REVLIST können kombiniert werden, um eine Liste in absteigender Folge zu sortieren.

Manipulation der Elemente einer Liste

Das PRG (Programmier) Menü enthält eine Untermenü LIST mit verschiedenen Funktionen, die zur Manipulation von Elementen in einer Liste dienen. Mit dem Systemflag 117 auf CHOOSE boxes gesetzt:

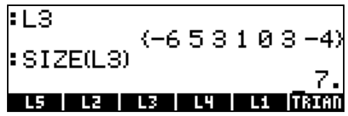


Position 1. ELEMENTS.. enthält die folgenden Funktionen, die zur Manipulation von Elementen in Listen dienen:



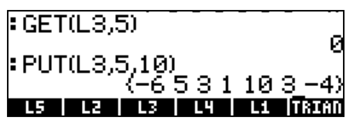
Listengröße

Die Funktion SIZE (Größe) aus dem Untermenü PRG/LIST/ELEMENTS kann zur Ermittlung der Größe (oder Länge) der Liste verwendet werden, so z. B.



Extrahieren und Einfügen von Elementen in eine Liste

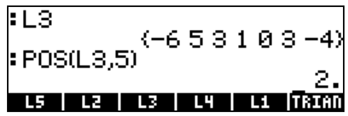
Um Elemente aus einer Liste zu extrahieren, benutzen wir die Funktion GET, welche im Untermenu PRG/LIST/ELEMENTS zu finden ist. Die Argumente der Funktion GET bilden eine Liste und die Anzahl der Elemente, die Sie aus dieser Liste extrahieren möchten. Um Elemente in eine Liste einzufügen, benutzen wir die Funktion PUT, welche ebenfalls im Untermenu PRG/LIST/ELEMENTS zu finden ist. Die Argumente der Funktion PUT bilden eine Liste mit der Position, welche wir ersetzen möchten und den Wert, mit dem diese ersetzt werden soll. Anwendungsbeispiele für die Funktionen GET und PUT finden Sie nachfolgend:



Die Funktionen GETI und PUTI, auch im Unterverzeichnis PRG/ELEMENTS/ zu finden, können auch zum Extrahieren oder Einfügen von Elementen einer Liste verwendet werden. Diese beiden Funktion werden jedoch hauptsächlich in der Programmierung verwendet. Die Funktion GETI benutzt die gleichen Argumente wie GET und gibt eine Liste, die Position der Elemente plus Eins und das Element an der gewünschten Position zurück. Die Funktion PUTI verwendet die gleichen Argumente wie GET und gibt sowohl die Liste als auch die Listengröße zurück.

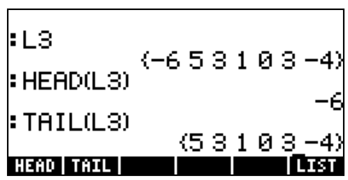
Position eines Elementes in der Liste

Zur Bestimmung der Position eines Elementes in einer Liste verwenden Sie die Funktion POS, welche die Liste und das gewünschte Element als Argument enthält. So zum Beispiel:



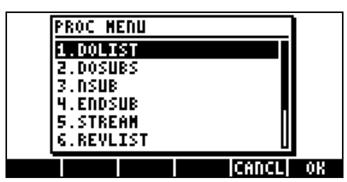
Die Funktionen HEAD und TAIL

Die Funktion HEAD extrahiert das erste Element der Liste. Die Funktion TAIL entfernt das erste Element einer Liste und gibt die noch verbleibende Liste zurück. Nachfolgend einige Beispiele:



Die Funktion SEQ

Position 2. PROCEDURES.. im Menü PRG/LIST enthält folgende Funktionen, die in Zusammenhang mit Listenoperationen verwendet werden.

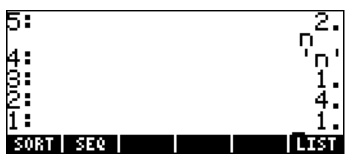


Die Funktionen REVLIST und SORT wurden bereits als Teil des Menüs MTH/LIST vorgestellt. Die Funktionen DOLIST, DOSUBS, NSUB, ENDSUB und STREAM sind zur Programmierung von Funktionen für Operationslisten im RPN-Modus gedacht. Die Funktion SEQ dient zur Erstellung einer Werteliste für einen bestimmten gegebenen Ausdruck und wird nachfolgend ausführlich beschrieben.

Die Funktion SEQ enthält als Argumente einen Ausdruck in Form eines Index, den Namen dieses Index und Start- und Endwerte, sowie deren Inkrement und gibt eine Liste zurück, die aus der Auswertung des Ausdruckes für alle möglichen Werte des Index zusammengesetzt ist. Die allgemeine Form der Funktion ist SEQ(*expression, index, start, end, increment – Ausdruck, Index, Start, Ende, Inkrement*).

Im nachfolgenden Beispiel stellen wir im ALG-Modus fest, dass für den *Ausdruck* = n^2 der *Index* = n, *Start* = 1, *Ende* = 4 und *Inkrement* = 1 ist.

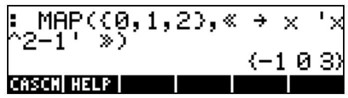
Die erzeugte Liste entspricht den Werten {1^2, 2^2, 3^2, 4^2}. Im RPN-Modus können Sie vor Anwendung der Funktion SEQ verschiedene Argumente der Funktion wie folgt auflisten:



Die Funktion MAP

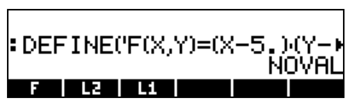
Die Funktion MAP, welche ebenfalls über den Befehls-Katalog (CAT) aufgerufen werden kann, nimmt als Argumente eine Liste von Zahlen und eine Funktion f(X) oder ein Programm << → a … >> und erzeugt eine Liste, die aus der Anwendung der Funktion f oder des Programms auf die Zahlenliste entsteht. So z. B. wendet der nachfolgende Funktionsaufruf von MAP eine SIN(X) Funktion auf die Liste {1,2,3} an:

Der nachfolgende Aufruf der Funktion MAP verwendet als zweites Argument ein Programm anstelle einer Funktion:

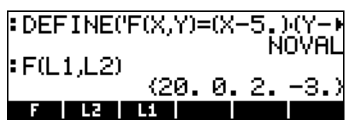


Funktionen definieren, die Listen benutzen

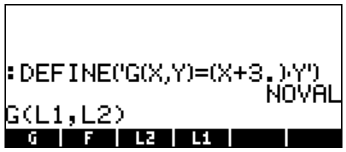
In Kapitel 3 haben wir die Funktion DEFINE (⇦ DEF), die zum Erzeugen von Funktionen aus reellen Zahlen mit einem oder mehreren Argumenten dient, vorgestellt. Eine über DEF definierte Funktion wird auch mit Argumentlisten verwendet. Beachten Sie, dass man bei Funktionen, die eine Addition enthalten, den Operator ADD und nicht das Pluszeichen (+) verwenden muss. So z. B., wenn wir im ALG-Modus (nachfolgend gezeigt) die Funktion F(X,Y) = (X-5)*(Y-2) definieren,

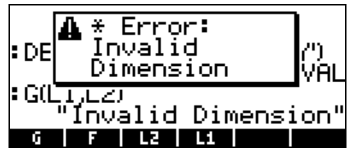


können wir Listen (z. B. die Variablen L1 und L2, welche wir weiter oben in diesem Kapitel definiert haben) zur Auswertung der Funktion verwenden, was zu folgendem Ergebnis führt:

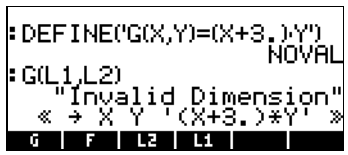


Da in der Funktion keine Addition vorkommt, ist die Anwendung dieser Funktion zur Auflistung von Argumenten recht einfach. Wenn wir jedoch die Funktion G(X,Y) = (X+3)*Y erstellen, wird ein Versuch, diese Funktion mit Argumentlisten (L1, L2) zu berechnen, fehlschlagen:

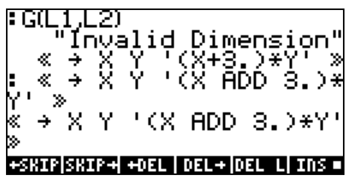




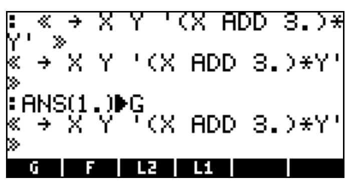
Um dieses Problem zu beheben, können wir die Inhalte der Variablen G, welche wir im Stack über die Tasten ↱ G anzeigen können, bearbeiten:



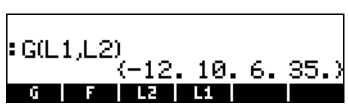
um das Pluszeichen (+) mit ADD zu ersetzen:



Anschließend speichern wir den bearbeiteten Ausdruck in der Variablen G:

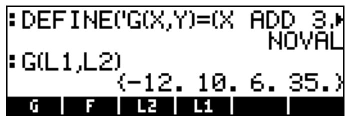


Die Auswertung G(L1,L2) ergibt nur folgendes Ergebnis:



Alternativ dazu können Sie die Funktion mit ADD anstelle des Pluszeichens (+) von Anfang an definieren, d. h. Sie verwenden

DEFINE('G(X,Y)=(X ADD 3)*Y'):



Sie können die Funktion jedoch auch als G(X,Y) = (X–3)*Y definieren.

Anwendungen für Listen

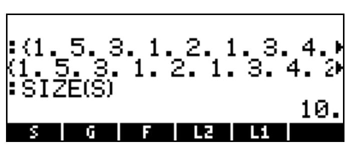
Dieser Abschnitt zeigte eine Reihe von Anwendungen von Listen, zur Berechnung von Statistiken einer Stichprobe. Unter Stichprobe verstehen wir eine Liste von Werten, beispielsweise $\{s_1, s_2, \ldots, s_n\}$. Angenommen, diese Stichprobe ist die Liste

{1, 5, 3, 1, 2, 1, 3, 4, 2, 1}

und wir speichern diese in eine Variable mit dem Namen S (in der nachfolgenden Abbildung sehen Sie dies im ALG-Modus, der entsprechende Vorgang im RPN-Modus ist sehr ähnlich. Vergessen Sie jedoch nicht, dass Sie im RPN-Modus erst die Argumente zu den Funktionen in den Stack eingeben müssen und erst dann die Funktion starten):

Harmonischer Mittelwert einer Liste

Dieses Muster ist klein genug, um die Anzahl der Elemente im Display abzählen zu können (n=10). Für eine größere Liste, können wir die Funktion SIZE benutzen, um die Anzahl der Elemente in der Liste anzuzeigen, z. B.

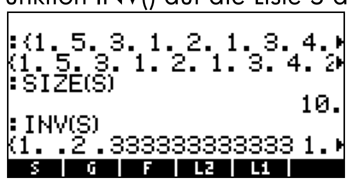


Angenommen, wir möchten das harmonische Mittel der Stichprobe, welches wie nachfolgend definiert ist, berechnen

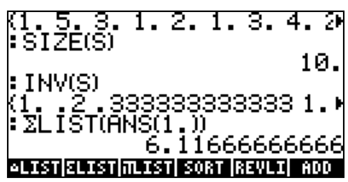
$$s_h = \frac{1}{\frac{1}{n}\sum_{k=1}^{n}\frac{1}{s_n}} = \frac{1}{\frac{1}{n}\left(\frac{1}{s_1}+\frac{1}{s_2}+\cdots+\frac{1}{s_n}\right)}.$$

Um diesen Wert zu berechnen können wir wie folgt vorgehen:

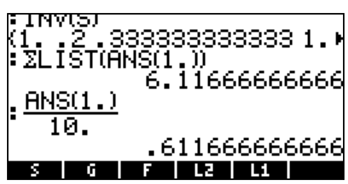
1. Wenden Sie die Funktion INV() auf die Liste S an



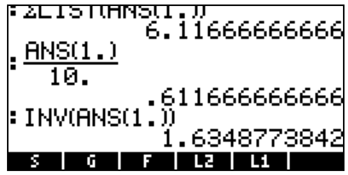
2. Wenden Sie nun auf die in Ebene 1 erhaltene Liste die Funktion ΣLIST() an.



3. Teilen Sie das obige Ergebnis durch n = 10:



4. Wenden Sie auf das letzte Ergebnis die Funktion INV() an:



Somit ist der harmonische Mittelwert der Liste S gleich s_h = 1,6348…

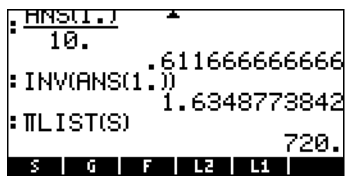
Geometrischer Mittelwert einer Liste

Der geometrische Mittelwert einer Stichprobe wird wie folgt definiert

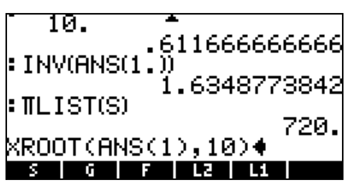
$$x_g = \sqrt[n]{\prod_{k=1}^{n} x_k} = \sqrt[n]{x_1 \cdot x_2 \cdots x_n}$$

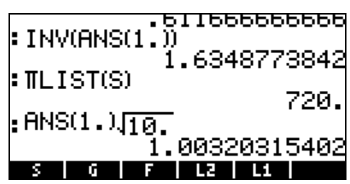
Um den geometrischen Mittelwert der in S gespeicherten Liste zu berechnen, können wir wie folgt vorgehen:

1. Wenden Sie die Funktion ΠLIST() auf die Liste S an:



2. Wenden Sie die Funktion XROOT(x,y), d. h. die Tastenfolge $\sqrt[x]{y}$ an, um das Ergebnis in Stack-Ebene 1 zu erhalten:





Somit ist der geometrische Mittelwert einer Liste S gleich s_g = 1,003203…

Gewogenes Mittel

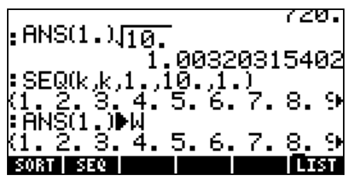
Angenommen, die Daten in Liste S, wie oben definiert, also

S = {1,5,3,1,2,1,3,4,2,1}

werden von folgenden Gewichten beeinflusst

W = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}

Wenn wir die Liste der Gewichte als W = $\{w_1, w_2, \ldots, w_n\}$ erstellen, stellen wir fest, dass das k-te Element in der Liste W, durch $w_k = k$ definiert werden kann. Somit können wir die Funktion SEQ dazu verwenden, eine Liste zu erzeugen und diese dann in die Variable W wie folgt speichern:

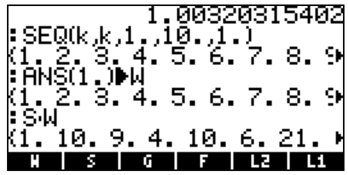


Wenn wir eine Liste mit Daten $\{s_1, s_2, \ldots, s_n\}$ und eine Liste mit Gewichten $\{w_1, w_2, \ldots, w_n\}$ haben, kann das gewogene Mittel der Daten in S wie folgt definiert werden:

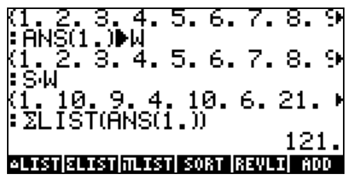
$$s_w = \frac{\sum_{k=1}^{n} w_k \cdot s_k}{\sum_{k=1}^{n} w_k}.$$

Um nun das gewogene Mittel der Daten aus der Liste S mit den Werten der Gewichte aus der Liste W zu berechnen, können wir wie folgt vorgehen:

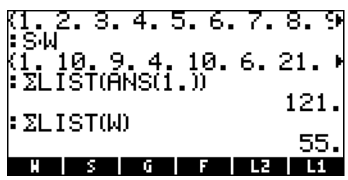
1. Multiplizieren Sie die Listen S und W:



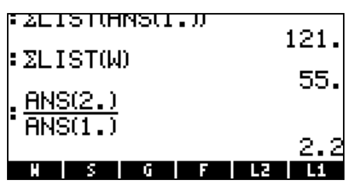
2. Wenden Sie die Funktion ΣLIST auf das erzielte Ergebnis an, um den Zähler s_w zu berechnen:



3. Wenden Sie die Funktion ΣLIST ein weiteres Mal an, um den Nenner von s_w zu berechnen:



4. Verwenden Sie den Ausdruck ANS(2)/ANS(1), um das ausgewogene Mittel zu berechnen:



Somit ist das ausgewogene Mittel einer Liste S mit Gewichten in Liste W gleich s_w= 2,2.

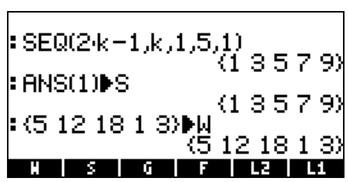
Anmerkung: ANS(1) bezieht sich auf das letzte Ergebnis (55), während sich ANS(2) auf das vorletzte Ergebnis (121) bezieht.

Statistiken gruppierter Daten

Gruppierte Daten werden normalerweise als Tabelle, unter Angabe der Frequenz (w) der Daten in Klassen oder Bins angezeigt. Jede Klasse oder Bin wird durch eine Klassenmarke (s), normalerweise der Mittelpunkt der Klasse, repräsentiert. Nachfolgend ein Beispiel von gruppierten Daten:

Klasse Grenzen	Klasse Marke s_k	Häufigkeit Zähler w_k
0 - 2	1	5
2 - 4	3	12
4 - 6	5	18
6 - 8	7	1
8 -10	9	3

Die Daten der Klassenmarken können in der Variablen S gespeichert werden, während der Häufigkeitszähler in der Variablen W wie folgt gespeichert werden kann:

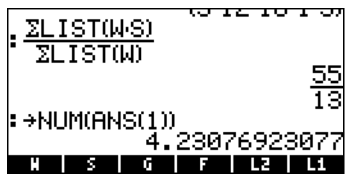


Angenommen, unsere Liste von Klassenmarken ist S = {s_1, s_2, …, s_n } und die Liste der Häufigkeitszähler W = {w_1, w_2, …, w_n }, dann stellt der Mittelwert der Daten in S mit den Gewichten W den Mittelwert der gruppierten Daten dar, welche in diesem Kontext als s bezeichnet werden.

$$\overline{s} = \frac{\sum_{k=1}^{n} w_k \cdot s_k}{\sum_{k=1}^{n} w_k} = \frac{\sum_{k=1}^{n} w_k \cdot s_k}{N},$$

wobei $N = \sum_{k=1}^{n} w_k$ den gesamten Häufigkeitszähler darstellt.

Der Mittelwert der Daten in Liste S und W kann somit mit folgendem Verfahren, wie oben für den gewogenen Mittelwert hervorgehoben, berechnet werden, d. h.

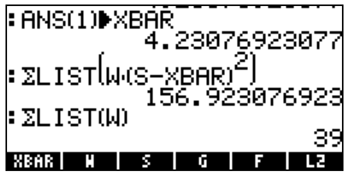


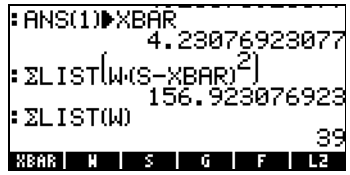
Wir speichern diesen Wert in einer Variablen mit den Namen XBAR:

Die Abweichung dieser gruppierten Daten wird wie folgt definiert

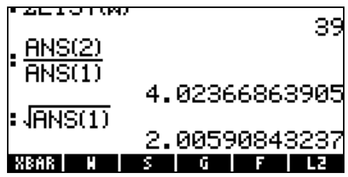
$$V = \frac{\sum_{k=1}^{n} w_k \cdot (s_k - \overline{s})^2}{\sum_{k=1}^{n} w_k} = \frac{\sum_{k=1}^{n} w_k \cdot (s_k - \overline{s})^2}{N}$$

Um das letzte Ergebnis zu berechnen, können wir wie folgt vorgehen:





Die Standardabweichung der gruppierten Daten ist die Quadratwurzel der Abweichung:



Kapitel 9
Vektoren

Dieses Kapitel stellt Beispiele zur Eingabe und Operation mit Vektoren zur Verfügung, für beide, den mathematischen bestehend aus vielen Elementen, aber auch den physikalischen bestehend aus nur 2 bis 3 Komponenten.

Definitionen

Aus mathematischer Sicht ist ein Vektor eine Gruppierung von 2 oder mehr in einer Spalte oder Zeile angeordneten Elemente. Diese bezeichnen wir als Zeilen- oder Spaltenvektoren. Nachfolgend werden einige Beispiele gezeigt:

$$v = \begin{bmatrix} -1 \\ 3 \\ 6 \end{bmatrix}, \quad u = [1, -3, 5, 2]$$

Physikalische Vektoren haben zwei oder drei Komponenten und können zur Darstellung von physikalischen Dimensionen wie: Position, Geschwindigkeit, Beschleunigung, Kraft, Momente, Impulse und Drehimpulse, Winkelgeschwindigkeit und Drehbeschleunigung usw. eingesetzt werden. Auf das Kartesische Koordinatensystem (x,y,z) bezogen, gibt es Einheitsvektoren **i**, **j**, **k,** die den einzelnen Richtungen der Koordinaten zugeordnet sind, sodass ein physikalischer Vektor hinsichtlich seiner Komponenten wie folgt geschrieben werden kann A_x, A_y, A_z als **A** = A_x**i** + A_y**j** + A_z**k**.
Eine alternative Notation für diese Vektoren ist: **A** = [A_x, A_y, A_z], **A** = (A_x, A_y, A_z), oder **A** = < A_x, A_y, A_z >. Eine zweidimensionale Version dieses Vektors wird als **A** = A_x**i** + A_y**j**, **A** = [A_x, A_y], **A** = (A_x, A_y) oder **A** = < A_x, A_y > dargestellt. Da im Taschenrechner die Vektoren zwischen eckigen Klammern [ ] geschrieben werden, wählen wir von nun an die Schreibweise **A** = [A_x, A_y, A_z] oder **A** = [A_x, A_y, A_z], um auf zwei- oder dreidimensionale Vektoren hinzuweisen. Die Magnitude eines Vektors **A** wird als |**A**| = $\sqrt{A_x^2 + A_y^2 + A_z^2}$ definiert. Ein Einheitsvektor in Richtung des Vektors **A** wird als $\mathbf{e}_A$ = **A**/|**A**| definiert. Vektoren können mit einer Skalarzahl multipliziert werden, z. B. k**A** = [kA_x, kA_y, kA_z]. Physikalisch gesehen ist der Vektor k**A**

parallel zu Vektor **A** wenn k>0 oder antiparallel zu Vektor **A**, wenn k<0 ist. Die Negative eines Vektors wird als –**A** = (–1)**A** = [$-A_x$, $-A_y$, $-A_z$] definiert. Division durch eine Skalarzahl kann als Multiplikation interpretiert werden, d. h. **A**/k = (1/k)·**A**. Addition und Subtraktion von Vektoren wird als **A**±**B** = [$A_x \pm B_x$, $A_y \pm B_y$, $A_z \pm B_y$] definiert, wobei **B** den Vektor **B** = [B_x, B_y, B_z] darstellt.

Es gibt zwei Definitionen von Produkten von physikalischen Vektoren, ein Skalar- oder internes Produkt (Skalarprodukt) und ein Vektor oder äußeres Produkt (Kreuzprodukt). Das Skalarprodukt erzeugt einen Skalar, welcher als **A•B** = |**A**||**B**|cos(θ) definiert wird, wobei θ den Winkel zwischen den beiden Vektoren darstellt. Das Kreuzprodukt erzeugt einen Vektor **A**×**B** dessen Magnitude |**A**×**B**| = |**A**||**B**|sin(θ) ist, und dessen Richtung durch die sogenannte Dreifingerregel bestimmt wird (für eine grafische Darstellung dieses Vorgangs, sehen Sie in einem Physik-, Mathe- oder Mechanik-Buch nach. Hinsichtlich Kartesischer Komponenten ist **A•B** = $A_xB_x+A_yB_y+A_zB_z$ und **A**×**B** = [A_yB_z-A_zB_y,A_zB_x-A_xB_z,A_xB_y-A_yB_x]. Der Winkel zwischen zwei Vektoren kann aus der Definition des Skalarproduktes als cos(θ) = **A•B**/|**A**||**B**|= $\mathbf{e}_A$•$\mathbf{e}_B$ ermittelt werden. Somit, wenn zwei Vektoren **A** und **B** senkrecht zueinander stehen ist (θ = 90^0 = $\pi/2^{rad}$), **A•B** = 0.

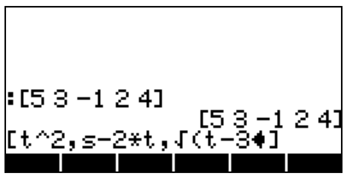
Eingabe von Vektoren

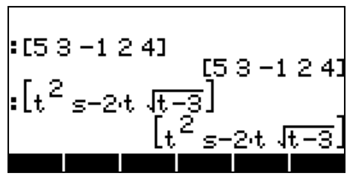
Im Taschenrechner werden die Vektoren als eine, in Klammer eingeschlossene, Reihe von Zahlen dargestellt und typischerweise als Reihe von Vektoren eingegeben. Im Taschenrechner werden die Klammern mit der Tastenkombination [←] [] erzeugt, welche der Taste [×] zugeordnet ist. Nachfolgend einige Beispiele von Vektoren im Taschenrechner.

`[3.5, 2.2, -1.3, 5.6, 2.3]`	Eine allgemeine Reihe von Vektoren
`[1.5,-2.2]`	Ein 2-D Vektor
`[3,-1,2]`	Ein 3-D Vektor
`['t','t^2','SIN(t)']`	Ein Vektor von algebraischen Objekten

Eingabe von Vektoren in den Stack

Ist der Taschenrechner im ALG-Modus, wird der Vektor durch Öffnen eines Klammerpaares (← []) und eintippen der Komponenten oder Elemente innerhalb dieser, durch Komma getrennt (→ ,), eingegeben. Die nachfolgenden Abbildungen zeigen die Eingabe eines numerischen, gefolgt von einem algebraischen Vektor. Die linke Abbildung zeigt den algebraischen Vektor vor Drücken der Taste ←. Die Abbildung rechts zeigt die Anzeige des Taschenrechners, nachdem der algebraische Vektor eingegeben wurde:



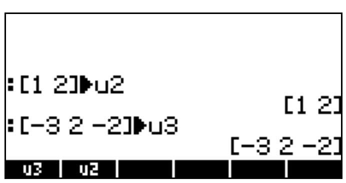


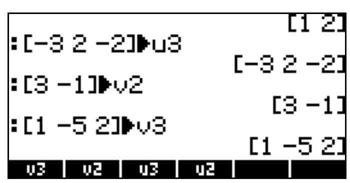
Im RPN-Modus können Sie einen Vektor in den Stack eingeben, indem Sie ein Klammernpaar öffnen und die Komponenten oder Elemente des Vektors entweder durch Komma (→ ,) oder Leerzeichen (SPC) getrennt eingeben. Beachten Sie, dass nachdem Sie die Taste ENTER gedrückt haben, der Taschenrechner in beiden Fällen die Elemente des Vektors durch Leerzeichen getrennt anzeigt.

Vektoren in Variablen speichern

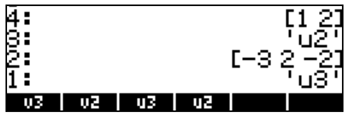
Vektoren können in Variablen gespeichert werden. In den folgenden Abbildungen sehen Sie die Vektoren

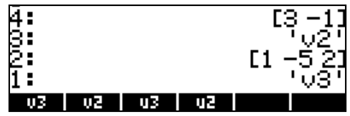
u$_2$ = [1, 2], **u**$_3$ = [-3, 2, -2], **v**$_2$ = [3,-1], **v**$_3$ = [1, -5, 2] entsprechend in den Variablen u2, u3, v2 und v3 gespeichert. Als Erstes im ALG-Modus:





Dann im RPN-Modus (bevor Sie wiederholt die Taste STO▶ drücken):





Eingabe von Vektoren mithilfe des MatrixWriters (MTRW)

Vektoren können auch über den MatrixWriter ⬅ MTRW , eingegeben werden (dritte Taste vierte Reihe von oben). Dieser Befehl erzeugt eine Art Tabelle, welche den Reihen und Spalten einer Matrix entspricht (Details zur Anwendung und Benutzung des MatrixWriters werden in einem nachfolgenden Kapitel erörtert). Für einen Vektor möchten wir Daten nur in die oberste Reihe eingeben. Standardmäßig ist die Zelle der ersten Zeile und Spalte ausgewählt. Am unteren Teil der Tabelle finden Sie nachfolgende Funktionstasten:

EDIT VEC■ ←WID WID→ GO→■ GO↓

Benutzen Sie die Taste EDIT, um die Inhalte einer ausgewählten Zelle des MatrixWriters zu ändern.

Wenn ausgewählt, wird die Taste VEC einen Vektor erzeugen, im Gegensatz zu einer Matrix, welche eine Zeile und mehreren Spalten erzeugt. .

Vektoren vs. Matrizen

Um zu sehen wie die Taste VEC funktioniert, machen Sie folgende Übung:

(1) Starten Sie den MatrixWriter (⬅ MTRW) VEC■ und GO→■ ausgewählt, geben Sie ein wie folgt 3 ENTER 5 ENTER 2 ENTER ENTER . Dies ergibt [3. 5. 2.]. Im RPN-Modus können Sie nachfolgende Tastenfolge benutzen, um das gleiche Ergebnis zu erhalten: 3 SPC 5 SPC 2 ENTER ENTER).

(2) VEC nicht ausgewählt und GO→■ ausgewählt, geben Sie ein wie folgt 3 SPC 5 SPC 2 ENTER ENTER . Dies ergibt [[3. 5. 2.]].

Obwohl diese beiden Ergebnisse sich nur in der Anzahl der verwendeten Klammern unterscheiden, stellen diese für den Taschenrechner unterschiedliche mathematische Objekte dar. Ersteres ist ein Vektor mit 3 Elementen und das zweite stellt eine Matrix mit einer Zeile und drei Spalten dar. Es gibt drei Unterschiede in der Art wie mathematische Operationen auf einen Vektor im Gegensatz zur Matrix angewendet werden. Deshalb behalten wir die Funktionstaste VEC■ selektiert, während wir den MatrixWriter benutzen.

Die Taste ←WID wird verwendet, um die Breite der Spalten in der Tabelle zu verringern. Drücken Sie die Taste mehrmals, um zu sehen wie sich die Spaltenbreite im MatrixWriter verringert.

Die Taste WID→ wird verwendet, um die Breite der Spalten in der Tabelle zu vergrößern. Drücken Sie die Taste mehrmals, um zu sehen wie sich die Spaltenbreite in ihrem MatrixWriter vergrößert.

Wenn ausgewählt, wählt die Taste GO→■ beim Drücken der Taste ENTER automatisch die nächste Zelle rechts von der Position der aktuellen Zelle. Diese Option ist standardmäßig eingestellt.

Wenn ausgewählt, wählt die Taste GO↓ beim Drücken der Taste ENTER automatisch die nächste Zelle unten von der Position der aktuellen Zelle.

Nach rechts bewegen vs. nach unten bewegen im MatrixWriter

Aktivieren Sie den MatrixWriter und geben Sie wie folgt ein 3 ENTER 5 ENTER 2 ENTER ENTER, die Taste GO→■ ist ausgewählt (Standard). Um den Unterschied zu sehen, geben Sie als Nächstes die gleiche Zahlenfolge, mit ausgewählter GO↓■ Taste ein. Im ersten Fall haben Sie einen Vektor bestehend aus drei Elementen eingegeben. Im zweiten Fall haben Sie eine Matrix bestehend aus drei Zeilen und einer Spalte eingegeben

Starten Sie den MatrixWriter über ← MTRW und drücken Sie NXT, um die zweite Zeile des Funktionsmenüs am unteren Teil der Anzeige, anzuzeigen. Die nachfolgenden Tasten werden angezeigt:

+ROW -ROW +COL -COL →STK GOTO

Die Taste +ROW fügt eine Zeile mit lauter Nullen an der Stelle der ausgewählten Zelle in der Tabelle hinzu.

Die Taste -ROW löscht die ganze Zeile, in der sie eine Zelle ausgewählt haben.

Die Taste +COL trägt eine ganze Spalte Nullen an der Stelle der ausgewählten Zelle der Tabelle ein.

Die Taste -COL löscht die Spalte in der sie eine Zelle ausgewählt haben.

Die Taste ←STK verschiebt den Inhalt der ausgewählten Zelle in den Stack.

Wenn die Taste GOTO gedrückt ist, wird der Anwender aufgefordert die Zahl für die Zeile und Spalte, an die der Cursor positioniert werden soll, einzugeben.

Wird die Taste NXT ein weiteres Mal gedrückt, erscheint das letzte Menü, welches nur noch die eine Funktion DEL (löschen) enthält.

Die Funktion DEL löscht die Inhalte der ausgewählten Zelle und ersetzt diese mit einer Null.

Um zu sehen wie diese Tasten funktionieren, machen Sie folgende Übung:

(1) Starten Sie den MatrixWriter über ← MTRW . Stellen Sie sicher, dass die Tasten VEC■ und GO→■ ausgewählt sind.

(2) Geben Sie Folgendes ein:

1 ENTER 2 ENTER 3 ENTER
NXT GOTO 2 OK 1 OK OK
2 ENTER 1 ENTER 5 ENTER
4 ENTER 5 ENTER 6 ENTER
7 ENTER 8 ENTER 9 ENTER

(3) Bewegen Sie den Cursor zwei Positionen nach oben, indem Sie die Pfeiltaste ▲▲▲ zweimal drücken. Drücken Sie anschließend -ROW. Die zweite Zeile wird verschwinden.

(4) Drücken Sie nun +ROW. Eine Zeile von drei Nullen erscheint nun in der zweiten Zeile.

(5) Drücken Sie -COL. Die erste Spalte verschwindet.

(6) Drücken Sie +COL. Eine Zeile von zwei Nullen erscheint nun in der ersten Zeile.

(7) Drücken Sie nun GOTO 3 OK 3 OK OK, um zu Position (3,3) zu springen.

(8) Drücken Sie ←STK. Der Inhalt der Zelle (3,3) wird in den Stack verschoben, obwohl Sie diese im Moment nicht bereits sehen können.

(9) Drücken Sie ENTER. Dies sollte eine Null an Position (3,3) eintragen, trotzdem aber scheint diese Funktion einwandfrei zu funktionieren.

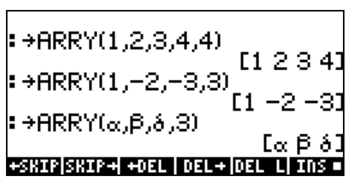
Zusammenfassung der Verwendung des MatrixWriters zur Eingabe von Vektoren
Zusammengefasst, um einen Vektor anhand des MatrixWriters einzugeben, starten Sie diesen (← MTRW) und geben Sie die Elemente des Vektors ein, indem Sie nach jedem einzelnen Element die Taste ENTER drücken. Drücken Sie anschließend ENTER ENTER. Stellen Sie sicher, dass die Tasten VEC■ und GO→■ ausgewählt sind.

Beispiel: ← MTRW ' ALPHA ← X Y^x 2 ENTER 2 ENTER 5 +/- ENTER ENTER

erzeugt: ['x^2' 2 –5]

Erstellen eines Vektors mithilfe von →ARRY

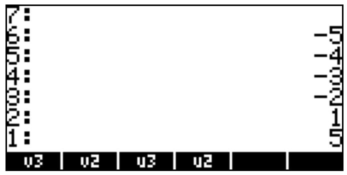
Auch die Funktion →ARRY, welche im Funktionskatalog (→ CAT → ⟶, verwenden Sie die Pfeiltasten ▲▼, um die Funktion zu lokalisieren), zu finden ist, kann zur Eingabe von Vektoren verwendet werden. Geben Sie im ALG-Modus →ARRY(*Elemente des Vektors, Anzahl der Elemente*) ein, z. B.

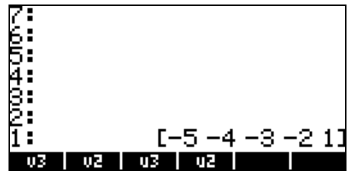


Im RPN-Modus:

(1) Geben Sie die n Elemente in der Reihenfolge, wie Sie diese angezeigt haben möchten, (wenn von links nach rechts gelesen) in den RPN-Stack ein.
(2) Geben Sie *n* als letzten Eintrag ein.
(3) Verwenden Sie die Funktion →ARRY.

Ihre Anzeige wird im RPN-Stack wie folgt angezeigt – vor und nach dem Anwenden der Funktion →ARRY:



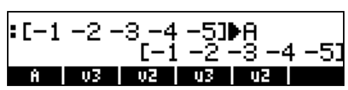


Im RPN-Modus, nimmt die Funktion [→ARRY] die Objekte aus den Stack-Ebenen *n+1, n, n-1*, …, bis hin zu Ebenen 3 und 2 und konvertiert diese in einen Vektor bestehend aus *n* Elementen. Das Objekt, das sich ursprünglich in Stack-Ebene n+1 befindet, wird so zum ersten Element, das Objekt aus Ebene n das zweite Element und so weiter.

Anmerkung: Die Funktion →ARRY kann auch über das Menü PRG/TYPE (⬅ PRG) aufgerufen werden.

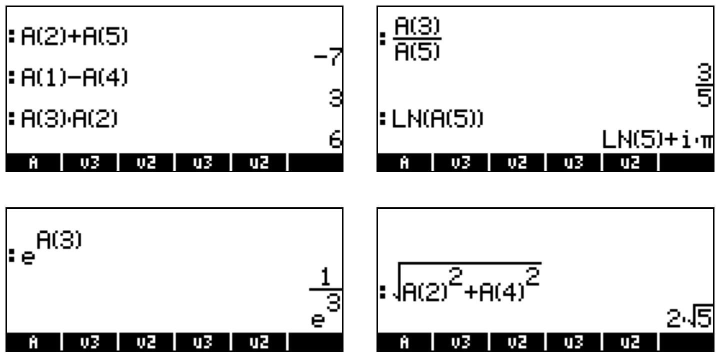
Kennung, Extrahieren und Hinzufügen von Elementen des Vektors

Speichern Sie einen Vektor in einer Variablen, beispielsweise A, können Sie die Elemente des Vektor kennzeichnen, indem Sie A(i) verwenden, wobei i eine Integer-Zahl, kleiner oder gleich der Vektorgröße darstellt. Erstellen Sie z. B. nachfolgendes Array (Reihe) und speichern Sie dieses in der Variablen A: [-1, -2, -3, -4, -5]:



Zur Erinnerung das dritte Element in A, z. B. könnten Sie als A(3) in den Taschenrechner eingeben. Im ALG-Modus geben Sie einfach A(3) ein. Im RPN-Modus müssen Sie 'A(3)' ENTER EVAL eingeben.

Mit den Elementen des Arrays können Sie Operationen durchführen, indem Sie algebraische Ausdrücke eingeben und berechnen, wie z. B.:

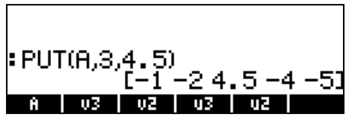


Sie können auch kompliziertere _Ausdrücke, in denen die Elemente_ von A vorkommen, erstellen. So können wir z. B. mithilfe des EquationWriters (↱ EQW) die folgende Summenbildung der Elemente aus A eingeben:

Markieren wir nun den gesamten Ausdruck und benutzen die Funktionstaste EVAL, erhalten wir das Ergebnis: -15.

Anmerkung: Vektor A können wir auch als *indexierte Variable* bezeichnen, weil A nicht nur einen, sondern mehrere Werte, welche durch den Unterindex identifiziert werden, darstellt.

Um ein Element im Array zu ersetzen verwenden wir die Funktion PUT (diese kann aus dem Funktionen Katalog ↱ CAT oder dem Untermenü PRG/LIST/ELEMENTS – letzteres wurde in Kapitel 8 erläutert – aufgerufen werden). Im ALG-Modus müssen Sie die Funktion PUT mit folgenden Argumenten verwenden: PUT(*Array, zu ersetzende Position, neuer Wert*). Um beispielsweise den Inhalt von A(3) auf 4,5 zu ändern, geben Sie wie folgt ein:



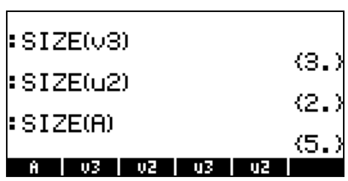
Im RPN-Modus können sie den *Wert eines Elementes* aus A *ändern*, indem Sie einen neuen Wert in diesem Element speichern. Wenn wir z. B. den Inhalt von A(3) von seinem derzeitigen Wert -3 auf 4,5 ändern möchten, gehen wir wie folgt vor:

4 • 5 ENTER ' ALPHA A ← () 3 ENTER STO▶

Um diese Änderung zu überprüfen drücken wir: ↱ A. Das Ergebnis sieht nun wie folgt aus: [-1 -2 4.5 -4 -5].

Anmerkung: Dieser Ansatz den Wert eines Elementes im Array zu ändern, ist im ALG-Modus nicht erlaubt; versuchen Sie den Wert 4,5 in A(3) in diesem Modus zu speichern, erhalten Sie nachfolgende Fehlermeldung: Invalid Syntax (ungültige Syntax).

Die Länge eines Vektors können Sie über die Funktion SIZE ermitteln – diese kann über den Befehls-Katalog (N) oder über das Untermenü PRG/LIST/ELEMENTS aufgerufen werden. Nachfolgend einige Beispiele, die auf vorhin gespeicherte Arrays und Vektoren basieren:

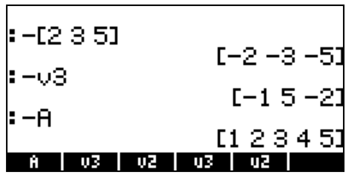


Einfache Operationen mit Vektoren

Um Operationen mit Vektoren zu veranschaulichen, verwenden wir die Vektoren A, u2, u3, v2 und v3, die wir in einer vorangegangenen Übung gespeichert haben.

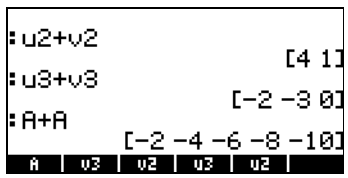
Änderung des Vorzeichens

Um das Vorzeichen eines Vektors zu ändern, benutzen Sie die Taste +/-, z. B.



Addition, Subtraktion

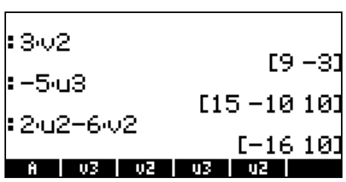
Bei der Addition und Subtraktion von Vektoren müssen die beiden Operanden des Vektors die gleiche Länge haben:

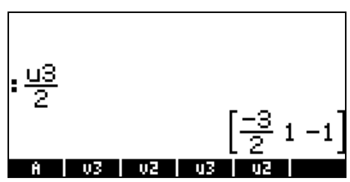


Ein Versuch Vektoren verschiedener Länge zu addieren oder zu subtrahieren, erzeugt eine Fehlermeldung (Invalid Dimension – ungültige Größe), so z. B. v2+v3, u2+u3, A+v3, usw.

Multiplikation und Division mit einem Skalar

Multiplikation und Division mit einem Skalar ist ganz einfach:

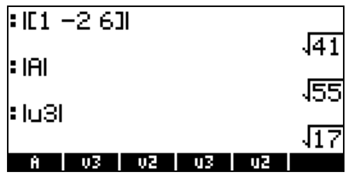




Funktion Absoluter Wert

Wird die Funktion Absoluter Wert (ABS) auf einen Vektor angewandt, ermittelt diese die Magnitude des Vektors. Die Magnitude für einen Vektor A =

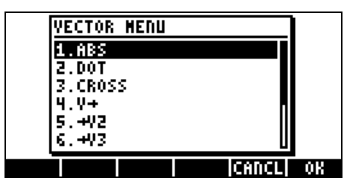
[A_1,A_2,…,A_n] wird wie folgt definiert $|A| = \sqrt{A_x^2 + A_y^2 + \cdots + A_z^2}$. Im ALG-Modus geben Sie den Namen der Funktion, gefolgt von den Argumenten des Vektors, ein. So zum Beispiel wird der Ausdruck ABS([1,-2,6]), ABS(A), ABS(u3) in der Anzeige wie folgt aussehen:

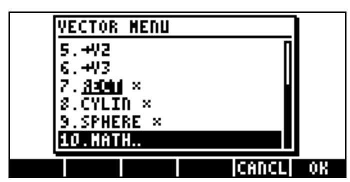


Das Menü MTH/VECTOR

Das Menü MTH (⬅ MTH) enthält ein für Objekte von Vektoren spezifisches Funktionsmenü.

Das Menü VECTOR enthält die folgenden Funktionen (Systemflag 117 ist auf CHOOSE boxes gesetzt):



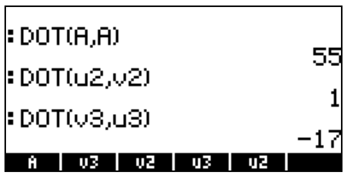


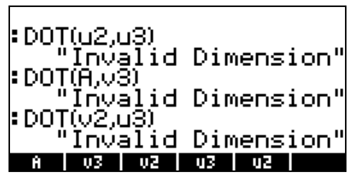
Magnitude

Die Magnitude eines Vektors, wie zuvor beschrieben, kann mit der Funktion ABS ermittelt werden. Die Funktion steht auch über die Tastatur (⬅ ABS) zur Verfügung. Anwendungsbeispiele für die Funktion ABS wurden vorher gezeigt.

Skalarprodukt

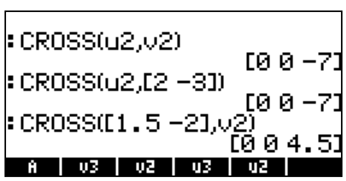
Die Funktion DOT wird zur Berechnung des Skalarproduktes zweier Vektoren der gleichen Länge verwendet. Einige Beispiele zur Anwendung der Funktion DOT, unter Verwendung der zuvor gespeicherten Vektoren A, u2, u3, v2, and v3, werden als Nächstes im ALG-Modus gezeigt. Der Versuch das Skalarprodukt zweier Vektoren unterschiedlicher Länge zu berechnen, führt zu einer Fehlermeldung:

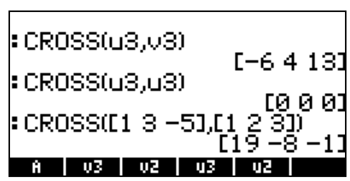




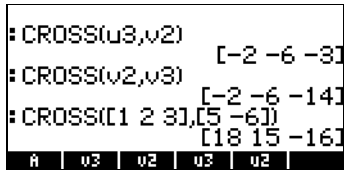
Kreuzprodukt

Die Funktion CROSS wird zur Berechnung des Kreuzproduktes zweier 2-D Vektoren, zweier 3-D Vektoren oder eines 2-D und eines 3-D Vektors, eingesetzt. Um ein Kreuzprodukt zu berechnen, wird ein 2-D Vektor der Form [A_x, A_y] als 3-D Vektor [A_x, A_y,0] behandelt. Nachfolgend werden Beispiele zweier 2-D und zweier 3-D Vektoren im ALG-Modus angezeigt. Beachten Sie, dass das Kreuzprodukt zweier 2-D Vektoren einen Vektor nur in z-Richtung, d. h. einen Vektor der Form [0, 0, C_z] erzeugt.





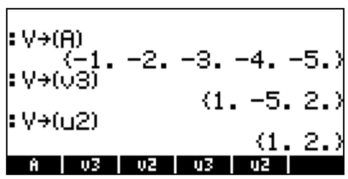
Beispiele eines Kreuzproduktes eines 3-D Vektors mit einem 2-D Vektor, oder umgekehrt, werden nachfolgend gezeigt:



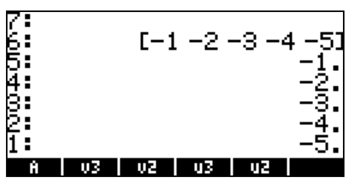
Der Versuch ein Kreuzprodukt zweier Vektoren deren Länge nicht 2 oder 3 ist, wird eine Fehlermeldung erzeugen: (Invalid Dimension), z. B. CROSS(v3,A), usw.

Zerlegen eines Vektors

Zum Zerlegen eines Vektors in seine Elemente und Komponenten wird die Funktion V→ verwendet. Wird diese im ALG-Modus benutzt, erzeugt V→ eine Liste mit den Elementen des Vektors, z. B.

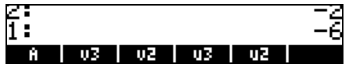


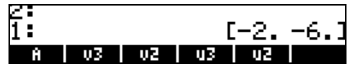
Wenn im RPN-Modus angewendet, listet die Funktion V→ die Liste der Elemente im Stack auf, z. B. erstellt V→(A) die folgende Ausgabe im RPN-Stack (Vektor A wird in Ebene 6 angezeigt).



Erstellen eines zweidimensionalen Vektors

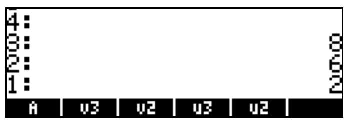
Die Funktion →V2 wird im RPN-Modus zur Erstellung eines Vektors mit den Werten in Stack-Ebene 1: und 2: verwendet. Ihre Anzeige, vor und nach Anwenden der Funktion →V2, wird wie folgt aussehen:

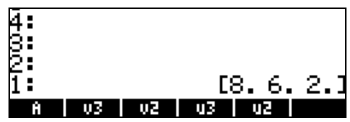




Erstellen eines dreidimensionalen Vektors

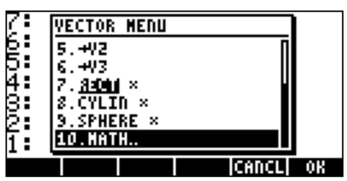
Die Funktion →V3 wird im RPN-Modus zur Erstellung eines Vektors mit den Werten in Stack-Ebene 1:, 2: und 3: verwendet. Ihre Anzeige, vor und nach Anwenden der Funktion →V2, wird wie folgt aussehen:



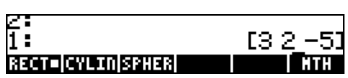


Änderung des Koordinatensystems

Um das aktuelle Koordinatensystem in ein rechtwinkliges (Kartesisches), zylindrisches (Polar) oder sphärisches zu ändern, werden die Funktionen RECT, CYLIN und SPHERE verwendet. Das aktuelle System wird entsprechend markiert in CHOOSE boxes (Systemflag 117 nicht gesetzt) oder als ausgewählt im SOFT-Menü (Systemflag 117 gesetzt) angezeigt. In der nachfolgenden Abbildung sehen Sie die Darstellung des RECT (rechtwinkligen) Koordinatensystems in diesen beiden Formaten:



Ist das rechtwinklige oder Kartesische Koordinatensystem ausgewählt, erscheint in der oberen Zeile des Displays ein Feld XYZ, und alle 2-D oder 3-D Vektoren, die sich in Ihrem Taschenrechner befinden, als Komponenten (x,y,z) des Vektors dargestellt. Um den Vektor A = 3**i**+2**j**-5**k** einzugeben, verwenden wir die Werte [3,2,-5]; der Vektor wird wie folgt angezeigt:



Um anstelle einer Kartesischen Komponente eines Vektors eine zylindrische (Polar) Komponente einzugeben, müssen wir die Magnitude r zur Verfügung stellen, die Projektion des Vektors auf die x-y Ebene, einen Winkel θ (im aktuellen Winkelmaß), welcher die Neigung von r auf die positive x-Achse darstellt, sowie eine Z-Komponente des Vektors. Dem Winkel θ muss das Winkelzeichen (∠) vorangesetzt sein, erzeugt mit den Tasten ALPHA ↱ 6. Nehmen wir z. B. an, dass wir einen Vektor r = 5, θ = 25° (DEG sollte als Winkelmaß ausgewählt sein) und z = 2,3 haben, können wir den Vektor wie folgt eingeben:

↰ [] 5 ↱ , ALPHA ↱ 6 2 5 ↱ , 2 . 3

Bevor Sie ENTER drücken, sieht die Anzeige wie die links dargestellte Abbildung aus. Nach Drücken der Taste ENTER, sieht Ihre Anzeige wie in der rechten Abbildung aus (Das numerische System wurde auf Fix mit drei Dezimalstellen geändert).

Beachten Sie, dass der Vektor in Kartesischen Koordinaten angezeigt wird, mit den Komponenten x = r cos(θ), y = r sin(θ), z = z, obwohl wir diese in Polar-Koordinatensystem eingegeben haben. Das ist deshalb so, weil die Anzeige des Vektors immer im standardmäßig eingestellten Koordinatensystem erfolgt. In diesem Fall haben wir x = 4,532, y = 2,112 und z = 2,300

Angenommen, wir geben nun einen Vektor in das sphärische Koordinatensystem ein (d. h. als (ρ,θ,φ), wobei ρ die Länge des Vektors darstellt, θ der Winkel der xy-Projektion des Vektors auf die positiven Seite der x-Achse ist und φ der von ρ und der positiven Seite der z-Achse gebildete Winkel darstellt) mit den Werten ρ = 5, θ = 25° und φ = 45°. Wir verwenden dazu: ↰ [] 5 ↱ , ALPHA ↱ 6 2 5 , ALPHA ↱ 6 4 5

Die nachfolgende Abbildung zeigt die Umwandlung des Vektors von sphärische in Kartesische Koordinaten, mit x = ρ sin(ϕ) cos(θ), y = ρ sin (ϕ) cos (θ), z = ρ cos(ϕ). In diesem Fall ist x = 3,204, y = 1,494 und z = 3,536

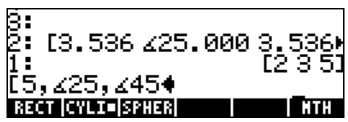
Wenn das CYLINdrical (zylindrische) System gewählt wurde, erscheint in der obersten Zeile des Displays ein Feld R∠Z und ein in zylindrischen Koordinaten eingegebener Vektor mit dessen zylindrischen (Polar) Koordinaten als (r,θ,z). Um dies zu veranschaulichen, ändern wir das Koordinatensystem auf CYLINdrical (zylindrisch), um zu sehen wie sich der Vektor in der letzten Anzeige in seine zylindrischen (Polar) Koordinaten ändert. Die zweite Komponente wird mit dem Winkelzeichen vor der Zahl (um deren rechtwinklige Eigenschaften zu betonen) dargestellt.

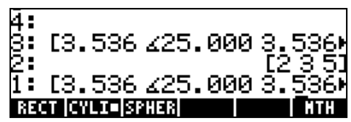
Die Konvertierung von Kartesischen auf zylindrische Koordinaten ist so, dass r = $(x^2+y^2)^{1/2}$, θ = $\tan^{-1}(y/x)$ und z = z, darstellt. Im obigen Fall wurde die Transformation so durchgeführt, dass (x,y,z) = (3,204, 2,112, 2,300) ergibt (r,θ,z) = (3,536,25°,3,536).

Geben wir nun einen Vektor in Kartesischer Form ein, obwohl das CYLINdrical (zylindrischen) Koordinatensystem aktiv ist, wir dieser in Kartesischen Koordinaten angezeigt, so z. B.

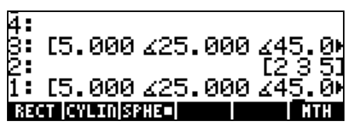
Das ist, weil die Integer-Zahlen alle für die Anwendung mit dem CAS gedacht sind, und somit die Komponenten des Vektors in Kartesischer Form erhalten bleiben. Um eine Konvertierung in Polar-Koordinaten zu erzwingen, geben Sie die Komponenten des Vektors als reelle Zahlen ein (d. h. Sie fügen einen Dezimalpunkt hinzu), so z. B. [2., 3., 5.].

Ist das zylindrische Koordinatensystem ausgewählt und wir geben einen Vektor mit sphärischen Koordinaten, wird dieser automatisch in seine zylindrischen (Polar) Äquivalente (r,θ,z) geändert, wobei r = ρ sin φ, θ = θ, z = ρ cos ist. Nachfolgend sehen Sie ein Beispiel eines Vektors, der mit sphärischen Koordinaten eingegeben und in seine Polar-Koordinaten umgewandelt wurde. In diesem Fall ρ = 5, θ = 25° und φ = 45°, während die Umwandlung anzeigt, dass r = 3,563 und z = 3,536 ist (Ändern Sie zu DEG):





Als Nächstes ändern wir das Koordinatensystem auf sphärische Koordinaten, indem wir die Funktion SPHERE aus dem Untermenü VECTOR des Menüs MTH verwenden. Sobald dieses Koordinatensystem ausgewählt wurde, wird in der obersten Zeile des Displays das R∠∠ Format angezeigt. Die letzte Anzeige wird sich wie folgt ändern:



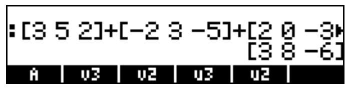
Beachten Sie, dass Vektoren die wir in zylindrischen Polar Koordinaten eingegeben haben, auf sphärisch umgeändert wurden. Die Umwandlung besteht darin, dass $\rho = (r^2+z^2)^{1/2}$, θ = θ und $\phi = \tan^{-1}(r/z)$ ist. Der Vektor der ursprünglich in Kartesischen Koordinaten dargestellt wurde, bleibt unverändert.

Anwendungen von Vektor-Operationen

In diesem Abschnitt zeigen wir Ihnen einige Beispiele von Operationen mit Vektoren, die in der Physik oder Mechanik verwendet werden.

Resultante von Kräften

Angenommen, ein Teilchen wird nachfolgenden Kräften (in N) ausgesetzt: $\mathbf{F}_1 = 3\mathbf{i}+5\mathbf{j}+2\mathbf{k}$, $\mathbf{F}_2 = -2\mathbf{i}+3\mathbf{j}-5\mathbf{k}$ und $F_3 = 2\mathbf{i}-3\mathbf{k}$. Um die Resultante zu ermitteln, d. h. die Summe all dieser Kräfte, können Sie im ALG-Modus folgenden Ansatz verwenden:



Somit ist die Resultante $R = \mathbf{F}_1 + \mathbf{F}_2 + \mathbf{F}_3 = (3\mathbf{i}+8\mathbf{j}-6\mathbf{k})$N. Im RPN-Modus verwenden Sie:

[3,5,2] ENTER [-2,3,-5] ENTER [2,0,3] ENTER + +

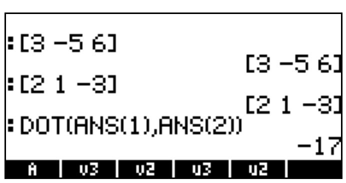
Winkel zwischen den Vektoren

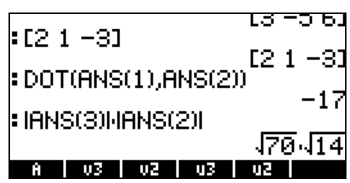
Der Winkel zwischen zwei Vektoren **A, B,** kann mithilfe der Formel $\theta = \cos^{-1}(\mathbf{A}\bullet\mathbf{B}/|\mathbf{A}||\mathbf{B}|)$ ermittelt werden.

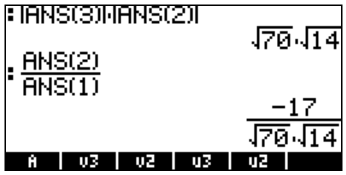
Angenommen, Sie möchten den Winkel zwischen den Vektoren $\mathbf{A} = 3\mathbf{i}-5\mathbf{j}+6\mathbf{k}$, $\mathbf{B} = 2\mathbf{i}+\mathbf{j}-3\mathbf{k}$ ermitteln, können Sie im ALG-Modus wie folgt vorgehen (Winkelmaß auf Grad eingestellt):

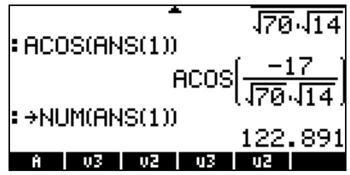
1 - Geben Sie die Vektoren [3,-5,6] ein, drücken Sie ENTER, [2,1,-3] dann ENTER.
2 - DOT(ANS(1),ANS(2)) berechnet das Skalarprodukt
3 - ABS(ANS(3))*ABS((ANS(2)) berechnet das Produkt der Magnituden
4 - ANS(2)/ANS(1) berechnet cos(θ)
5 - ACOS(ANS(1)) gefolgt von →NUM(ANS(1)), berechnet θ

Die Schritte werden in den nachfolgenden Abbildungen (natürlich im ALG-Modus) angezeigt:







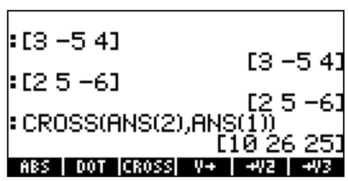


Dies ergibt das Ergebnis θ = 122,891°. Im RPN-Modus gehen Sie wie folgt vor:

[3,-5,6] ENTER [2,1,-3] ENTER DOT
[3,-5,6] ENTER ABS [2,1,-3] ENTER ABS ×
÷ ACOS →NUM

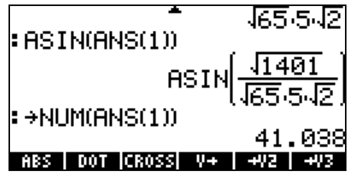
Kraftmoment

Das Moment das von einer Kraft **F** auf einen Punkt O ausgeübt wird, wird als Kreuzprodukt **M** = **r**×**F** bezeichnet, wobei **r**, auch als Kraftarm bekannt ist und die Position des Vektors in Punkt O in Richtung des Anwendungspunktes der Kraft darstellt. Angenommen, eine Kraft **F** = (2**i**+5**j**-6**k**) hat einen Kraftarm von **r** = (3**i**-5**j**+4**k**)m. Um den Moment, den diese Kraft auf den Arm ausübt zu ermitteln, verwenden wir die Funktion CROSS, wie nachfolgend gezeigt:



Somit ist **M** = (10**i**+26**j**+25**k**) m·N. Wir wissen, dass die Magnitude von **M** sich so verhält, dass |**M**| = |**r**||**F**|sin(θ), wobei θ den Winkel zwischen **r** und **F** darstellt. Wir können diesen Winkel als $\theta = \sin^{-1}$(|**M**| /|**r**||**F**|), über nachfolgende Operationen ermitteln:

1 - ABS(ANS(1))/(ABS(ANS(2))*ABS(ANS(3)) berechnet sin(θ)
2 - ASIN(ANS(1)), gefolgt von →NUM(ANS(1)) berechnet θ
Diese Rechenvorgänge werden in den nachfolgenden Abbildungen im ALG-Modus dargestellt:

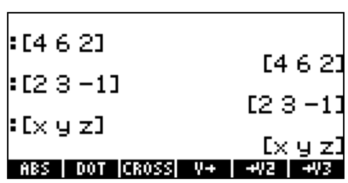


Somit beträgt der Winkel zwischen den Vektoren **r** und **F** θ = 41,038°. Im RPN-Modus können wir wie folgt vorgehen: [3,-5,4] ENTER [2,5,-6] ENTER CROSS ABS [3,-5,4] ENTER ABS [2,5,-6] ENTER ABS × ÷ ASIN →NUM

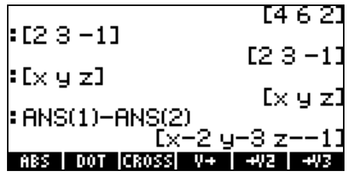
Gleichung einer Ebene im Raum

Nehmen wir an, dass wir einen Punkt $P_0(x_0,y_0,z_0)$ im Raum haben und einen Vektor **N** = N_x**i**+N_y**j**+N_z**k** normal zu einem Punkt auf dieser Ebene, welche den Punkt P_0 enthält hat, unser Problem ist es die Gleichung für diese Ebene zu finden. Wir können einen Vektor mit dem Startpunkt P_0 und dem Endpunkt an Position P(x,y,z), ein willkürlicher Punkt auf dieser Ebene, erstellen. Somit ist dieser Vektor **r** = P_0P = $(x-x_0)$**i**+ $(y-y_0)$**j** + $(z-z_0)$**k** senkrecht auf den normalen Vektor **N**, da **r** sich vollständig in der Ebene befindet. Wir haben gesehen, dass für zwei normale Vektoren **N** und **r**, **N•r** =0 ist. Somit können wir dieses Ergebnis zur Ermittlung der Gleichung der Ebene verwenden.

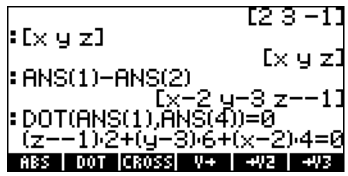
Um diesen Ansatz zu veranschaulichen, nehmen wir an der Punkt P_0 ist $P_0(2,3,-1)$ und der Normalvektor **N** = 4**i**+6**j**+2**k,** dann können wir den Vektor **N** und den Punkt P_0 als zwei Vektoren, wie unten gezeigt, eingeben: Als Letztes geben wir noch den Vektor [x,y,z] ein:



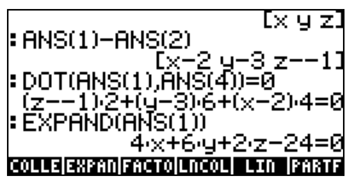
Dann berechnen wir den Vektor P_0P = **r** als ANS(1) – ANS(2), d. h.



Schließlich nehmen wir das Skalarprodukt von ANS(1) und ANS(4) und setzen dies gleich Null, um die Operation **N•r** =0 zu vervollständigen:



Nun können wir die Funktion EXPAND (im ALG-Menü) verwenden, um den Ausdruck zu expandieren (aufzufächern):



Somit ist die Gleichung der Ebene durch den Punkt P_0(2,3,-1) mit einem normalen Vektor von **N** = 4**i**+6**j**+2**k**, 4x + 6y + 2z – 24 = 0. Im RPN-Modus verwenden Sie:

`[2,3,-1] ENTER ['x','y','z'] ENTER − [4,6,2] DOT EXPAND`

Zeilen- und Spaltenvektoren sowie Listen

Alle in diesem Kapitel gezeigten Vektoren sind Zeilenvektoren. In einigen Fällen, jedoch ist es erforderlich Spaltenvektoren zu erstellen (z. B., um vordefinierte statistische Funktionen im Taschenrechner anzuwenden). Der einfachste Weg einen Spaltenvektor einzugeben, ist, jedes einzelne Element des Vektors in ein Klammerpaar zu setzen, alle zusammen befinden sich in einem weiteren Klammerpaar. So zum Beispiel, geben Sie ein:

[[1.2],[2.5],[3.2],[4.5],[6.2]] ENTER

Dies wird im nachfolgenden Spaltenvektor dargestellt:

In diesem Abschnitt zeigen wir Ihnen, wie Sie einen Spalten- in einen Zeilenvektor, einen Zeilen- in einen Spaltenvektor, eine Liste in einen Vektor und einen Vektor (oder Matrix) in eine Liste umwandeln können.

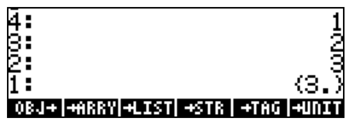
Zunächst zeigen wir diese Umwandlungen im RPN-Modus. In diesem Modus verwenden wir die Funktionen OBJ→, →LIST, →ARRY und DROP, um die Umwandlung durchzuführen. Um einen einfacheren Zugang zu diesen Funktionen zu bekommen, setzen wir das Systemflag 117 auf SOFT-Menüs (siehe Kapitel 1). Wenn dieses Flag gesetzt ist, können die Funktionen OBJ→, →ARRY und →LIST über ← PRG TYPE aufgerufen werden. Die Funktionen OBJ→, →ARRY und →LIST können über die Funktionstasten F1 , F2 und F3 aufgerufen werden. Auf die Funktion DROP kann über ← PRG STACK DROP zugegriffen werden.

Nachfolgend erläutern wir die Anwendung der Funktionen OBJ→, →LIST, →ARRY und DROP mit einigen Beispielen.

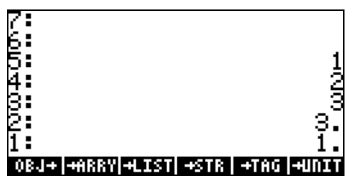
Funktion OBJ→

Diese Funktion zerlegt ein Objekt in seine Komponenten. Ist das Argument eine Liste, wird die Funktion OBJ→ die Elemente der Liste im Stack anzeigen, wobei die Anzahl der Elemente in Stack-Ebene 1 angezeigt wird, so z. B. {1,2,3} ENTER ← PRG TYPE OBJ→ ergibt:

Wird die Funktion OBJ→ auf einen Vektor angewandt, wird eine Liste mit den Elementen des Vektors im Stack angezeigt, die Anzahl der Elemente des Vektors innerhalb von Klammern (eine Liste) in Stack-Ebene 1. Folgendes Beispiel veranschaulicht diese Anwendung: [1,2,3] ENTER
← PRG TYPE OBJ→ ergibt:



Wenden wir nun die Funktion OBJ→ erneut an, wird die Liste {3.} in Stack-Ebene 1 wie folgt zerlegt:



Funktion →LIST

Diese Funktion wird zur Erstellung einer Liste eingesetzt, wenn die Elemente der Liste und die Länge oder Größe der Liste bekannt ist. Im RPN-Modus sollte die Listengröße, beispielsweise n, in Stack-Ebene 1 eingegeben werden. Die Elemente der Liste sollten in die Stack-Ebenen 2:, 3:,…, n+1: eingegeben werden. Um z. B. die Liste {1, 2, 3} zu erstellen, geben Sie Folgendes ein:

1 ENTER 2 ENTER 3 ENTER 3 ENTER ← PRG TYPE →LIST.

Funktion →ARRY

Diese Funktion wird zur Erstellung eines Vektors oder einer Matrix verwendet. In diesem Abschnitt werden wir diese zur Erstellung eines Vektors oder eines Spaltenvektors (d. h. eine Matrix aus n Zeilen und einer Spalte) verwenden. Um einen regulären Vektor zu erstellen, tragen wir die Elemente des Vektors in den Stack ein und in Stack-Ebene 1 geben wir die Vektorgröße als Liste an, z. B.:

1 ENTER 2 ENTER 3 ENTER ← {} 3 ENTER ← PRG TYPE →ARRY ein.

Um einen Spaltenvektor bestehend aus n Elementen zu erstellen, geben wir die Elemente des Vektors in den Stack und in Stack-Ebene 1 die Liste {n 1} ein. So z. B. 1 ENTER 2 ENTER 3 ENTER ← {} 1 → , 3 ENTER ← PRG TYPE →ARRY.

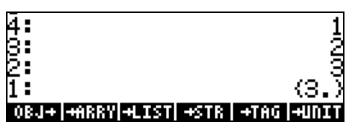
Funktion DROP

Diese Funktion hat die gleiche Wirkung wie die Löschtaste (◄).

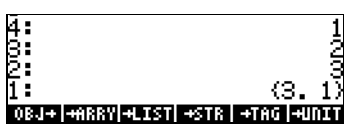
Umwandlung eines Zeilenvektors in einen Spaltenvektor

Wir veranschaulichen die Umwandlung mit dem Vektor [1,2,3]. Geben Sie diesen Vektor in den RPN-Stack, um die Übung zu verfolgen. Um einen Zeilen- in einen Spaltenvektor umzuwandeln, müssen wir die nachfolgenden Operationen im RPN-Stack durchführen:

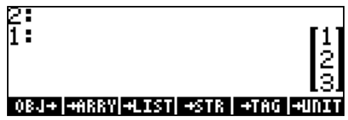
1 - den Vektor mit der Funktion OBJ→ zerlegen



2 - + 1 drücken, um die Liste in Stack-Ebene 1 von {3} in {3,1} zu ändern



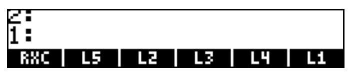
3 - die Funktion →ARRY verwenden, um den Spaltenvektor zu erzeugen



Wir können diese drei Schritte in ein UserRPL-Programm, wie nachfolgend (im RPN-Modus, immer noch) gezeigt, eingeben:

→ « » ← PRG TYPE OBJ→ 1 + →ARRY
ENTER ' ALPHA ALPHA R X C ENTER STO▶

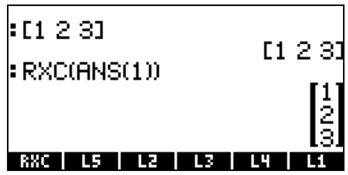
Eine neue Variable, RXC, wird nach Drücken von VAR im Funktionsmenü zur Verfügung stehen:



Drücken Sie → RXC, um das in der Variablen RCX enthaltene Programm anzuzeigen:

<< OBJ→ 1 + →ARRY >>

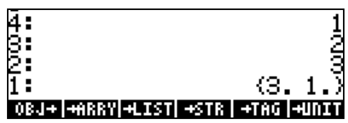
Diese Variable, RXC, kann nun zur direkten Umwandlung eines Zeilenvektors in einen Spaltenvektor verwendet werden. Geben Sie im RPN-Modus den Zeilenvektor ein, und drücken Sie anschließend RXC. Versuchen Sie z. B.: [1,2,3] ENTER RXC.
Nachdem wir nun diese Variable definiert haben, können wir sie auch im ALG-Modus dazu verwenden, einen Zeilen- in einen Spaltenvektor umzuwandeln. Ändern wir nun den Modus des Taschenrechners auf ALG und versuchen wir nachfolgende Prozedur: [1,2,3] ENTER VAR RXC ← () ← ANS ergibt:



Umwandlung eines Spaltenvektors in einen Zeilenvektor

Um diese Umwandlung zu veranschaulichen, geben wir den Spaltenvektor `[[1],[2],[3]]` im RPN-Modus ein. Gehen Sie dann wie in nachfolgender Übung gezeigt vor, um den Spalten- in einen Zeilenvektor umzuwandeln.

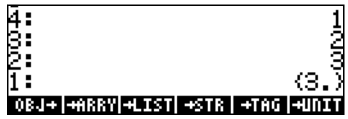
1 - verwenden Sie die Funktion OBJ→, um den Spaltenvektor zu zerlegen



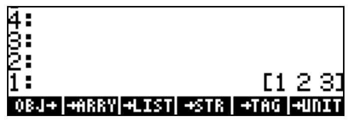
2 - verwenden Sie die Funktion OBJ→, um die Liste in Stack-Ebene 1 zu zerlegen

3 - drücken Sie die Löschtaste (auch als Funktion DROP bekannt), um die Zahl aus Stack-Ebene 1 zu entfernen

4 - verwenden Sie die Funktion →LIST, um eine Liste zu erzeugen



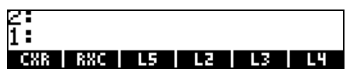
5 - verwenden Sie die Funktion →ARRY, um den Zeilenvektor zu erzeugen



Wir können diese fünf Schritte wie nachfolgend (immer noch im RPN-Modus) gezeigt, in ein UserRPL-Programm eingeben:

→ « » ← PRG TYPE OBJ→ OBJ→
← PRG STACK DROP ← PRG TYPE →LIST →ARRY ENTER
' ALPHA ALPHA C X R ENTER STO▶

Eine neue Variable, CXR, wird nach Drücken von VAR im Funktionsmenü zur Verfügung stehen:



Drücken Sie → CXR, um das in der Variablen CXR enthaltene Programm anzuzeigen:

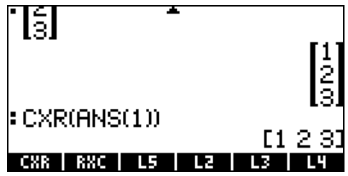
<< OBJ→ OBJ→ DROP →ARRY >>

Die Variable CXR kann nun zur direkten Umwandlung eines Spaltenvektors in einen Zeilenvektor verwendet werden. Geben Sie im RPN-Modus den Zeilenvektor ein, und drücken Sie anschließend CXR. Versuchen Sie z. B.: [[1],[2],[3]] ENTER CXR.

Nachdem wir nun die Variable CXR definiert haben, können wir sie auch im ALG-Modus dazu verwenden, einen Zeilen- in einen Spaltenvektor umzuwandeln. Ändern wir nun den Modus des Taschenrechners auf ALG und versuchen wir nachfolgende Prozedur:

[[1],[2],[3]] ENTER VAR CXR ← () ← ANS

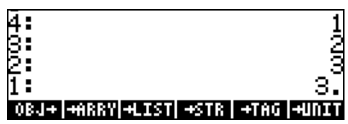
Das Resultat sieht dann so aus:



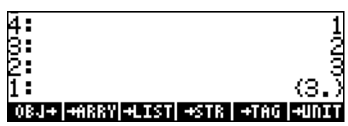
Eine Liste in einen Vektor umwandeln

Um diese Umwandlung zu veranschaulichen, geben wir die Liste {1,2,3} im RPN-Modus ein. Gehen Sie dann wie in der folgenden Übung gezeigt vor, um die Liste in einen Vektor umzuwandeln.

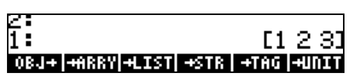
1 - verwenden Sie die Funktion OBJ→, um den Spaltenvektor zu zerlegen



2 - geben Sie eine 1 ein und verwenden dann die Funktion →LIST, um eine Liste in Stack-Ebene 1 zu erstellen.



3 - verwenden Sie die Funktion →ARRY, um den Vektor zu erzeugen



Wir können diese drei Schritte wie nachfolgend (im RPN-Modus) gezeigt in ein UserRPL-Programm eingeben:

↱ « » ↰ PRG TYPE OBJ→ OBJ→ / →LIST →ARRY ENTER
' ALPHA ALPHA L X V ENTER STO▸

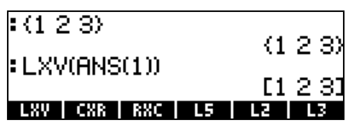
Eine neue Variable, LXV, wird nach Drücken der Taste VAR unter den Funktionstasten zur Verfügung stehen.

Drücken Sie ↱ LXV, um das in der Variablen LVX enthaltene Programm anzuzeigen:

<< OBJ→ 1 →LIST →ARRY >>

Die Variable LXV kann nun zur direkten Umwandlung einer Liste in einen Vektor verwendet werden. Geben Sie im RPN-Modus den Zeilenvektor ein, und drücken Sie anschließend LXV. Versuchen Sie z. B.: {1,2,3} ENTER LXV.

Nachdem wir nun die Variable LXV definiert haben, können wir sie auch im ALG-Modus dazu verwenden, eine Liste in einen Vektor umzuwandeln. Ändern wir nun den Modus des Taschenrechners auf ALG, und versuchen wir folgendes Verfahren: {1,2,3} ENTER VAR LXV ↰ () ↰ ANS , ergibt:

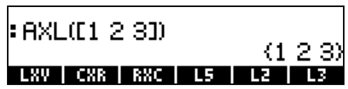


Einen Vektor oder eine Matrix in eine Liste umwandeln

Im Taschenrechner wird die Funktion AXL zur Umwandlung eines Vektors in eine Liste bereitgestellt. Diese Funktion können Sie aus dem Befehls-Katalog wie folgt aufrufen:

↱ CAT ALPHA ALPHA A X L ALPHA OK

Als Beispiel wenden Sie im RPN-Modus die Funktion AXL auf den Vektor [1,2,3] unter Verwendung von [1,2,3] ENTER AXL an. Die folgende Anzeige zeigt die Anwendung der Funktion AXL auf den gleichen Vektor im ALG-Modus.



Kapitel 10
Erstellen und Manipulieren von Matrizen

In diesem Kapitel finden Sie Beispiele zur Erstellung von Matrizen im Taschenrechner und zur Veranschaulichung der Manipulation von Matrizen-Elementen.

Definitionen

Bei einer <u>Matrix</u> handelt es sich ganz einfach um ein rechtwinkliges Array von Objekten (d. h. Zahlen, Algebraiks), bestehend aus mehreren Zeilen und Spalten. Eine Matrix **A** mit n Zeilen und m Spalten enthält somit n×m Elemente. Ein generisches Element einer Matrix wird durch die indexierte Variable a_{ij}, welche der Zeile i und Spalte j entspricht, dargestellt. Anhand dieser Notation kann die Matrix **A** als **A** = $[a_{ij}]_{n\times m}$ definiert werden. Nachfolgend die gesamte Matrix:

$$\mathbf{A} = [a_{ij}]_{n\times m} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \vdots & \vdots & \ddots & \\ a_{n1} & a_{n2} & \cdots & a_{nm} \end{bmatrix}.$$

Eine Matrix ist hermitisch (quadratisch), wenn m = n zutrifft. Das <u>Transponieren</u> einer Matrix besteht darin, die Zeilen gegen die Spalten auszutauschen und umgekehrt. Somit ist die Transponierte der Matrix **A**, $\mathbf{A}^T$ = $[(a^T)_{ij}]_{m\times n} = [a_{ji}]_{m\times n}$. Die <u>Hauptdiagonale</u> einer hermitischen Matrix ist die Menge der Elemente a_{ii}. Eine <u>Identitätsmatrix</u>, $\mathbf{I}_{n\times n}$, ist eine hermitische Matrix, deren Elemente der Hauptdiagonale alle 1, während alle weiteren Elemente außerhalb der Diagonalen Null sind. So wird z. B. eine Identitätsmatrix 3×3 wie folgt definiert:

$$\mathbf{I} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Auch kann eine Identitätsmatrix als $\mathbf{I}_{n\times n} = [\delta_{ij}]$ dargestellt werden, wobei δ_{ij}, eine Funktion, bekannt als <u>Kroneckers Delta</u>, darstellt und wie folgt definiert ist:

$$\delta_{ij} = \begin{cases} 1, & if \ i = j \\ 0, & if \ i \neq j \end{cases}.$$

Eingaben von Matrizen in den Stack

In diesem Abschnitt werden zwei unterschiedliche Methoden zur Eingabe von Matrizen in den Stack des Taschenrechners gezeigt: (1) mithilfe des Matrix Editors und (2) durch direktes Eingeben der Matrix in den Stack.

Verwendung des Matrix Editors

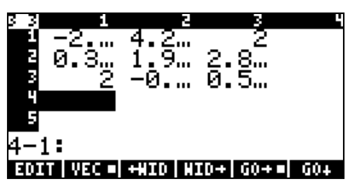
Analog zu Vektoren, wie in Kapitel 9 beschrieben, können Matrizen mithilfe des Matrix Editors in den Stack eingegeben werden. Um z. B. die folgende Matrix einzugeben:

$$\begin{bmatrix} -2.5 & 4.2 & 2.0 \\ 0.3 & 1.9 & 2.8 \\ 2 & -0.1 & 0.5 \end{bmatrix},$$

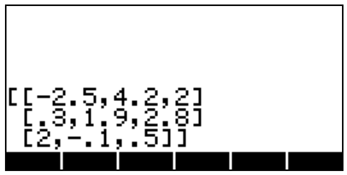
Starten Sie zuerst den Matrix Editor über ← MTRW . Stellen Sie sicher, dass die Option ▦→■ gewählt ist. Verwenden Sie dazu die nachstehende Tastenfolge:

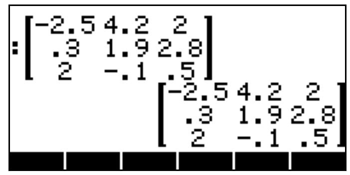
2 • 5 +/- ENTER 4 • 2 ENTER 2 ENTER ▼ ◄ ◄ ◄
• 3 ENTER 1 • 9 ENTER 2 • 8 ENTER
2 ENTER • 1 +/- ENTER • 5 ENTER

An dieser Stelle wird Ihre Anzeige wie folgt dargestellt:

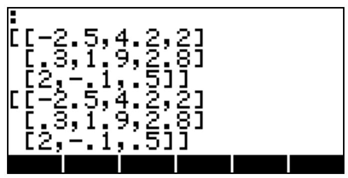


Drücken Sie ENTER ein weiteres Mal, um die Matrix in den Stack zu verschieben. Nachfolgend wird der Stack im ALG-Modus, vor und nach dem zweiten Mal drücken gezeigt:





Bei ausgewählter Textbuch-Anzeige (über MODE DISP und ✓Textbook angekreuzt), wird die Matrix wie oben angezeigt, andernfalls folgendermaßen:



Im RPN-Modus wird die Anzeige annähernd gleich dargestellt.

Anmerkung: Der Matrix Writer wurde in Kapitel 9 ausführlich erklärt.

Die Matrix direkt in den Stack eingeben

Dasselbe Ergebnis wie oben wird erzielt, wenn Nachfolgendes direkt in den Stack eingeben wird:

⇦ [ ]
⇦ [ ] 2 . 5 +/- ⇨ , 4 . 2 ⇨ , 2 ▶
⇨ ,
⇦ [ ] . 3 ⇨ , 1 . 9 ⇨ , 2 . 8 ▶
⇨ ,
⇦ [ ] 2 ⇨ , . 1 +/- ⇨ , . 5

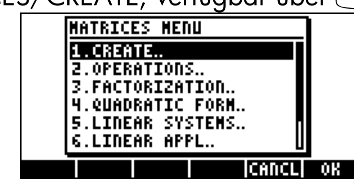
Um eine Matrix direkt in den Stack einzugeben, öffnen Sie ein Klammerpaar (⇦ [ ]), und schließen Sie jede Zeile der Matrix in ein weiteres Klammernpaar (⇦ [ ]) ein. Die Elemente der Matrix müssen durch Komma (⇨ , .) voneinander getrennt werden, gleichermaßen die Klammern zwischen den Zeilen. (**Anmerkung**: Im RPN-Modus können Sie die inneren Klammern nach Eingabe des ersten Zahlenpaares aussparen, so können Sie z. B. anstelle von [[1 2 3] [4 5 6] [7 8 9]] einfach [[1 2 3] 4 5 6 7 8 9] eingeben.)

Speichern Sie diese Matrix nun für spätere Übungen unter dem Namen A. Verwenden Sie hierzu im ALG-Modus [STO▶][ALPHA][A], im RPN-Modus ['][ALPHA][A] [STO▶].

Erstellen von Matrizen mit den Funktionen des Rechners

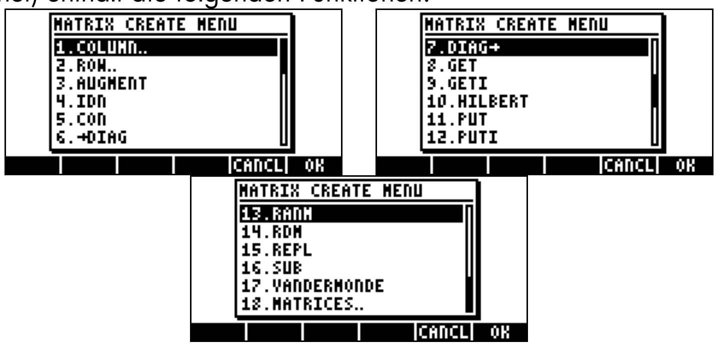
Einige Matrizen können mit den bestehenden Funktionen des Taschenrechners, die entweder über das Untermenü MTH/MATRIX/MAKE oder über das MTH-Menü ([←] MTH) zur Verfügung stehen, erstellt werden,

oder im Menü MATRICES/CREATE, verfügbar über [←] MATRICES :



Das Untermenü MTH/MATRIX/MAKE (der Einfachheit halber als Menü MAKE bezeichnet) enthält die folgenden Funktionen:

Das Untermenü MATRICES/CREATE (der Einfachheit halber als Menü CREATE bezeichnet) enthält die folgenden Funktionen:

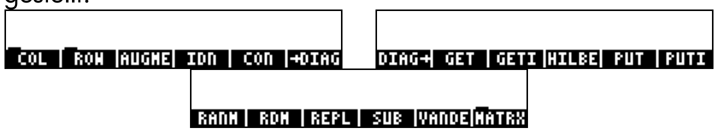


Wenn Sie die Menüs (MAKE und CREATE) näher betrachten, werden Sie feststellen, dass beide die gleichen Funktion enthalten (GET, GETI, PUT, PUTI, SUB, REPL, RDM, RANM, HILBERT, VANDERMONDE, IDN, CON, →DIAG und DIAG→). Im Menü CREATE finden Sie die Untermenüs COLUMN (Spalte) und ROW (Zeile), welche Sie jedoch auch im Menü MTH/MATRIX finden. Im Menü MAKE sind die Funktionen SIZE enthalten, die im Menü CREATE hingegen nicht enthalten sind. Im Allgemeinen jedoch stellen beide Menüs, MAKE und CREATE, die gleichen Funktionen zur Verfügung. In den nachfolgenden Beispielen wird der Zugriff der Funktionen über das Menü MAKE erörtert. Am Ende dieses Abschnittes finden Sie eine Tabelle mit den entsprechenden Tastenfolgen, die erforderlich sind, um auch dann die gleichen Funktionen über das Menü CREATE zu erhalten, wenn das Systemflag 117 auf SOFT-Menü eingestellt ist.

Ist das Systemflag (Flag 117) auf SOFT-Menü eingestellt, kann das Menü MAKE über die nachfolgende Tastenfolge gestartet werden: ← MTH MATRX MAKE

Die zur Verfügung stehenden Funktionen werden als Funktionstasten des Menüs angezeigt (drücken Sie NXT, um den nächsten Satz von Funktionen anzuzeigen):

Ist Systemflag 117 auf SOFT-Menüs eingestellt, können die Funktionen des Menüs CREATE über ← MATRICES CREAT ausgewählt werden und werden wie folgt dargestellt:

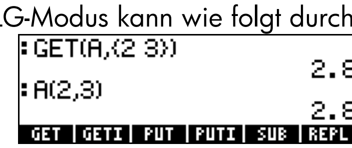


In den nächsten Abschnitten wird die Anwendung der Matrix-Funktionen im Menü MAKE und CREATE vorgestellt.

Funktionen GET und PUT

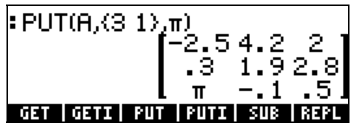
Die Funktionsweise von GET, GETI, PUT und PUTI mit Matrizen ist mit derjenigen mit Listen oder Vektoren vergleichbar, d. h., Sie müssen die Position der Elemente, welche Sie mit GET oder PUT verwenden möchten, angeben. Während jedoch in Listen und Vektoren lediglich ein Index zur Identifizierung eines Elementes erforderlich ist, benötigen Sie in Matrizen zwei Indizes {Zeile, Spalte} zur Identifizierung der Elemente in der Matrix. Nachfolgend Anwendungsbeispiele für GET und PUT:

Die oben gespeicherte Matrix wird in die Variable A aufgenommen, um die Funktionen GET und PUT zu veranschaulichen. Die Extraktion des Elements a_{23} aus der Matrix A im ALG-Modus kann wie folgt durchgeführt werden:



Beachten Sie dabei, dass das gleiche Ergebnis erzielt wird, wenn A(2,3) eingegeben und anschließend ENTER gedrückt wird. Im RPN-Modus erfolgt dies, wenn A ENTER 3 ENTER GET oder A(2,3) ENTER eingegeben wird.

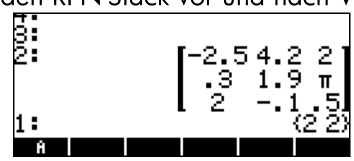
Angenommen, es soll der Wert 'π' in Element a_{31} der Matrix eingegeben werden. Dazu wird die Funktion PUT verwendet, z. B folgendermaßen.

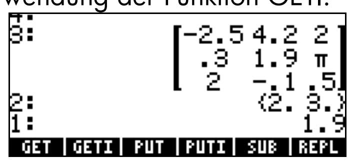


Im RPN-Modus kann dies auf folgende Weise erfolgen: VAR A {3,1} ENTER ← π PUT. Alternativ kann im RPN-Modus auch Nachfolgendes eingegeben werden: ← π ' A(2,3) ENTER STO▸. Um den Inhalt der Variablen A anzuzeigen, drücken Sie A.

Funktionen GETI und PUTI

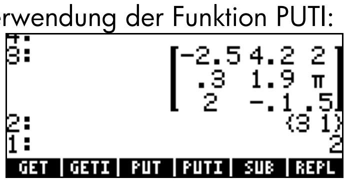
Die Funktionen PUTI und GETI werden in UserRPL-Programmen verwendet, da sie in der Lage sind, einen Index für wiederholte Verwendung von PUT und GET zu speichern. Die Liste von Indizes in Matrizen variiert zunächst in Spalten. Um die Anwendung zu veranschaulichen, wird nachfolgende Übung im RPN-Modus vorgeschlagen: A {2,2} ENTER GETI. Folgende Abbildungen zeigen den RPN-Stack vor und nach Verwendung der Funktion GETI:

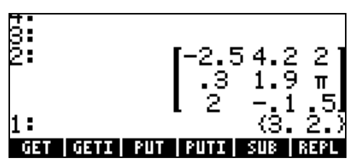




Beachten Sie dabei, dass die Abbildung für die anschließende Verwendung der Funktion GETI oder GET zur Erhöhung des Spaltenindex der ursprünglichen Referenz um 1 (d. h. von {2,2} auf {2,3}) bereit ist, während gleichzeitig der extrahierte Wert, A(2,2) = 1,9, in Stack-Ebene 1 angezeigt wird.

Angenommen, dass Sie den Wert 2 in Element {3 1} mithilfe der Funktion PUTI eingeben möchten. Während Sie sich nach wie vor im RPN-Modus befinden, probieren Sie die nachfolgende Tastefolge aus: ◄ ◄ {3 1} ENTER 2 ENTER PUTI. Folgende Abbildungen zeigen den RPN-Stack vor und nach Verwendung der Funktion PUTI:

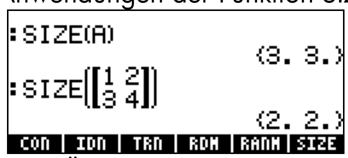




In diesem Fall wurde die 2 in Position {3 1} ersetzt, d. h. jetziger Wert A(3,1) = 2 und die Indexliste um 1 (Spalte zuerst), d. h. von {3,1} auf {3,2} erhöht. Die Matrix befindet sich in Ebene 2 und die um einen Schritt erhöhte Indexliste in Ebene 1.

Funktion SIZE

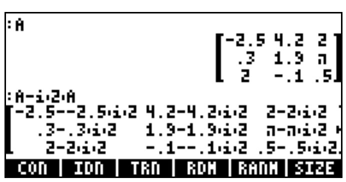
Die Funktion SIZE stellt eine Liste bereit, in welcher die Anzahl der Zeilen und Spalten der Matrix in Stack Ebene 1 angezeigt wird. Die nachfolgende Abbildung zeigt einige Anwendungen der Funktion SIZE im ALG-Modus:

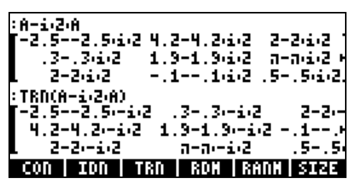


Im RPN-Modus werden diese Übungen mithilfe von A SIZE und [[1,2],[3,4]] ENTER SIZE durchgeführt.

Funktion TRN

Die Funktion TRN wird zur Erstellung der Transkonjugierten einer Matrix, d. h. der Transponierten (TRAN), gefolgt von der Komplex-Konjugierten (CONJ) verwendet. In den nachfolgenden Abbildungen sehen Sie die ursprüngliche Matrix der Variablen A und deren Transponierte, zur vollständigen Anzeige in Kleinschrift angezeigt (siehe Kapitel 1).



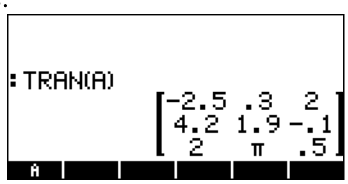


Ist das Argument eine reelle Matrix, erstellt TRN einfach die Transponierte der reellen Matrix. Versuchen Sie z. B. TRN(A), und vergleichen Sie das Ergebnis mit TRAN(A).

Im RPN-Modus wird die Transkonjugierte einer Matrix **A** über A TRN ermittelt.

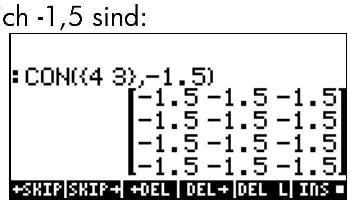
Anmerkung: Im Taschenrechner steht auch die Funktion TRAN im Untermenü MATRICES/OPERATIONS zur Verfügung:

So z. B. im ALG-Modus:



Funktion CON

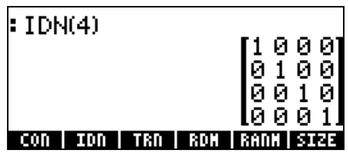
Das Argument der Funktion ist eine Liste zweier Elemente, entsprechend der Anzahl der Zeilen und Spalten der zu erzeugenden Matrix und einem konstanten Wert. Die Funktion CON erstellt eine Matrix mit konstanten Elementen. So erzeugt z. B. der Befehl im ALG-Modus eine 4×3 Matrix, deren Elemente alle gleich -1,5 sind:



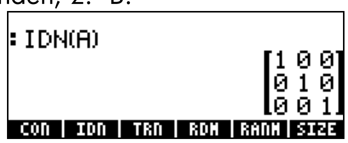
Im RPN-Modus wird das gleiche Ergebnis über {4,3} ENTER 1 . 5 +/- ENTER CON erzielt.

Funktion IDN

Die Funktion IDN (IDeNtity matrix) erstellt eine Identitätsmatrix von vorgegebener Größe. Beachten Sie, dass es sich bei einer Identitätsmatrix um eine hermitische Matrix handeln muss. Es wird nur ein Wert benötigt, um diese vollständig zu beschreiben. Um z. B. eine Identitätsmatrix von 4×4 im ALG-Modus zu erstellen, verwenden Sie:



Sie können jedoch ebenso eine bestehende hermitische Matrix als Argument der Funktion IDN verwenden, z. B.



Die erhaltene Identitätsmatrix verfügt über die gleichen Dimensionen wie das Argument der Matrix. Beachten Sie dabei, dass der Versuch, eine rechteckige Matrix (d. h. nicht hermitisch – quadratisch) als Argument von IDN zu erstellen, eine Fehlermeldung erzeugt.

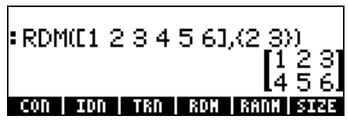
Im RPN-Modus wurden die beiden oben genannten Beispiele mithilfe von 4 ENTER IDN und A IDN erzeugt.

Funktion RDM

Die Funktion RDM (Re-DiMensioning) wird zur Konvertierung von Vektoren und Matrizen in Matrizen und Vektoren verwendet. Die Eingabe zur Funktion besteht aus dem ursprünglichen Vektor oder der Matrize, gefolgt von einer Liste, bestehend aus einer einzigen Zahl, wenn in einen Vektor konvertiert werden soll, bzw. aus zwei Zahlen, wenn die Konvertierung in eine Matrix erfolgen soll. Im vorangegangenen Fall stellt die Zahl die Dimension des Vektors dar, im letzteren Fall die Zahl der Zeilen und Spalten der Matrix. Die nachfolgenden Beispiele veranschaulichen die Anwendung der Funktion RDM:

Umdimensionieren eines Vektors in eine Matrix

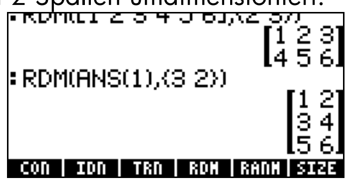
Die nachfolgenden Beispiele veranschaulichen, wie im ALG-Modus ein Vektor, bestehend aus 6 Elementen, in eine Matrix von 2 Zeilen und 3 Spalten umdimensioniert wird:



Um die obige Matrix im RPN-Modus zu erstellen, kann hierzu die Abfolge [1,2,3,4,5,6] ENTER {2,3} ENTER RDM verwendet werden.

Umdimensionieren einer Matrix in eine andere Matrix

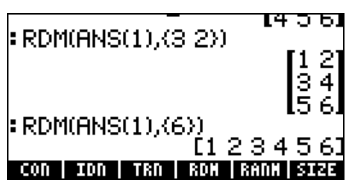
Im ALG-Modus wird die oben erstellte Matrix verwendet und diese in eine Matrix von 3 Zeilen und 2 Spalten umdimensioniert:



Im RPN-Modus wird ganz einfach {3,2} ENTER RDM verwendet.

Umdimensionieren einer Matrix in einen Vektor

Um eine Matrix in einen Vektor umzudimensionieren, verwenden Sie als Argumente die Matrix, gefolgt von einer Liste, die die Anzahl der Elemente in der Matrix enthält. Um beispielsweise die Matrix aus vorangegangenem Beispiel in einen Vektor der Länge 6 im ALG-Modus zu erstellen, verwenden Sie:

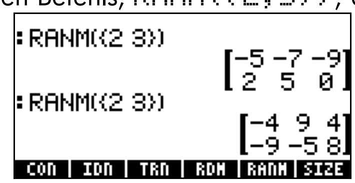


Im RPN-Modus verwenden Sie {6} ENTER RDM, vorausgesetzt, die Matrix befindet sich im Stack.

Anmerkung: Die Funktion RDM stellt einen direkteren und effizienteren Weg zur Umwandlung von Listen in Arrays und umgekehrt als diejenige Möglichkeit dar, die am Ende von Kapitel 9 beschrieben wurde.

Funktion RANM

Die Funktion RANM (RANdom Matrix) erstellt eine Matrix mit zufällig erzeugten Integer-Elementen, mit vorgegebener Liste der Anzahl der Zeilen und Spalten (die Dimensionen der Matrix). So werden z. B. im ALG-Modus zwei verschiede 2×3 Matrizen mit zufällig erzeugten Elementen unter Verwendung des gleichen Befehls, RANM({2,3}), erzeugt.

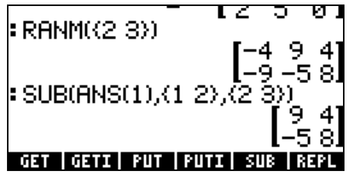


Im RPN-Modus verwenden Sie {2,3} ENTER RANM.

Offensichtlich weichen die Ergebnisse aus Ihrem Taschenrechner von den oben gezeigten Ergebnissen ab. Die erzeugten Zufallszahlen sind Integer-Werte, gleichmäßig zwischen [-10,10] erzeugt, d. h. jede einzelne dieser 21 Zahlen kann mit derselben Wahrscheinlichkeit ausgewählt werden. Die Funktion RANM ist bei der Erstellung von Matrizen jeder Größe hilfreich, um Operationen mit Matrizen oder die Anwendung von Matrix-Funktionen veranschaulichen zu können.

Funktion SUB

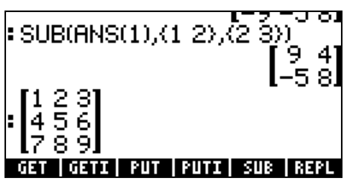
Die Funktion SUB extrahiert eine Untermatrix aus einer bestehenden Matrix, vorausgesetzt, Sie geben den Anfangs- und Endwert der Untermatrix an. Wenn wir beispielsweise die Elemente a_{12}, a_{13}, a_{22} und a_{23} aus dem letzten Ergebnis als eine Matrix 2×2 extrahieren möchten, verwenden wir im ALG-Modus:

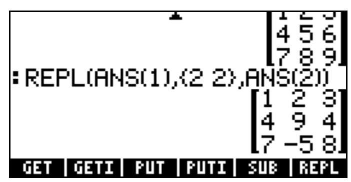


Im RPN-Modus, vorausgesetzt, die ursprüngliche Matrix 2×3 befindet sich bereits im Stack, verwenden wir {1,2} ENTER {2,3} ENTER SUB.

Funktion REPL

Die Funktion REPL ersetzt oder fügt eine Untermatrix in eine größere Matrix ein. Die Eingabe für diese Funktion ist die Matrix, in welcher der Austausch erfolgen soll, die Position an welcher dieser Austausch zu erfolgen hat und die einzufügende Matrix. Als Beispiel nehmen wir die Matrix, die wir aus vorangegangenem Beispiel erhalten haben, und geben die Matrix [[1,2,3],[4,5,6],[7,8,9]] ein. In der nachfolgenden linken Abbildung ist die neue Matrix im ALG-Modus vor dem Drücken der Taste ENTER zu sehen. In der rechten Abbildung ist die Anwendung der Funktion RPL zum Ersetzen der Matrix 2×2 in ANS(2) in die Matrix 3×3, die sich im Moment in ANS(1) befindet, Anfangsposition in {2,2} zu sehen:





Wenn Sie im RPN-Modus arbeiten gehen wir wie folgt vor, vorausgesetzt, die Matrix 2×2 befindet sich im Stack:

[[1,2,3],[4,5,6],[7,8,9]] ENTER ▶ (diese letzte Taste vertauscht die Inhalte der Stack-Ebene 1 mit 2) {1,2} ENTER ▶ (ein weiterer Austausch der Ebene 1 und 2) REPL.

Funktion →DIAG

Die Funktion →DIAG nimmt die Hauptdiagonale einer hermitischen Matrix mit den Dimensionen n×n und erstellt einen Vektor mit der Dimension n, der die Elemente der Hauptdiagonalen enthält. So können wir z. B. für die Matrix, die uns aus vorangegangenem Beispiel bleibt, die Hauptdiagonale wie folgt extrahieren:

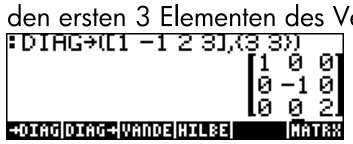
Im RPN-Modus, die Matrix 3×3 befindet sich im Stack, müssen wir einfach nur die Funktion →DIAG starten, um das gleiche Ergebnis wie oben zu erzielen.

Funktion DIAG→

Die Funktion DIAG→ nimmt einen Vektor und eine Liste von Matrix-Dimensionen {Zeilen, Spalten} und erstellt eine Diagonalmatrix mit einer Hauptdiagonale, deren Elemente mit den richtigen Vektorelementen ersetzt wurde. So erzeugt z. B. der Befehl

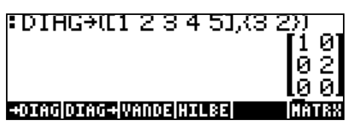
DIAG→([1,-1,2,3],{3,3})

eine Diagonalmatrix mit den ersten 3 Elementen des Vektorargumentes:



Im RPN-Modus können wir [1,-1,2,3] ENTER {3,3} ENTER DIAG→ verwenden, um das gleiche Ergebnis wie oben zu erzielen.

Ein weiteres Beispiel zur Anwendung der Funktion DIAG→ wird nun im ALG-Modus gezeigt:



Im RPN-Modus verwenden wir dafür [1,2,3,4,5] ENTER {3,2} ENTER DIAG→.

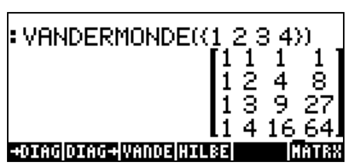
In diesem Fall soll eine 3×2 Matrix, mit so vielen Elementen des Vektors [1,2,3,4,5] wie möglich als Hauptdiagonalelemente erzeugt werden. Die Hauptdiagonale für eine rechtwinklige Matrix beginnt in Position (1,1) und bewegt sich weiter zu (2,2), (3,3) usw. bis entweder die Anzahl der Zeilen oder der Spalten aufgebraucht ist. In diesem Fall wurde die Anzahl der Spalten (2) vor der Anzahl der Zeilen (3) aufgebraucht, sodass die Hauptdiagonale nur die Elemente in den Positionen (1,1) und (2,2) enthält. Somit wurden lediglich die ersten beiden Elemente des Vektors zur Bildung der Hauptdiagonale benötigt.

Funktion VANDERMONDE

Die Funktion VANDERMONDE erzeugt die Vandermonde-Matrix der Dimension n, basierend auf einer vorgegebenen Liste von Eingabedaten. Die Dimension n stellt natürlich die Länge der Liste dar. Besteht die Eingabeliste aus den Objekten {x_1, x_2,… x_n}, dann ist eine Vandermonde-Matrix im Taschenrechner eine Matrix, die aus folgenden Elementen besteht:

$$\begin{bmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^{n-1} \\ 1 & x_2 & x_2^2 & \cdots & x_2^{n-1} \\ 1 & x_3 & x_3^2 & \cdots & x_3^{n-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^{n-1} \end{bmatrix}$$

Als Beispiel geben Sie folgenden Befehl im ALG-Modus für die Liste {1,2,3,4} ein:



Im RPN-Modus geben Sie `{1,2,3,4}` ENTER `VANDERMONDE` ein.

Funktion HILBERT

Die Funktion HILBERT erstellt die Hilbert-Matrix für eine Dimension n. Die n×n Hilbert-Matrix $\mathbf{H}_n = [h_{jk}]_{n\times n}$, verhält sich wie folgt

$$h_{jk} = \frac{1}{j+k-1}$$

Die Hilbert-Matrix wird zur Anpassung numerischer Kurven durch die lineare Quadrat-Methode verwendet.

Programm zur Erstellung einer Matrix aus einer Anzahl von Listen

In diesem Abschnitt stellen wir einige UserRPL-Programme zur Erstellung einer Matrix aus einer Anzahl von Listen von Objekten zur Verfügung. Die Listen können Spalten (Programm CRMC) oder Zeilen der Matrix (Programm CRMR) darstellen. Die Programme werden im RPN-Modus eingegeben und die Anweisungen der Tastenanschläge für Systemflag 117 auf SOFT-Menüs gesetzt. Dieser Abschnitt ist für Sie als Übung gedacht, um sich mit den Programmierfunktionen des Taschenrechners vertraut zu machen. Die Programme werden unten aufgelistet, auf der linken Seite sind die zum Starten der Programmschritte erforderlichen Tastenanschläge zu finden, während auf der rechten Seite die Zeichen zu sehen sind, die ins Display eingegeben werden, um jene Tastenanschläge durchzuführen. Zunächst stellen wir die zur Erstellung des Programms CRMC erforderlichen Schritte vor.

Die Listen stellen Spalten der Matrix dar

Das Programm CRMC ermöglicht Ihnen eine p×n Matrix (d. h. p Zeilen und n Spalten), jeweils bestehend aus n Einträgen von p Elementen, zu erstellen. Um dieses Programm zu erstellen, verwenden Sie folgende Tastenanschläge:

Tastenfolge:	erzeugt:
↱ « »	«
↰ PRG STACK DUP	DUP
↱ → SPC ALPHA ↰ N	→ n
↱ « »	<<
1 ↰ PRG STACK SWAP	1 SWAP

[←][PRG] BRCH FOR FOR	FOR
[ALPHA][←][J]	j
[←][PRG] TYPE OBJ→	OBJ→
→ARRY	→ARRY
[←][PRG] BRCH IF IF	IF
[ALPHA] [←][J][SPC]	j
[ALPHA] [←][N]	n
[←][PRG] TEST <	<
[←][PRG] BRCH IF THEN	THEN
[ALPHA] [←][J] [SPC][1][+]	j 1 +
[←][PRG] STACK [NXT] ROLL	ROLL
[←][PRG] BRCH IF END	END
[←][PRG] BRCH FOR NEXT	NEXT
[←][PRG] BRCH IF IF	IF
[ALPHA] [←][N] [SPC][1]	n 1
[←][PRG] TEST >	>
[←][PRG] BRCH IF THEN	THEN
[1][SPC]	1
[ALPHA] [←][N] [SPC][1][−]	n 1 -
[←][PRG] BRCH FOR FOR	FOR
[ALPHA] [←][J] [SPC]	j
[ALPHA] [←][J] [SPC][1][+]	j 1 +
[←][PRG] STACK [NXT] ROLL	ROLL
[←][PRG] BRCH FOR NEXT	NEXT
[←][PRG] BRCH IF END	END
[ALPHA][←][N] [SPC]	n
[←][MTH] MATRX COL COL→	COL→
[ENTER]	Das Programm wird in Ebene 1 angezeigt

Zum Sichern des Programms ['][ALPHA][ALPHA][C][R][M][C][ALPHA] [STO▶]

Anmerkung: Wenn Sie dieses Programm im HOME-Verzeichnis speichern, kann dieses von jedem Unterverzeichnis aus aufgerufen werden.

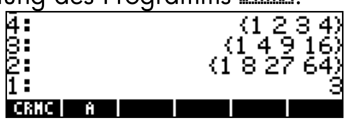
Um sich den Inhalt des Programms anzusehen, drücken Sie [VAR][→] CRMC. Das Programm-Listing sieht wie folgt aus:

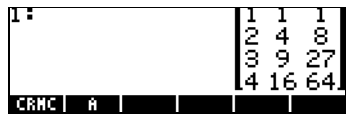
```
« DUP → n « 1 SWAP FOR j OBJ→ →ARRY IF j n < THEN j 1 +
 ROLL END NEXT IF n 1 > THEN 1 n 1 - FOR j j 1 + ROLL
 NEXT END n COL→ » »
```

Um dieses Programm im RPN-Modus zu verwenden, geben Sie die n Einträge in der Reihenfolge, in welcher diese als Spalten der Matrix dargestellt werden sollen, ein, tragen dann den Wert n ein und drücken CRMC. Als Beispiel versuchen Sie nachfolgende Übung:

`{1,2,3,4}` ENTER `{1,4,9,16}` ENTER `{1,8,27,64}` ENTER 3 ENTER CRMC

Ihre Anzeige wird im RPN Stack wie folgt angezeigt – vor und nach Anwendung des Programms CRMC:

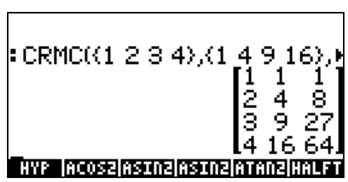




Um dieses Programm im ALG-Modus zu verwenden, drücken Sie CRMC, gefolgt von einem Klammerpaar (← ()). Innerhalb der Klammern geben Sie die Einträge, welche die Spalten der Matrix darstellen, durch Kommas getrennt ein, und schließlich ein weiteres Komma sowie die Anzahl der Spalten. Der Befehl sollte wie folgt aussehen:

`CRMC({1,2,3,4}, {1,4,9,16}, {1,8,27,64}, 3)`

Nachfolgend ist die Ausführung des Programms CRMC im ALG-Modus zu sehen:



Die Listen stellen Zeilen der Matrix dar

Das vorangegangene Programm kann leicht angepasst werden, um eine Matrix zu erstellen, wenn die Eingabedaten Zeilen der fertigen Matrix werden sollen. Die einzige Änderung, die Sie im Programm-Listing vornehmen müssen, ist COL→ mit ROW→ auszutauschen. Um diese Änderung vorzunehmen, gehen Sie wie folgt vor:

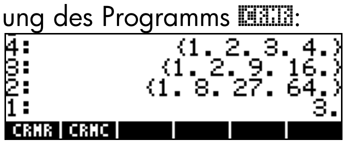
↱ CRMC	Das Programm CRMC im Stack listen
▼ ↱ ▼ ▲ ◀ ◀ ◀	Ans Ende des Programms gehen
⇐ ⇐ ⇐	Löschen von COL
ALPHA ALPHA R O W ALPHA ENTER	ROW eintippen, ins Programm gehen

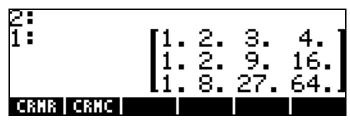
Zum Speichern des Programms verwenden Sie:

' ALPHA ALPHA C R M R ALPHA STO▶

{1,2,3,4} ENTER {1,4,9,16} ENTER {1,8,27,64} ENTER 3 ENTER CRMR

Ihre Anzeige wird im RPN-Stack wie folgt aussehen – vor und nach Anwendung des Programms CRMR:

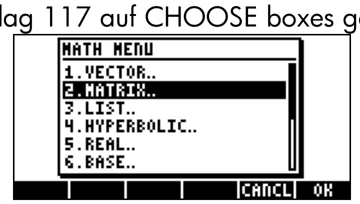


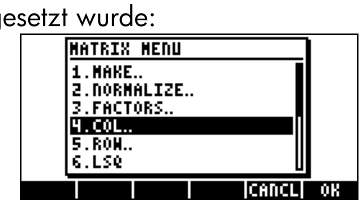


Diese Programme werden hauptsächlich bei statistischen Anwendungen verwendet, im Speziellen jedoch bei der Erstellung der Statistik-Matrix ΣDAT. Beispiele zur Anwendung dieses Programms werden in einem späteren Kapitel gezeigt.

Manipulation der Spalten von Matrizen

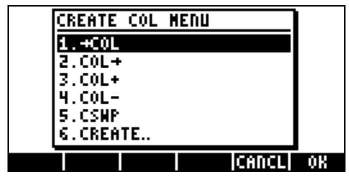
Der Taschenrechner enthält ein Menü mit Funktionen zur Manipulation von Matrizen in deren Spalten. Dieses Menü kann wie in den folgender Abbildungen über MTH/MATRIX/COL.. (↰ MTH) aufgerufen werden, wenn Systemflag 117 auf CHOOSE boxes gesetzt wurde:



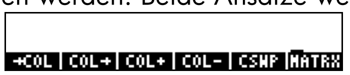


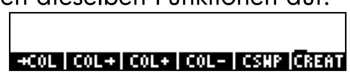
oder über das Untermenü MATRICES/CREATE/COLUMN:

Beide Ansätze weisen die gleichen Funktionen auf:



Ist das Systemflag 117 auf SOFT-Menüs gesetzt, kann das Menü COL entweder über ⇦ MTH MATRX MAKE COL oder über ⇦ MATRICES CREAT COL aufgerufen werden. Beide Ansätze weisen dieselben Funktionen auf:

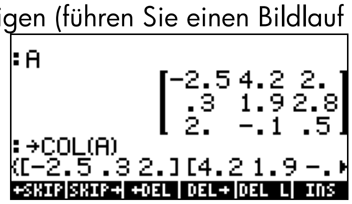




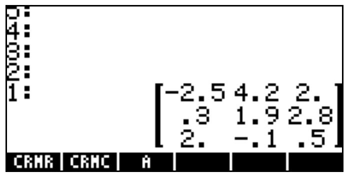
Die Anwendung dieser Funktionen wird nachfolgend dargestellt.

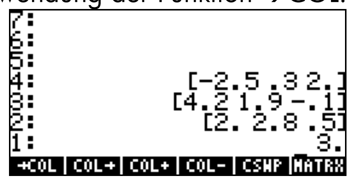
Funktion →COL

Die Funktion →COL nimmt als Argument eine Matrix und zerlegt diese in Vektoren, entsprechend ihrer Spalten. Nachfolgend wird eine Anwendung der Funktion →COL im ALG-Modus gezeigt. Die verwendete Matrix wurde zu einem früheren Zeitpunkt bereits in der Variablen A gespeichert. Die Matrix ist in der linken Abbildung zu sehen: Die rechte Abbildung zeigt die in Spalten zerlegte Matrix. Verwenden Sie den Zeileneditor, um das gesamte Ergebnis anzuzeigen (führen Sie einen Bildlauf mit der Taste ▽ durch).



Im RPN-Modus müssen Sie die Matrix zuerst in den Stack laden, und dann erst die Funktion →COL, d. h. →COL starten. Nachfolgende Abbildungen zeigen den RPN-Stack vor und nach Anwendung der Funktion →COL.



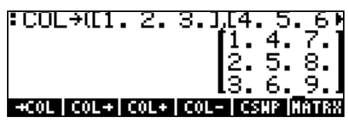


In diesem Ergebnis befindet sich die erste Spalte nach der Zerlegung in der obersten Stack-Ebene, während in Stack-Ebene 1 die Anzahl der Spalten der ursprünglichen Matrix zu finden ist. Die Matrix bleibt bei der Zerlegung nicht erhalten, d. h. sie ist im Stack nicht mehr verfügbar.

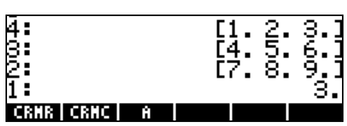
Funktion COL→

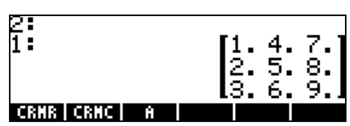
Die Funktion COL→ hat die entgegensetzte Wirkung der Funktion →COL, d. h. wenn Sie n Vektoren der gleichen Länge haben und die Zahl n, so bildet die Funktion COL→ eine Matrix, indem sie die eingegebenen Vektoren als Spalten der Matrix darstellt. Hier ein Beispiel im ALG-Modus. Der verwendete Befehl war:

```
COL→([1,2,3],[4,5,6],[7,8,9],3)
```



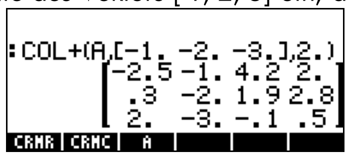
Geben Sie im RPN-Modus die n Vektoren in die Stack Ebenen n+1, n, n-1,…,2 ein und anschließend die Zahl n in Stack Ebene 1. Bei dieser Eingabe wird die Funktion COL→ die Vektoren als Spalten in die Matrix eingeben. Die nachfolgende Abbildung zeigt den RPN-Stack vor und nach Anwendung der Funktion COL→.



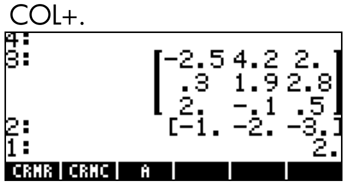


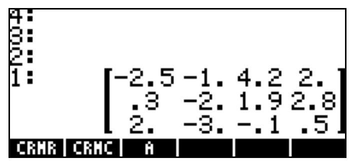
Funktion COL+

Die Funktion COL+ nimmt als Argumente eine Matrix, einen Vektor der gleichen Länge wie die Anzahl der Zeilen in der Matrix und eine Integer-Zahl n, die die Position einer Spalte darstellt. Die Funktion COL+ fügt den Vektor in Spalte n der Matrix ein. Als Beispiel setzen wir im ALG-Modus die zweite Spalte in Matrix A mithilfe des Vektors [-1,-2,-3] ein, d. h.



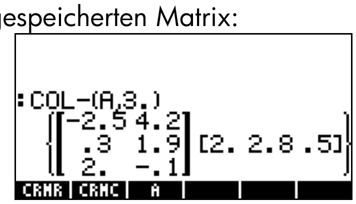
Im RPN-Modus geben wir zuerst die Matrix ein, dann den Vektor und die Nummer der Spalte, bevor wir die Funktion COL+ anwenden. Die nachfolgende Abbildung zeigt den RPN-Stack vor und nach Anwendung der Funktion COL+.



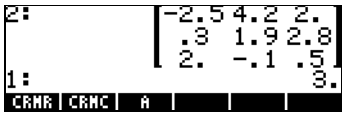


Funktion COL-

Die Funktion COL- verwendet als Argument eine Matrix und eine Integer-Zahl, welche die Position einer Spalte in der Matrix darstellt. Die Funktion gibt die ursprüngliche Matrix mit einer Spalte weniger, wieder, genauso wird auch die extrahierte Spalte als Vektor dargestellt. Hier ein Beispiel im ALG-Modus unter Verwendung der in A gespeicherten Matrix:

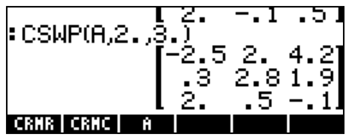


Im RPN-Modus laden Sie die Matrix erst in den Stack, dann geben Sie die Zahl, die eine Spalte der Matrix darstellt, vor Anwendung der Funktion COL- ein. Die nachfolgende Abbildung zeigt den RPN-Stack vor und nach Anwendung der Funktion COL-.

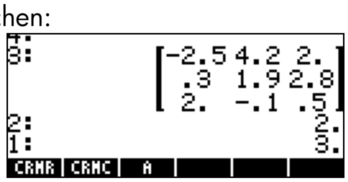


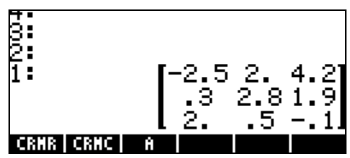
Funktion CSWP

Die Funktion CSWP (Column SWaP – Austauschen von Spalten) verwendet als Argumente zwei Indizes, beispielsweise i und j, (welche zwei unterschiedliche Spalten in der Matrix darstellen) und eine Matrix und erstellt daraus eine neue Matrix mit den Spalten i und j vertauscht. Das nachfolgende Beispiel, im ALG-Modus, zeigt die Anwendung dieser Funktion. Als Beispiel nehmen wir die in der Variablen A gespeicherte Matrix. Zuerst wird diese Matrix gelistet.



Im RPN-Modus können Sie mit der Funktion CSWP die Spalten einer Matrix in Stack-Ebene 3, deren Indizes in Stack-Ebene 1 und 2 aufgelistet sind, vertauschen. Als Beispiel, sehen Sie nachfolgend den RPN-Stack vor und nach Anwendung der Funktion CSWP auf die Matrix A, um die Spalten 2 und 3 zu vertauschen:

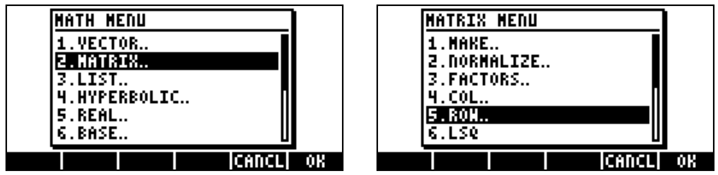




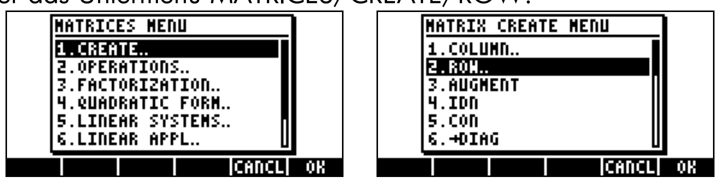
Wie Sie sehen können, wurden die Spalten, welche sich in Position 2 und 3 befunden haben, ausgetauscht. Der Austausch von Spalten und Zeilen (siehe unten) wird häufig bei der Lösung von linearen Gleichungen mit Matrizen verwendet. Eine genauere Beschreibung dieser Operationen erfolgt in einem späteren Kapitel.

Manipulation der Zeilen von Matrizen

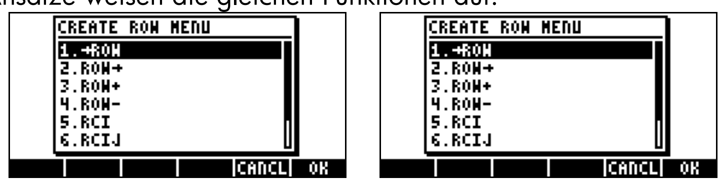
Der Taschenrechner enthält ein Menü mit Funktionen zur Manipulation der Zeilen von Matrizen. Dieses Menü kann über MTH/MATRIX/ROW.. (← MTH), wie in nachfolgender Abbildung, in Reihenfolge, gezeigt aufgerufen werden, während Systemflag 117 auf CHOOSE boxes gesetzt wurde:



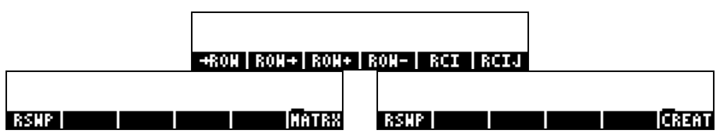
oder über das Untermenü MATRICES/CREATE/ROW:



Beide Ansätze weisen die gleichen Funktionen auf:



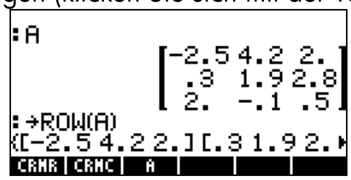
Ist das Systemflag 117 auf SOFT-Menüs gesetzt, kann das Menü ROW entweder über ← MTH MATRX MAKE ROW oder über ← MATRICES CREAT ROW aufgerufen werden. Beide Ansätze weisen dieselben Funktionen auf:

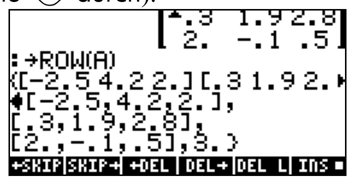


Die Anwendung dieser Funktionen wird nachfolgend dargestellt.

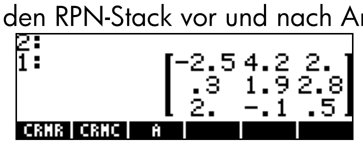
Funktion →ROW

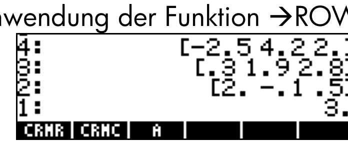
Die Funktion →ROW nimmt als Argument eine Matrix und zerlegt diese in Vektoren, entsprechend ihrer Zeilen. Nachfolgend wird eine Anwendung der Funktion →ROW im ALG-Modus gezeigt. Die verwendete Matrix wurde zu einem früheren Zeitpunkt bereits in der Variablen A gespeichert. Die Matrix ist in der linken Abbildung zu sehen: Die rechte Abbildung zeigt die in Zeilen zerlegte Matrix. Verwenden Sie den Zeileneditor, um das gesamte Ergebnis anzuzeigen (klicken Sie sich mit der Taste ▽ durch).





Im RPN-Modus müssen Sie die Matrix zuerst in den Stack laden, und dann erst die Funktion →ROW, d. h. →ROW starten. Die folgenden Abbildungen zeigen den RPN-Stack vor und nach Anwendung der Funktion →ROW.





In diesem Ergebnis ist die erste Zeile, nach der Zerlegung, in der höchsten Stack-Ebene, während in Stack-Ebene 1 die Anzahl der Zeilen der ursprünglichen Matrix zu finden sind. Die Matrix bleibt bei der Zerlegung nicht erhalten, d. h. sie ist im Stack nicht mehr verfügbar.

Funktion ROW→

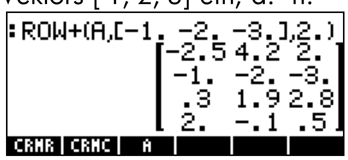
Die Funktion ROW→ hat die entgegensetzte Wirkung der Funktion →ROW, d. h. wenn Sie n Vektoren der gleiche Länge haben und die Zahl n, bildet die Funktion ROW→ eine Matrix, indem sie die eingegebenen Vektoren als Zeilen der Matrix darstellt. Hier ein Beispiel im ALG-Modus. Der verwendete Befehl war:

ROW→([1,2,3],[4,5,6],[7,8,9],3)

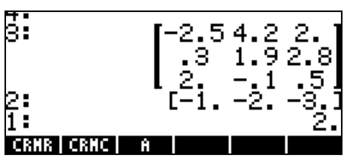
Geben Sie im RPN-Modus die n Vektoren in die Stack-Ebenen n+1, n, n-1,…,2 ein und anschließend die Zahl n in Stack-Ebene 1. So eingegeben, wird die Funktion ROW→ die Vektoren als Zeilen in die Matrix eingeben. Die nachfolgende Abbildung zeigt den RPN-Stack vor und nach Anwendung der Funktion ROW→.

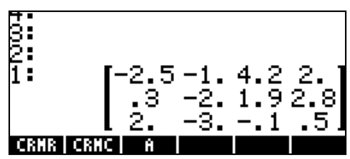
Funktion ROW+

Die Funktion ROW+ nimmt als Argument eine Matrix, einen Vektor der gleichen Länge wie die Anzahl der Zeilen in der Matrix und eine Integer-Zahl n, die die Position einer Zeile darstellt. Die Funktion ROW+ fügt den Vektor in Zeile n der Matrix ein. Als Beispiel setzen wir im ALG-Modus die zweite Zeile in Matrix A mithilfe des Vektors [-1,-2,-3] ein, d. h.



Im RPN-Modus geben wir zuerst die Matrix ein, dann den Vektor und die Nummer der Zeile, bevor wir die Funktion ROW+ anwenden. Die nachfolgende Abbildung zeigt den RPN-Stack vor und nach Anwendung der Funktion ROW+.

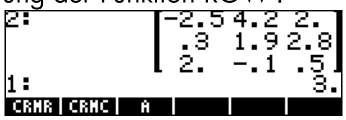




Funktion ROW-

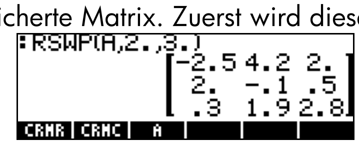
Die Funktion ROW- verwendet als Argument eine Matrix und eine Integer-Zahl, welche die Position der Zeile in der Matrix darstellt. Die Funktion gibt die ursprüngliche Matrix mit einer Zeile weniger wieder, ebenso wird auch die extrahierte Zeile als Vektor dargestellt. Hier ein Beispiel im ALG-Modus unter Verwendung der in A gespeicherten Matrix:

Im RPN-Modus laden Sie die Matrix erst in den Stack, dann geben Sie die Zahl, die eine Zeile der Matrix darstellt, vor Anwendung der Funktion ROW- ein. Die nachfolgende Abbildung zeigt den RPN-Stack vor und nach Anwendung der Funktion ROW-.

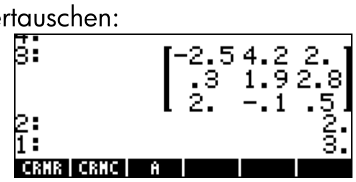


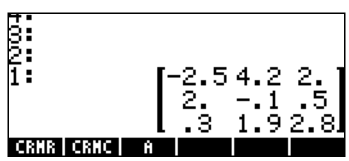
Funktion RSWP

Die Funktion RSWP (Row SWaP – Austauschen von Zeilen) verwendet als Argumente zwei Indizes, beispielsweise i und j, (welche zwei unterschiedliche Zeilen in der Matrix darstellen) und eine Matrix und erstellt daraus eine neue Matrix mit den Zeilen i und j vertauscht. Das nachfolgende Beispiel, im ALG-Modus, zeigt die Anwendung dieser Funktion. Als Beispiel nehmen wir die in der Variablen A gespeicherte Matrix. Zuerst wird diese Matrix aufgelistet.



Im RPN-Modus können Sie mit der Funktion CSWP die Zeilen einer Matrix angezeigt in Stack-Ebene 3, deren Indizes in Stack-Ebene 1 und 2 aufgelistet sind, vertauschen. Als Beispiel, sehen Sie nachfolgend den RPN-Stack vor und nach Anwendung der Funktion CSWP auf die Matrix A, um die Zeilen 2 und 3 zu vertauschen:

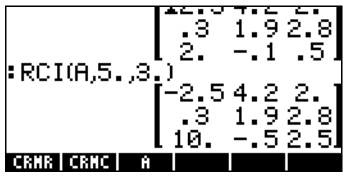




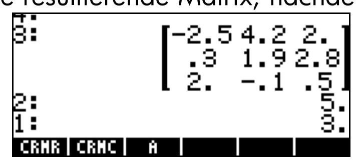
Wie Sie sehen können, wurden die Zeilen, welche sich in Position 2 und 3 befunden haben, ausgetauscht.

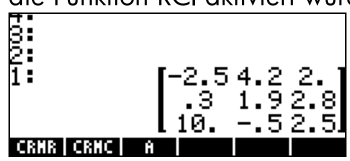
Funktion RCI

Die Funktion RCI steht für Multiplikation der Zeile I mit einem konstanten Wert und Ersetzen der so entstandenen Zeile an der gleichen Stelle wie die vorhergehende. Das nachfolgende Beispiel, im ALG-Modus, nimmt die in der Variablen A gespeicherte Matrix und multipliziert den konstanten Wert 5 mit der Zeile Nr. 3, und ersetzt diese Zeile mit dem Produkt der Multiplikation.



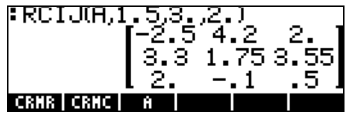
Die gleiche Übung wird in nachfolgender Abbildung im RPN-Modus angezeigt. Die linke Abbildung zeigt die Erstellung der Matrix, den Faktor und die Anzahl der Zeilen in Stack-Ebene 3, 2 und 1. Die rechte Abbildung zeigt die resultierende Matrix, nachdem die Funktion RCI aktiviert wurde.



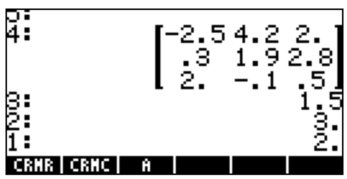


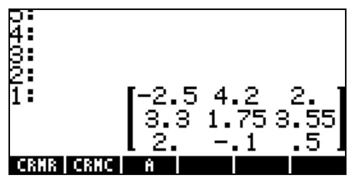
Funktion RCIJ

Die Funktion RCIJ steht für "nehme Zeile I und multipliziere diese mit der Konstanten C, setze dann die multiplizierte Reihe in Reihe J und ersetze die Reihe J mit der errechneten Summe". Diese Art von Operation wird häufig in der Gaußschen oder der Gauß-Jordan Elimination verwendet (weitere Einzelheiten zu dieser Prozedur finden Sie in einem späteren Kapitel). Die Argumente der Funktion sind: (1) die Matrix, (2) der konstante Wert, (3) die Zeile die mit der Konstanten in (2) multipliziert werden soll und (4) die Zeile, die mit dem Ergebnis, wie oben beschrieben, ersetzt werden soll. Nehmen wir z. B. die in A gespeicherte Matrix, multiplizieren wir die dritte Spalte mit 1,5 und addieren diese zu Spalte 2. Folgendes Beispiel wird im ALG-Modus ausgeführt:



Im RPN-Modus, geben Sie zuerst die Matrix, gefolgt von der Konstanten, ein dann die Zeile, die mit der Konstanten multipliziert werden soll und schließlich dann die Zeile, die ersetzt werden soll. Die nachfolgende Abbildung zeigt den RPN-Stack vor und nachdem die Funktion RCIJ, unter denselben Bedingungen, wie in dem Beispiel im ALG-Modus vorhin gezeigt, angewendet wird:





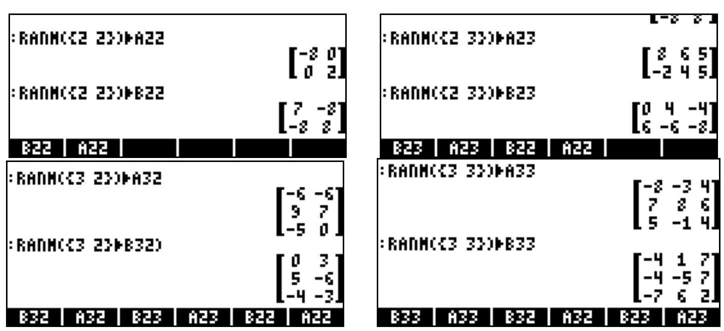
Kapitel 11
Matrix-Operationen und lineare Algebra

In Kapitel 10 führten wir das Konzept der Matrix ein und stellten mehrere Funktionen zum Eingeben, Erstellen und Bearbeiten von Matrizen vor. In diesem Kapitel präsentieren wir Beispiele für Matrix-Operationen und -Anwendungen in Bezug auf Probleme der linearen Algebra.

Operationen mit Matrizen

Matrizen können wie andere mathematische Objekte addiert und subtrahiert werden. Sie können mit einem Skalar oder auch miteinander multipliziert werden. Eine für Anwendungen der linearen Algebra wichtige Operation ist die Bildung der Inversen einer Matrix. Diese Operationen werden nun dargestellt.

Zur Veranschaulichung der Operationen erstellen wir mehrere Matrizen, die wir in Variablen speichern. Die generischen Namen der Matrizen lauten A*ij* und B*ij*, wobei *i* die Anzahl der Zeilen und *j* die Anzahl der Spalten der Matrizen darstellen. Die verwendeten Matrizen werden mithilfe der Funktion RANM (Zufallsmatrizen) erzeugt. Wenn Sie diese Übung mit dem Taschenrechner durchführen, erhalten Sie andere als die hier aufgeführten Matrizen, sofern Sie sie nicht genau wie unten dargestellt im Taschenrechner speichern. Unten sind die im ALG-Modus erzeugten Matrizen A22, B22, A23, B23, A32, B32, A33 und B33 dargestellt:

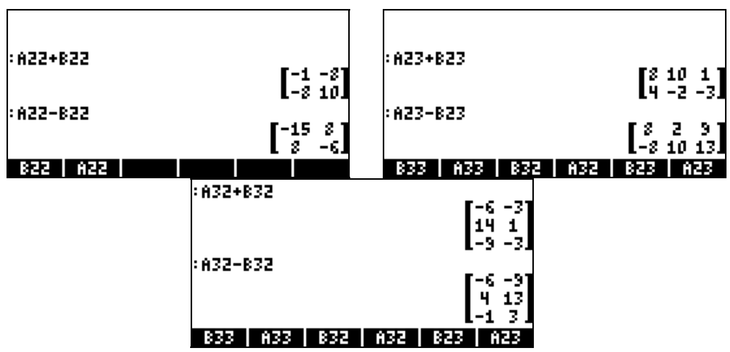


Im RPN-Modus lauten die Schritte wie folgt:

```
{2,2} ENTER RANM 'A22' STO▶   {2,2} ENTER RANM 'B22' STO▶
{2,3} ENTER RANM 'A23' STO▶   {2,3} ENTER RANM 'B23' STO▶
{3,2} ENTER RANM 'A32' STO▶   {3,2} ENTER RANM 'B32' STO▶
{3,3} ENTER RANM 'A33' STO▶   {3,3} ENTER RANM 'B33' STO▶
```

Addition und Subtraktion

Gegeben seien zwei Matrizen $\mathbf{A} = [a_{ij}]_{m\times n}$ und $\mathbf{B} = [b_{ij}]_{m\times n}$. Diese beiden Matrizen können nur addiert bzw. subtrahiert werden, wenn die Anzahl ihrer Zeilen und Spalten übereinstimmt. Die resultierende Matrix $\mathbf{C} = \mathbf{A} \pm \mathbf{B} = [c_{ij}]_{m\times n}$ besitzt die Elemente $c_{ij} = a_{ij} \pm b_{ij}$. Nachfolgend finden Sie einige Beispiele im ALG-Modus, unter Verwendung der oben gespeicherten Matrizen (z.B. A22 + B22)



Im RPN-Modus lauten die Schritte wie folgt:

```
A22 ENTER B22 ENTER +    A22 ENTER B22 ENTER -
A23 ENTER B23 ENTER +    A23 ENTER B23 ENTER -
A32 ENTER B32 ENTER +    A32 ENTER B32 ENTER -
```

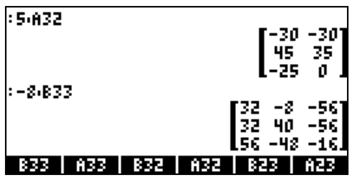
Wie hier veranschaulicht, ist die Umwandlung der Beispiele vom ALG-Modus in den RPN-Modus einfach. Die übrigen Beispiele für Matrix-Operationen werden ausschließlich im ALG-Modus durchgeführt.

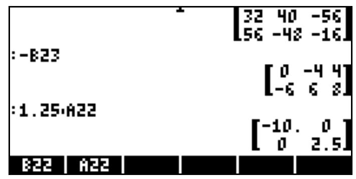
Multiplikation

Es sind mehrere Multiplikationsoperationen mit Matrizen möglich. Diese werden im Folgenden beschrieben.

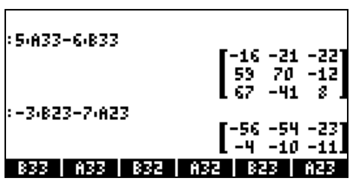
Multiplikation mit einem Skalar

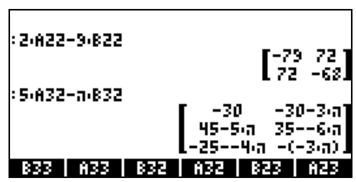
Durch Multiplikation der Matrix **A** = $[a_{ij}]_{m\times n}$ mit einem Skalar ergibt sich die Matrix **C** = k**A** = $[c_{ij}]_{m\times n}$ = $[ka_{ij}]_{m\times n}$. Eine negative Matrix wird durch die Operation -**A** =(-1)**A** = $[-a_{ij}]_{m\times n}$ definiert. Unten sind einige Beispiele für die Multiplikation einer Matrix mit einem Skalar dargestellt.



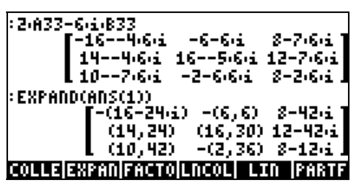


Durch die Kombination von Addition und Subtraktion mit der Skalarmultiplikation können wir lineare Kombinationen von Matrizen derselben Dimension bilden, z. B.



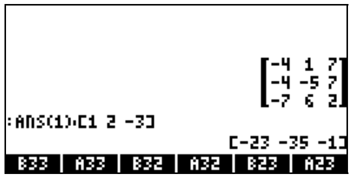


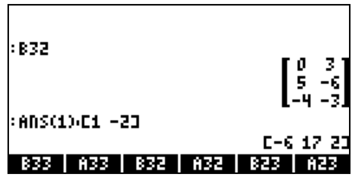
In einer Linearkombination von Matrizen können wir eine Matrix mit einer imaginären Zahl multiplizieren, um eine Matrix komplexer Zahlen zu erhalten, z. B.



Matrix-Vektor-Multiplikation

Die Matrix-Vektor-Multiplikation ist nur dann möglich, wenn die Anzahl der Matrix-Spalten mit der Länge des Vektors übereinstimmt. Diese Operation erfolgt nach den im nächsten Abschnitt dargestellten Regeln der Matrix-Multiplikation. Es folgen mehrere Beispiele für die Matrix-Vektor-Multiplikation:





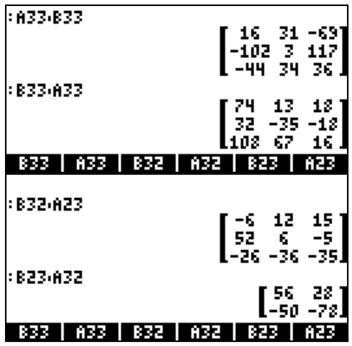
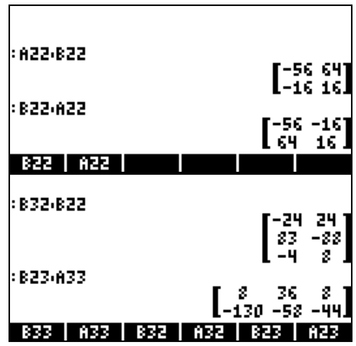
Die Vektor-Matrix-Multiplikation ist hingegen nicht definiert. Diese Multiplikation kann jedoch als spezieller Fall der im Folgenden definierten Matrix-Multiplikation ausgeführt werden.

Matrix-Multiplikation

Die Matrix-Multiplikation ist durch $\mathbf{C}_{m\times n} = \mathbf{A}_{m\times p}\cdot\mathbf{B}_{p\times n}$, definiert, wobei $\mathbf{A} = [a_{ij}]_{m\times p}$, $\mathbf{B} = [b_{ij}]_{p\times n}$ und $\mathbf{C} = [c_{ij}]_{m\times n}$. Beachten Sie, dass die Matrix-Multiplikation nur dann möglich ist, wenn die Anzahl der Spalten im ersten Operanden gleich der Anzahl der Zeilen im zweiten Operanden ist. Die allgemeine Größe des Produkts, c_{ij}, ist definiert als

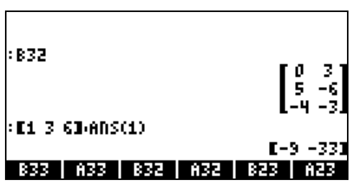
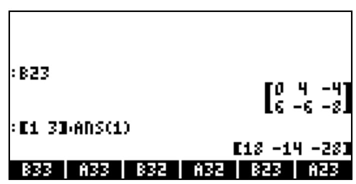
$$c_{ij} = \sum_{k=1}^{p} a_{ik} \cdot b_{kj}\,,\ f \quad i = 1,2,\ldots,m;\ j = 1,2,\ldots,n.$$

Dies bedeutet, dass das Element in der i-ten Zeile und j-ten Spalte des Produkts **C** dadurch gebildet wird, dass die einzelnen Größen der i-ten Zeile von **A** mit den einzelnen Größen der j-ten Spalte von **B** multipliziert und die Produkte addiert werden. Die Matrix-Multiplikation ist nicht kommutativ, d. h., allgemein gilt: $\mathbf{A}\cdot\mathbf{B} \neq \mathbf{B}\cdot\mathbf{A}$. Darüber hinaus kann sogar eine der Multiplikationen nicht vorhanden sein. In den folgenden Bildschirmabbildungen werden die Ergebnisse der Multiplikationen der zuvor gespeicherten Matrizen dargestellt:

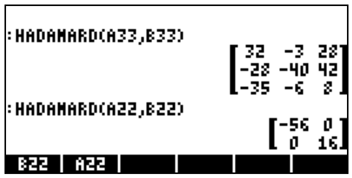
Die im vorherigen Abschnitt vorgestellte Matrix-Vektor-Multiplikation kann als Produkt einer m×n-Matrix mit einer n×1-Matrix (d. h. einem Spaltenvektor) gedacht werden, der eine m×1-Matrix (also einen anderen Vektor) ergibt. Überprüfen Sie die im vorherigen Abschnitt dargestellten Beispiele, um diese Aussage zu verifizieren. Für den Zweck der Matrixmultiplikation sind daher die in Kapitel 9 definierten Vektoren im Grunde genommen Spaltenvektoren.

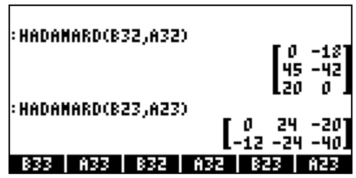
Das Produkt eines Vektors mit einer Matrix kann gebildet werden, wenn der Vektor ein Zeilenvektor ist, d. h. eine 1×m-Matrix, die bei der Multiplikation mit einer m×n-Matrix eine 1×n-Matrix ergibt. Damit der Taschenrechner einen Zeilenvektor erkennen kann, müssen Sie diesen in Klammern eingeben, z. B.

Multiplikation der einzelnen Größen nacheinander

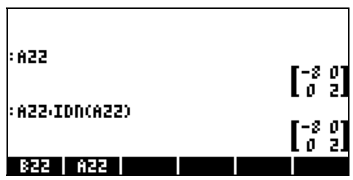
Die Multiplikation der einzelnen Größen zweier Matrizen derselben Dimension nacheinander ist durch Verwendung der Funktion HADAMARD möglich. Das Ergebnis ist natürlich eine weitere Matrix derselben Dimension. Diese Funktion ist über den Funktionskatalog (→ _CAT_) oder über das Untermenü MATRICES/OPERATIONS (← _MATRICES_) verfügbar. Im Folgenden werden Anwendungen der Funktion HADAMARD vorgestellt:





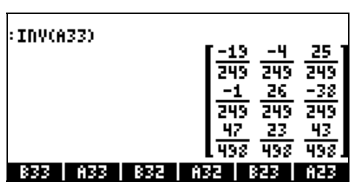
Die Einheitsmatrix

In Kapitel 9 wird die Einheitsmatrix als Matrix $\mathbf{I} = [\delta_{ij}]_{n\times n}$ vorgestellt, wobei δ_{ij} die Kronecker-Deltafunktion darstellt. Einheitsmatrizen können durch Verwendung der in Kapitel 9 beschriebenen Funktion IDN erzeugt werden. Für die Einheitsmatrix gilt: $\mathbf{A}\cdot\mathbf{I} = \mathbf{I}\cdot\mathbf{A} = \mathbf{A}$. Zur Überprüfung dieser Eigenschaft stellen wir die folgenden Beispiele dar und verwenden hierfür die bereits gespeicherten Matrizen:

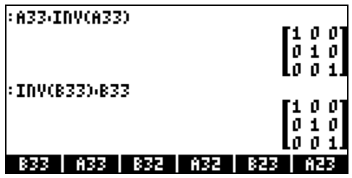


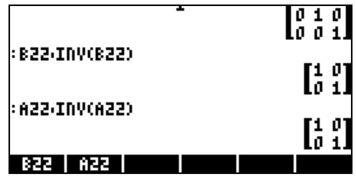
Die inverse Matrix

Die Inverse einer quadratischen Matrix $\mathbf{A}$ ist die Matrix $\mathbf{A}^{-1}$, sodass $\mathbf{A}\cdot\mathbf{A}^{-1} = \mathbf{A}^{-1}\cdot\mathbf{A} = \mathbf{I}$, wobei $\mathbf{I}$ die Einheitsmatrix mit derselben Dimension wie $\mathbf{A}$ ist. Mit dem Taschenrechner erhalten Sie die Inverse einer Matrix durch die Umkehrfunktion INV (d. h mit der 1/x-Taste). Unten sind Beispiele für die Inverse einiger zuvor gespeicherter Matrizen dargestellt:



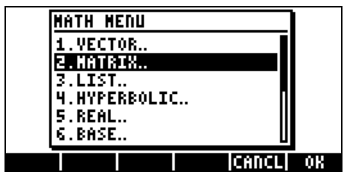
Zur Überprüfung der Eigenschaften der inversen Matrix stellen wir die folgenden Multiplikationen dar:





Beschreiben einer Matrix (Das Matrixmenü NORM)

Das Matrixmenü NORM (NORMALIZE) wird mit der Tastenkombination MTH aufgerufen (Systemflag 117 ist auf CHOOSE boxes gesetzt):



Das Menü enthält die folgenden Funktionen:

Diese Funktionen werden im Folgenden beschrieben. Da viele dieser Funktionen Konzepte der Matrixtheorie, z. B. Singulärwerte, Rang usw., verwenden, enthalten die Beschreibungen der Funktionen kurze Darstellungen dieser Konzepte.

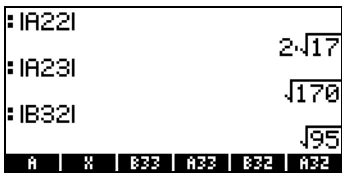
Funktion ABS

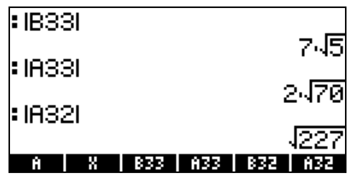
Mit der Funktion ABS wird die Frobenius-Norm einer Matrix berechnet. Für eine Matrix **A** = $[a_{ij}]_{m\times n}$ wird die Frobenius-Norm einer Matrix definiert als

$$\|A\|_F = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{m} a_{ij}^2}$$

Wenn es sich bei der betreffenden Matrix um einen Zeilen- oder Spaltenvektor handelt, ist die Frobenius-Norm $||\mathbf{A}||_F$ einfach der Betrag des Vektors. Die Funktion ABS kann direkt über die Tastenkombination ⬅ ABS aufgerufen werden.

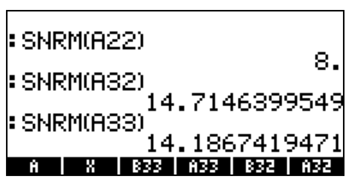
Führen Sie im ALG-Modus die folgenden Übungen durch (mit den zuvor für Matrix-Operationen gespeicherten Matrizen):





Funktion SNRM

Mit der Funktion SNRM wird die Spektralnorm einer Matrix berechnet, die als der größte Singulärwert der Matrix definiert ist und auch als euklidische Norm der Matrix bezeichnet wird. Beispiel:



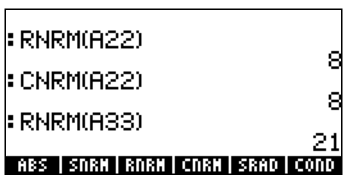
Singulärwertzerlegung

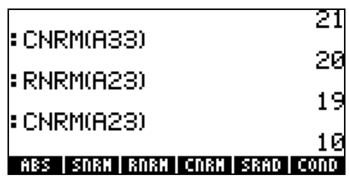
Zum Verständnis der Funktion SNRM müssen wir das Konzept der Matrixzerlegung erläutern. Die Matrixzerlegung umfasst im Wesentlichen die Bestimmung von mindestens zwei Matrizen, die die ursprüngliche Matrix ergeben, wenn sie in einer bestimmten Reihenfolge (eventuell mit Matrixinversion oder -transposition) multipliziert werden. Bei der Singulärwertzerlegung (SVD) wird eine rechteckige Matrix $\mathbf{A}_{m\times n}$ als $\mathbf{A}_{m\times n} = \mathbf{U}_{m\times m} \cdot \mathbf{S}_{m\times n} \cdot \mathbf{V}^T_{n\times n}$ beschrieben, wobei es sich bei **U** und V um Orthogonalmatrizen und bei **S** um eine Diagonalmatrix handelt. Die diagonalen Elemente von **S** werden als Singulärwerte von A bezeichnet und sind in der Regel so angeordnet, dass für i = 1, 2, …, n-1 gilt, dass $s_i \geq s_{i+1}$. Die Spalten [$\mathbf{u}_j$] von **U** und [$\mathbf{v}_j$] von **V** sind die entsprechenden Singulärvektoren. (Für Orthogonalmatrizen gilt: $\mathbf{U} \cdot \mathbf{U}^T = \mathbf{I}$. Eine Diagonalmatrix besitzt nur entlang ihrer Hauptdiagonalen Elemente ungleich Null.)

Der Rang einer Matrix kann anhand ihrer SVD bestimmt werden, indem die Anzahl der nichtsingulären Werte gezählt wird. Der nächste Abschnitt enthält Beispiele für die Singulärwertzerlegung.

Funktionen RNRM und CNRM

Die Funktion RNRM gibt die Zeilennorm einer Matrix und die Funktion CNRM die Spaltennorm einer Matrix zurück. Beispiele:

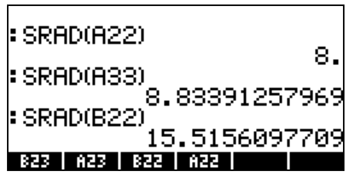




Zeilennnorm und Spaltennorm einer Matrix
Die Zeilennorm einer Matrix wird berechnet, indem die Summen der absoluten Werte aller Elemente in jeder Zeile gebildet werden und anschließend der Höchstwert dieser Summen ausgewählt wird. Die Spaltennorm einer Matrix wird berechnet, indem die Summen der absoluten Werte aller Elemente in jeder Spalte gebildet werden und anschließend der Höchstwert dieser Summen ausgewählt wird.

Funktion SRAD

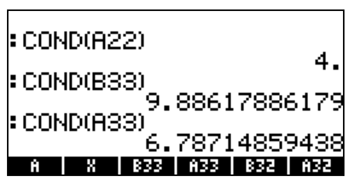
Mit der Funktion SRAD wird der Spektralradius einer Matrix bestimmt. Dieser ist als der größte absolute Wert der Eigenwerte einer Matrix definiert. Beispiel:



Definition der Eigenwerte und Eigenvektoren einer Matrix
Die Eigenwerte einer quadratischen Matrix sind das Ergebnis der Matrixgleichung $\mathbf{A}\cdot\mathbf{x} = \lambda\cdot\mathbf{x}$. Die der Gleichung entsprechenden Werte von λ werden als Eigenwerte der Matrix **A** bezeichnet. Die für jeden Wert von λ aus der Gleichung resultierenden Werte von x werden als Eigenvektoren der Matrix bezeichnet. Weitere Informationen über das Berechnen von Eigenwerten und Eigenvektoren erhalten Sie im nächsten Kapitel.

Funktion COND

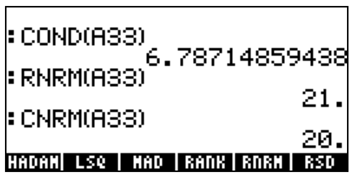
Mit der Funktion COND wird die Konditionszahl einer Matrix bestimmt. Beispiele:

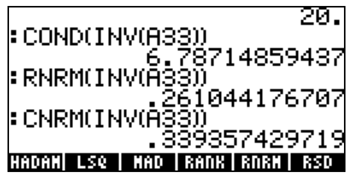


Konditionszahl einer Matrix
Die Konditionszahl einer quadratischen nichtsingulären Matrix ist als das Produkt der Matrixnorm und der Norm ihrer Inversen definiert, d. h. $\text{cond}(\mathbf{A}) = ||\mathbf{A}||\times||\mathbf{A}^{-1}||$. Wir wählen als Matrixnorm $||\mathbf{A}||$ den Höchstwert ihrer Zeilennorm (RNRM) und ihrer Spaltennorm (CNRM), während als Norm ihrer Inversen $||\mathbf{A}^{-1}||$ der Mindestwert ihrer Zeilennorm und Spaltennorm gewählt wird. Somit gilt $||\mathbf{A}|| = \max(\text{RNRM}(\mathbf{A}),\text{CNRM}(\mathbf{A}))$ und $||\mathbf{A}^{-1}|| = \min(\text{RNRM}(\mathbf{A}^{-1}), \text{CNRM}(\mathbf{A}^{-1}))$.

Die Konditionszahl einer singulären Matrix ist Unendlich. Die Konditionszahl einer nichtsingulären Matrix bestimmt, wie weit die Matrix von der Singularität entfernt ist. Je größer die Konditionszahl, desto näher befindet sich die Matrix an der Singularität. (Eine singuläre Matrix ist eine Matrix, für die keine inverse Matrix vorhanden ist.)

Führen Sie für Matrix A33 folgende Übung zur Matrixkonditionszahl durch. Die Konditionszahl lautet COND(A33). Zeilennorm und Spaltennorm für A33 werden auf der linken Seite angezeigt. Die entsprechenden Zahlen für die inverse Matrix INV(A33) werden auf der linken Seite angezeigt:

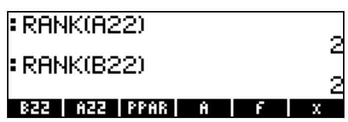




Da RNRM(A33) > CNRM(A33), ist ||A33|| = RNRM(A33) = 21. Da außerdem CNRM(INV(A33)) < RNRM(INV(A33)), ist ||INV(A33)|| = CNRM(INV(A33)) = 0.261044... Die Konditionszahl wird somit als CNRM(A33)*CNRM(INV(A33)) = COND(A33) = 6.7871485… berechnet.

Funktion RANK

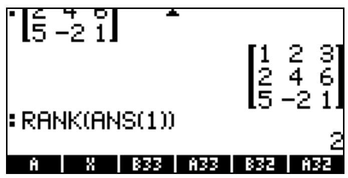
Mit der Funktion RANK wird der Rang einer quadratischen Matrix bestimmt. Testen Sie folgende Beispiele:



Rang einer Matrix

Der Rang einer quadratischen Matrix ist die maximale Anzahl linear unabhängiger Zeilen oder Spalten der Matrix. Angenommen Sie erstellen eine quadratische Matrix $\mathbf{A}_{n\times n}$ als $\mathbf{A} = [\mathbf{c}_1\ \mathbf{c}_2\ \ldots\ \mathbf{c}_n]$, wobei $\mathbf{c}_i$ (i = 1, 2, …, n) Vektoren sind, die die Spalten von Matrix **A** darstellen. Wenn dann eine dieser Spalten, z. B. $\mathbf{c}_k$, als $\mathbf{c}_k = \sum_{j\neq k,\, j\in\{1,2,\ldots,n\}} d_j \cdot \mathbf{c}_j$, beschrieben werden kann, wobei die Werte d_j konstant sind, ist $\mathbf{c}_k$ von den in der Summe enthaltenen Spalten linear unabhängig. (Beachten Sie, dass die Werte von j jeden Wert in der Menge {1, 2, …, n} in jeder beliebigen Kombination enthalten, solange j≠k.) Wenn der obige Ausdruck für keinen der Spaltenvektoren gebildet werden kann, sind alle Spalten linear unabhängig. Eine vergleichbare Definition der linearen Unabhängigkeit von Zeilen kann entwickelt werden, indem die Matrix als eine Spalte von Zeilenvektoren dargestellt wird. Wenn daher rank(**A**) = n, besitzt die Matrix eine Inverse und ist eine nichtsinguläre Matrix. Wenn hingegen rank(**A**) < n, ist die Matrix singulär und keine Inverse vorhanden.

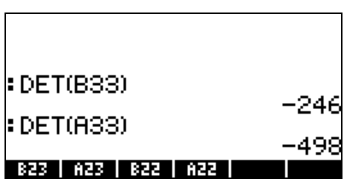
Bestimmen Sie beispielsweise den Rang der folgenden Matrix:

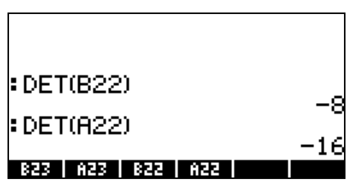


Der Rang ist 2. Der Grund hierfür ist, dass die zweite Zeile [2,4,6] gleich dem Produkt der ersten Zeile [1,2,3] mit 2 ist. Somit ist Zeile zwei von Zeile 1 linear abhängig und die maximale Anzahl linear unabhängiger Zeilen ist 2. Sie können überprüfen, ob die maximale Anzahl linear unabhängiger Zeilen 3 ist. Der Rang, die maximale Anzahl linear unabhängiger Zeilen oder Zeilen, ist in diesem Fall 2.

Funktion DET

Mit der Funktion DET wird die Determinante einer quadratischen Matrix berechnet. Beispiel:



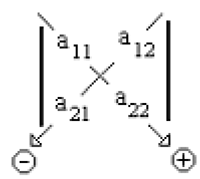


Determinante einer Matrix

Die Determinanten einer 2x2- und einer 3x3-Matrix werden durch dieselbe Anordnung der Elemente der Matrizen dargestellt, jedoch zwischen vertikalen Linien, also

$$\begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix}, \qquad \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix}$$

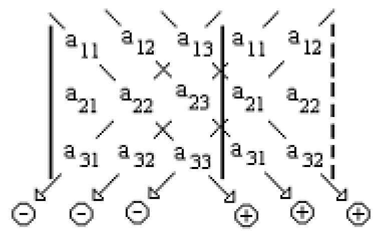
Eine 2×2-Determinante wird berechnet, indem die Elemente auf ihrer Diagonalen multipliziert und diese Produkte mit positivem bzw. negativem Vorzeichen addiert werden, wie im Diagramm unten dargestellt.



Die 2×2-Determinante lautet daher

$$\begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} = a_{11} \cdot a_{22} - a_{12} \cdot a_{21}$$

Eine 3×3-Determinante wird berechnet, indem die Determinante erweitert wird. Dies geschieht, indem die ersten beiden Spalten der Determinante kopiert und rechts von Spalte 3 eingefügt werden, wie im Diagramm unten dargestellt. Im Diagramm sind auch Elemente dargestellt, die auf die gleiche Weise wie zuvor bei einer 2×2-Determinante multipliziert werden, wobei das Produkt das entsprechende Vorzeichen erhält. Nach der Multiplikation werden die Ergebnisse addiert, um die Determinante zu erhalten.



Determinanten für quadratische Matrizen höherer Ordnung können mithilfe von Determinanten niedrigerer Ordnung, die als Kofaktor bezeichnet werden, berechnet werden. Hierbei wird die Determinante einer n×n-Matrix (auch als n×n-Determinante bezeichnet) zu einer Summe der Kofaktoren „erweitert", bei denen es sich um (n-1)×(n-1) Determinanten handelt, multipliziert mit den Elementen einer einzelnen Zeile oder Spalte, wobei die Vorzeichen abwechselnd positiv und negativ sind. Diese „Erweiterung" wird mit den Kofaktoren der Ordnung (n-2)×(n-2) auf die nächste (niedrigere) Ebene übertragen usw., bis nur noch eine umfangreiche Summe von 2×2-Determinanten vorhanden ist. Die 2×2-Determinanten werden dann mit der oben dargestellten Methode berechnet.

Die Berechnung einer Determinante durch Entwicklung der Kofaktoren ist insofern sehr ineffizient, als sie Operationen umfasst, deren Anzahl mit der Größe der Determinante sehr schnell zunimmt. Eine effizientere und bei numerischen Anwendungen bevorzugte Methode besteht darin, das Ergebnis einer Gauß-Elimination zu verwenden. Die Gauß-Elimination wird zum Lösen linearer Gleichungssysteme verwendet. Diese Methode wird weiter unten in diesem Kapitel dargestellt.

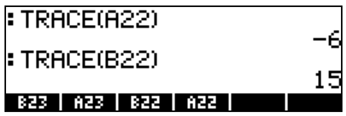
Die Determinante einer Matrix **A** wird als det(**A**) dargestellt. Die Determinante einer singulären Matrix ist gleich Null.

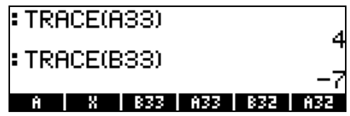
Funktion TRACE

Mit der Funktion TRACE wird die Spur einer quadratischen Matrix berechnet, die als die Summe der Elemente ihrer Hauptdiagonalen definiert ist oder als

$$tr(\mathbf{A}) = \sum_{i=1}^{n} a_{ii} \ .$$

Beispiele:



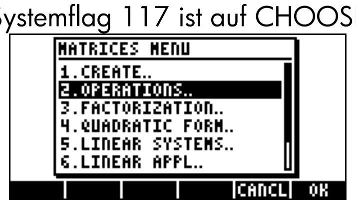


Funktion TRAN

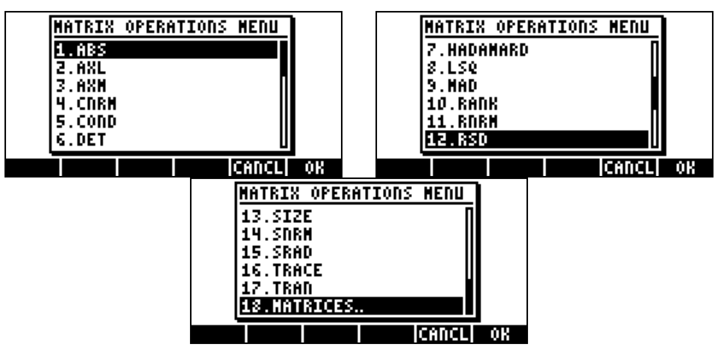
Die Funktion TRAN gibt die Transponierte einer reellen Matrix oder die konjugierte Transponierte einer komplexen Matrix zurück. TRAN ist mit TRN äquivalent. Die Funktion TRN wurde in Kapitel 10 erläutert.

Weitere Matrix-Operationen (Das Matrix-Menü OPER)

Das Matrixmenü OPER (OPERATIONS) wird mit der Tastenkombination ← MATRICES aufgerufen (Systemflag 117 ist auf CHOOSE boxes gesetzt):



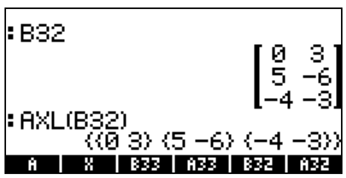
Das Menü OPERATIONS enthält die folgenden Funktionen:

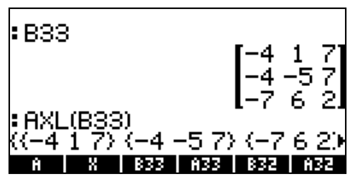


Die Funktionen ABS, CNRM, COND, DET, RANK, RNRM, SNRM, TRACE und TRAN sind auch im Menü MTH/MATRIX/NORM (das Thema des vorherigen Abschnitts) verfügbar. Die Funktion SIZE wurde in Kapitel 10 dargestellt. Die Funktion HADAMARD wurde bereits im Zusammenhang mit der Matrix-Multiplikation vorgestellt. Die Funktionen LSQ, MAD und RSD werden bei der Lösung linearer Gleichungssysteme verwendet und in einem späteren Abschnitt dieses Kapitels dargestellt. In diesem Abschnitt werden nur die Funktionen AXL und AXM erläutert.

Funktion AXL

Mit der Funktion AXL wird ein Feld (eine Matrix) in eine Liste umgewandelt und umgekehrt. Beispiele:

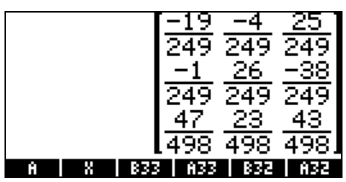


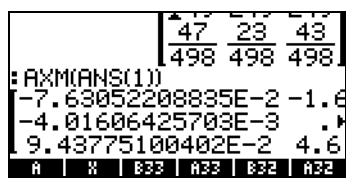


Hinweis: Die letzte Operation ist mit der Operation des in Kapitel 10 dargestellten Programms CRMR vergleichbar.

Funktion AXM

Mit der Funktion AXM wird ein Feld mit ganzen Zahlen oder Brüchen in die entsprechende Dezimal- oder Näherungsform umgewandelt. Beispiel:





Funktion LCXM

Mit der Funktion LCXM können Matrizen erzeugt werden, für die gilt, dass das Element aij eine Funktion von i und j ist. Die Eingangswerte dieser

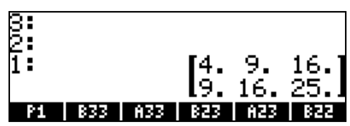
Funktion sind zwei Ganzzahlen n und m, die die Anzahl der Zeilen und Spalten der zu erzeugenden Matrix darstellen, und ein Programm mit den Eingangswerten i und j. Die Zahlen n und m sowie das Programm belegen jeweils Ebene 3, 2 und 1 des Stacks. Die Funktion LCXM kann über den Befehlskatalog ↱ _CAT aufgerufen werden.

Um beispielsweise eine 2´3-Matrix zu erzeugen, deren Elemente durch $a_{ij} = (i+j)^2$ gegeben sind, speichern Sie zunächst im RPN-Modus das folgende Programm in der Variablen P1. Bevor Sie STO▸ drücken, wird der RPN-Stack wie folgt angezeigt:

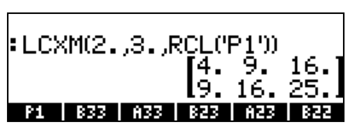
Die Ausführung der Funktion LCXM erfordert in diesem Fall, dass Sie Folgendes eingeben:

2 ENTER 3 ENTER ↱ LCXM ENTER

In der folgenden Abbildung ist der RPN-Stack vor und nach Anwendung der Funktion LCXM dargestellt:



Im ALG-Modus erhalten Sie dieses Beispiel durch folgende Eingabe:



Das Programm P1 muss jedoch im RPN-Modus erstellt und gespeichert worden sein.

Lösung linearer Gleichungssysteme

Ein System von *n* linearen Gleichungen mit *m* Variablen kann folgendermaßen beschrieben werden:

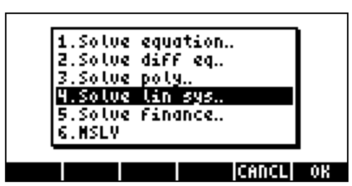
$$
\begin{array}{l}
a_{11}\cdot x_1 + a_{12}\cdot x_2 + a_{13}\cdot x_3 + \ldots + a_{1,m-1}\cdot x_{m-1} + a_{1,m}\cdot x_m = b_1, \\
a_{21}\cdot x_1 + a_{22}\cdot x_2 + a_{23}\cdot x_3 + \ldots + a_{2,m-1}\cdot x_{m-1} + a_{2,m}\cdot x_m = b_2, \\
a_{31}\cdot x_1 + a_{32}\cdot x_2 + a_{33}\cdot x_3 + \ldots + a_{3,m-1}\cdot x_{m-1} + a_{3,m}\cdot x_m = b_3, \\
\quad . \qquad . \qquad . \qquad \ldots \qquad . \qquad . \qquad . \\
\quad . \qquad . \qquad . \qquad \ldots \qquad . \qquad . \qquad . \\
a_{n-1,1}\cdot x_1 + a_{n-1,2}\cdot x_2 + a_{n-1,3}\cdot x_3 + \ldots + a_{n-1,m-1}\cdot x_{m-1} + a_{n-1,m}\cdot x_m = b_{n-1}, \\
a_{n1}\cdot x_1 + a_{n2}\cdot x_2 + a_{n3}\cdot x_3 + \ldots + a_{n,m-1}\cdot x_{m-1} + a_{n,m}\cdot x_m = b_n.
\end{array}
$$

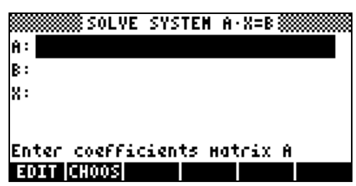
Dieses lineare Gleichungssystem kann als Matrixgleichung $\mathbf{A}_{n\times m}\cdot\mathbf{x}_{m\times 1} = \mathbf{b}_{n\times 1}$ beschrieben werden, wenn wir folgende Matrix und Vektoren definieren:

$$
A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nm} \end{bmatrix}_{n\times m}, \quad x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{bmatrix}_{m\times 1}, \quad b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}_{n\times 1}
$$

Verwenden der numerischen Lösung fur lineare Gleichungssysteme

Lineare Gleichungssysteme können mit dem Taschenrechner auf viele Arten gelöst werden. Eine Möglichkeit besteht in der Verwendung des numerischen Gleichungslösers NUM.SLV . Wählen Sie im unten (links) angezeigten Fenster des numerischen Gleichungslösers die Option *4. Solve lin sys* aus, und drücken Sie OK. Anschließend wird folgende Eingabemaske angezeigt (rechts):





Um das lineare Gleichungssystem **A·x** = **b** zu lösen, geben Sie die Matrix **A** im Format [[a_{11}, $a_{12,}$ …], … [….]] in das Feld A: ein. Geben Sie außerdem den Vektor **b** in das Feld B: ein. Wenn das Feld X: markiert ist, drücken Sie [SOLVE]. Ist eine Lösung verfügbar, wird im Feld X: der Lösungsvektor **x** angezeigt. Die Lösung wird außerdem in Ebene 1 des Stacks kopiert. Es folgen einige Beispiele.

Ein quadratisches Gleichungssystem

Das lineare Gleichungssystem

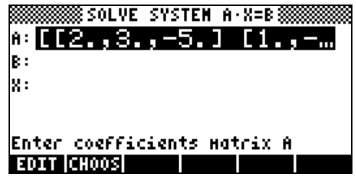
$$2x_1 + 3x_2 -5x_3 = 13,$$
$$x_1 - 3x_2 + 8x_3 = -13,$$
$$2x_1 - 2x_2 + 4x_3 = -6,$$

kann als Matrixgleichung **A·x** = **b** beschrieben werden, wenn

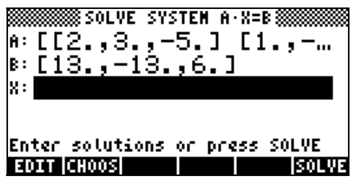
$$\mathbf{A} = \begin{bmatrix} 2 & 3 & -5 \\ 1 & -3 & 8 \\ 2 & -2 & 4 \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}, \quad und \quad \mathbf{b} = \begin{bmatrix} 13 \\ -13 \\ -6 \end{bmatrix}.$$

In diesem Gleichungssystem ist die Anzahl der Gleichungen mit der Anzahl der Unbekannten identisch. Es wird als quadratisches Gleichungssystem bezeichnet. Für dieses System ist grundsätzlich eine eindeutige Lösung vorhanden. Das Gleichungssystem bildet die Schnittmenge der drei Ebenen des durch die drei Gleichungen dargestellten Koordinatensystems (x_1, x_2, x_3).

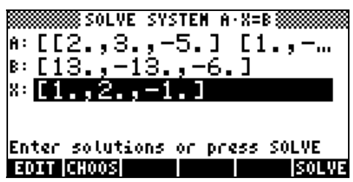
Zum Eingeben von Matrix **A** können Sie MatrixWriter aktivieren, während das Feld A: ausgewählt ist. In den folgenden Bildschirmabbildungen ist MatrixWriter für die Eingabe von Matrix **A** sowie die Eingabemaske für den numerischen Gleichungslöser nach der Eingabe von Matrix **A** dargestellt (drücken Sie in Matrix Writer ENTER):



Drücken Sie ▽, um das Feld B: auszuwählen. Vektor b kann mit einfachen Klammern als Zeilenvektor eingegeben werden, d. h. [13,-13,-6] OK. Nachdem wir Matrix A und Vektor b eingegeben haben und das Feld X: markiert ist, können wir SOLVE drücken, um eine Lösung für dieses Gleichungssystem zu bestimmen:

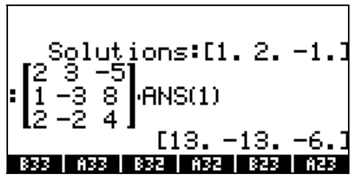


Die Lösung wird unten dargestellt.



Um die Lösung im Stack anzuzeigen, drücken Sie ENTER. Die Lösung lautet **x** = [1,2,-1].

Um die Lösung auf Richtigkeit zu überprüfen, geben Sie Matrix A ein und multiplizieren die Matrix mit diesem Lösungsvektor (Beispiel im algebraischen Modus):



Unterbestimmtes Gleichungssystem
Das lineare Gleichungssystem

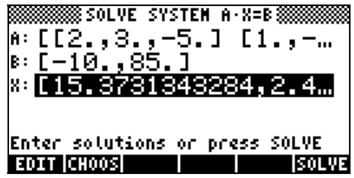
$$2x_1 + 3x_2 - 5x_3 = -10,$$
$$x_1 - 3x_2 + 8x_3 = 85,$$

kann als Matrixgleichung **A·x** = **b** beschrieben werden, wenn

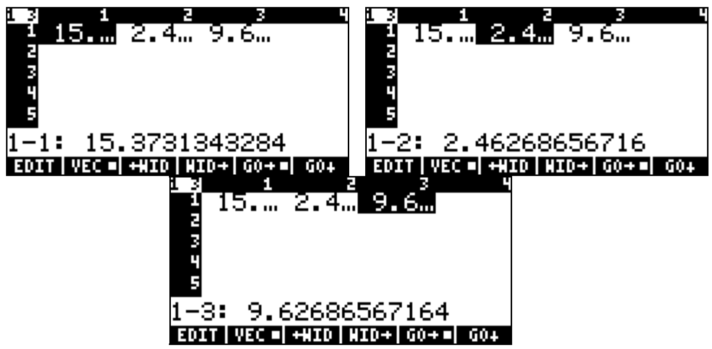
$$\mathbf{A} = \begin{bmatrix} 2 & 3 & -5 \\ 1 & -3 & 8 \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}, \quad und \quad \mathbf{b} = \begin{bmatrix} -10 \\ 85 \end{bmatrix}.$$

Dieses Gleichungssystem verfügt über mehr Unbekannte als Gleichungen und ist daher nicht eindeutig bestimmt. Wir können die Bedeutung dieser Aussage veranschaulichen, wenn wir uns vorstellen, dass jede der linearen Gleichungen eine Ebene in einem dreidimensionalen kartesischen Koordinatensystem (x_1, x_2, x_3) darstellt. Die Lösung des obigen Gleichungssystems stellt die Schnittmenge zweier Ebenen im Raum dar. Wir wissen jedoch, dass die Schnittmenge zweier (nichtparalleler) Ebenen eine gerade Linie und nicht ein einzelner Punkt ist. Daher erfüllen mehrere Punkte die Bedingungen des Gleichungssystems. In diesem Sinn ist das Gleichungssystem nicht eindeutig bestimmt.

Wir suchen nun mit dem numerischen Gleichungslöser nach einer Lösung dieses Gleichungssystems: ⮫ NUM.SLV ▼ ▼ ▼ OK. Geben Sie Matrix A und Vektor b wie im vorherigen Beispiel veranschaulicht ein, und drücken Sie SOLVE, wenn das Feld X: markiert ist:



Um ggf. Details des Lösungsvektors anzuzeigen, drücken Sie die Taste EDIT. Hierdurch wird MatrixWriter aktiviert. Verwenden Sie in MatrixWriter die Tasten mit dem Pfeil nach rechts bzw. nach links, um innerhalb des Vektors zu navigieren, z. B.



Die Lösung lautet somit **x** = [15,373 2,4626 9,6268].

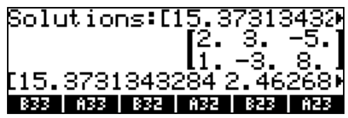
Um zum numerischen Gleichungslöser zurückzukehren, drücken Sie ENTER.

Das im Folgenden beschriebene Verfahren kann zum Kopieren von Matrix A und Lösungsvektor X in den Stack verwendet werden. Um die Richtigkeit der Lösung zu überprüfen, gehen Sie folgendermaßen vor:

- Drücken Sie ▲▲, um das Feld A: zu markieren.
- Drücken Sie NXT CALC ENTER, um Matrix A in den Stack zu kopieren.
- Drücken Sie OK, um zum numerischen Gleichungslöser zurückzukehren.
- Drücken Sie ▼ ▼ CALC ENTER, um den Lösungsvektor X in den Stack zu kopieren.

- Drücken Sie OK, um zum numerischen Gleichungslöser zurückzukehren.
- Drücken Sie ENTER, um zum Stack zurückzukehren.

Der Stack wird nun im ALG-Modus wie folgt angezeigt:

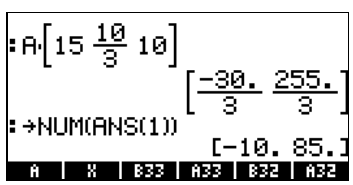


Speichern Sie nun das letzte Ergebnis in einer Variablen X und die Matrix in einer Variablen A:

Drücken Sie STO▸ ALPHA X ENTER, um den Lösungsvektor in der Variablen X zu speichern.
Drücken Sie ◄ ◄ ◄, um drei Ebenen des Stacks zu leeren.
Drücken Sie STO▸ ALPHA A ENTER, um die Matrix in der Variablen A zu speichern.

Überprüfen Sie nun die Lösung, indem Sie A × X ENTER drücken. Dies ergibt folgendes Ergebnis (drücken Sie ▼, um die Vektorelemente anzuzeigen): [-9,99999999999 85]. Dies unterscheidet sich nicht sehr vom ursprünglichen Vektor **b** = [-10 85].

Geben Sie außerdem A × [15,10/3,10] ENTER →NUM ENTER ein, also



Das Ergebnis bedeutet, dass **x** = [15,10/3,10] auch eine Lösung des Gleichungssystems darstellt, und bestätigt unsere Aussage, dass ein Gleichungssystem, das über mehr Unbekannte als Gleichungen verfügt, nicht eindeutig bestimmt (unterbestimmt) ist.

Wie berechnet der Taschenrechner die zuvor dargestellte Lösung **x** = [15,37… 2,46… 9,62…]? Der Taschenrechner minimiert den Abstand von einem Punkt, der die Lösung darstellt, zu jeder der durch die Gleichungen im linearen Gleichungssystem dargestellten Ebenen. Der Taschenrechner verwendet die Methode der kleinsten Quadrate , d. h., die Summe der Quadrate dieser Abstände bzw. Fehler wird minimiert.

Überbestimmtes Gleichungssystem

Das lineare Gleichungssystem

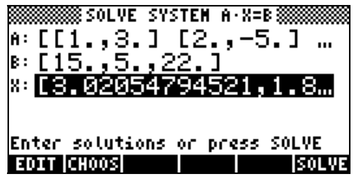
$$x_1 + 3x_2 = 15,$$
$$2x_1 - 5x_2 = 5,$$
$$-x_1 + x_2 = 22,$$

kann als Matrixgleichung **A**·**x** = **b** beschrieben werden, wenn

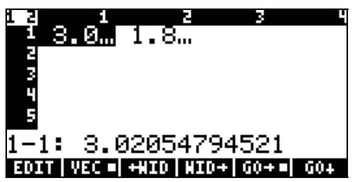
$$\mathbf{A} = \begin{bmatrix} 1 & 3 \\ 2 & -5 \\ -1 & 1 \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \quad und \quad \mathbf{b} = \begin{bmatrix} 15 \\ 5 \\ 22 \end{bmatrix}.$$

Dieses System verfügt über mehr Gleichungen als Unbekannte (überbestimmtes Gleichungssystem). Für das System gibt es keine einzelne Lösung. Jede lineare Gleichung im oben dargestellten Gleichungssystem stellt eine gerade Linie in einem zweidimensionalen kartesischen Koordinatensystem (x_1, x_2) dar. Sofern zwei der drei Gleichungen des Systems nicht dieselbe Gleichung darstellen, besitzen die drei Linien mehr als einen Schnittpunkt. Daher ist die Lösung nicht eindeutig. Mithilfe einiger numerischer Algorithmen kann eine Lösung für das Gleichungssystem erzwungen werden, indem der Abstand vom mutmaßlichen Lösungspunkt zu jeder Linie des Gleichungssystems minimiert wird. Dies ist der vom numerischen Gleichungslöser des HP 49 G verwendete Ansatz.

Wir suchen nun mit dem numerischen Gleichungslöser nach einer Lösung dieses Gleichungssystems: ⮣ NUM.SLV ▼ ▼ ▼ OK. Geben Sie Matrix A und Vektor b wie im vorherigen Beispiel veranschaulicht ein, und drücken Sie SOLVE, wenn das Feld X: markiert ist:



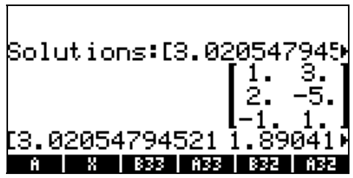
Um ggf. Details des Lösungsvektors anzuzeigen, drücken Sie die Taste EDIT. Hierdurch wird MatrixWriter aktiviert. Verwenden Sie in MatrixWriter die Tasten mit dem Pfeil nach rechts bzw. nach links, um innerhalb des Vektors zu navigieren, z. B.



Drücken Sie ENTER, um zum numerischen Gleichungslöser zurückzukehren. Um die Richtigkeit der Lösung zu überprüfen, gehen Sie folgendermaßen vor:

- Drücken Sie ▲▲, um das Feld A: zu markieren.
- Drücken Sie NXT CALC ENTER, um Matrix A in den Stack zu kopieren.
- Drücken Sie OK, um zum numerischen Gleichungslöser zurückzukehren.
- Drücken Sie ▼ ▼ CALC ENTER, um den Lösungsvektor X in den Stack zu kopieren.
- Drücken Sie OK, um zum numerischen Gleichungslöser zurückzukehren.
- Drücken Sie ENTER, um zum Stack zurückzukehren.

Der Stack wird nun im ALG-Modus wie folgt angezeigt:



Speichern Sie nun das letzte Ergebnis in einer Variablen X und die Matrix in einer Variablen A:

Drücken Sie STO▶ ALPHA X ENTER, um den Lösungsvektor in der Variablen X zu speichern.
Drücken Sie ◄ ◄ ◄, um drei Ebenen des Stacks zu leeren.
Drücken Sie STO▶ ALPHA A ENTER, um die Matrix in der Variablen A zu speichern.

Überprüfen Sie nun die Lösung, indem Sie A × X ENTER drücken, sodass als Ergebnis der Vektor [8.6917… -3.4109… -1.1301…] angezeigt wird. Dies unterscheidet sich von [15 5 22], dem ursprünglichen Vektor **b**. Bei der „Lösung" handelt es sich einfach um den Punkt mit der geringsten Entfernung zu den drei durch die Gleichungen des Systems dargestellten Linien und nicht um eine exakte Lösung.

Lösung nach der Methode der kleinsten Quadrate (Funktion LSQ)

Mit der Funktion LSQ wird eine Lösung nach der Methode der kleinsten Quadrate eines linearen Gleichungssystems Ax = b nach den folgenden Kriterien ausgegeben:

- Wenn **A** eine quadratische nichtsinguläre Matrix ist (d. h., sie verfügt über **A** eine inverse Matrix, oder ihre Determinante ist ungleich Null), gibt LSQ die exakte Lösung des linearen Gleichungssystems zurück.
- Wenn **A** keinen vollen Zeilenrang aufweist (unterbestimmtes Gleichungssystem), gibt LSQ aus einer unendlichen Anzahl von Lösungen die Lösung mit der minimalen euklidischen Länge zurück.
- Wenn **A** keinen vollen Spaltenrang aufweist (überbestimmtes Gleichungssystem), gibt LSQ die „Lösung" mit dem minimalen Residuum **e** = **A**·**x** – **b** zurück. Möglicherweise gibt es keine Lösung für das Gleichungssystem. Daher ist der zurückgegebene Wert keine

echte Lösung des Gleichungssystems, sondern lediglich der Wert mit dem kleinsten Residuum.

Die Eingangswerte für die Funktion LSQ sind Vektor **b** und Matrix **A** in dieser Reihenfolge. Die Funktion LSQ ist über den Funktionskatalog (CAT) verfügbar. Im Folgenden wiederholen wir die zuvor mit dem numerischen Gleichungslöser ermittelten Lösungen mit der Funktion LSQ:

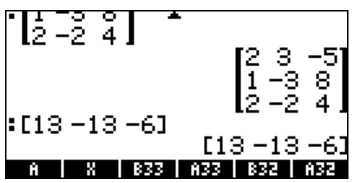
Quadratisches Gleichungssystem

Gegeben sei das System

$$2x_1 + 3x_2 - 5x_3 = 13,$$
$$x_1 - 3x_2 + 8x_3 = -13,$$
$$2x_1 - 2x_2 + 4x_3 = -6,$$

$$\mathbf{A} = \begin{bmatrix} 2 & 3 & -5 \\ 1 & -3 & 8 \\ 2 & -2 & 4 \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}, \quad und \quad \mathbf{b} = \begin{bmatrix} 13 \\ -13 \\ -6 \end{bmatrix}.$$

Die mit LSQ ermittelte Lösung wird unten dargestellt:



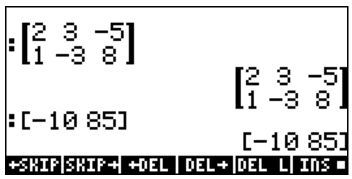
Unterbestimmtes Gleichungssystem

Gegeben sei das System

$$2x_1 + 3x_2 - 5x_3 = -10,$$
$$x_1 - 3x_2 + 8x_3 = 85,$$

$$\mathbf{A} = \begin{bmatrix} 2 & 3 & -5 \\ 1 & -3 & 8 \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}, \quad und \quad \mathbf{b} = \begin{bmatrix} -10 \\ 85 \end{bmatrix}.$$

Die mit LSQ ermittelte Lösung wird unten dargestellt:



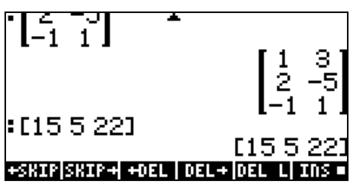
Überbestimmtes Gleichungssystem

Gegeben sei das System

$$x_1 + 3x_2 = 15,$$
$$2x_1 - 5x_2 = 5,$$
$$-x_1 + x_2 = 22,$$

$$\mathbf{A} = \begin{bmatrix} 1 & 3 \\ 2 & -5 \\ -1 & 1 \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \quad und \quad \mathbf{b} = \begin{bmatrix} 15 \\ 5 \\ 22 \end{bmatrix}.$$

Die mit LSQ ermittelte Lösung wird unten dargestellt:



Vergleichen Sie diese drei Lösungen mit den Lösungen, die mit dem numerischen Gleichungslöser berechnet wurden.

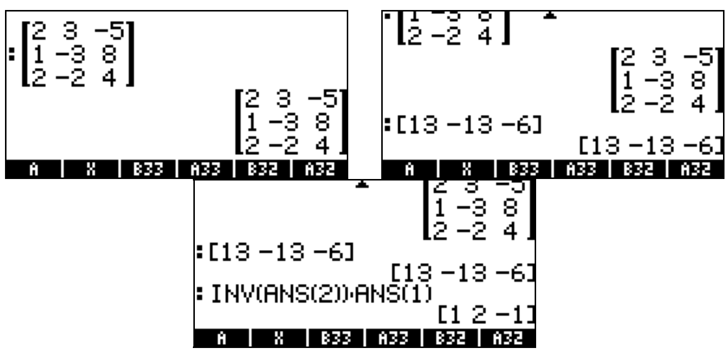
Lösung mit der inversen Matrix

Die Lösung des Gleichungssystems $\mathbf{A}\cdot\mathbf{x} = \mathbf{b}$, wobei $\mathbf{A}$ eine quadratische Matrix ist, lautet $\mathbf{x} = \mathbf{A}^{-1}\cdot\mathbf{b}$. Dieses Ergebnis entsteht durch Multiplikation der ersten Gleichung mit $\mathbf{A}^{-1}$, also $\mathbf{A}^{-1}\cdot\mathbf{A}\cdot\mathbf{x} = \mathbf{A}^{-1}\cdot\mathbf{b}$. Definitionsgemäß ist $\mathbf{A}^{-1}\cdot\mathbf{A} = \mathbf{I}$, daher schreiben wir $\mathbf{I}\cdot\mathbf{x} = \mathbf{A}^{-1}\cdot\mathbf{b}$. Darüber hinaus ist $\mathbf{I}\cdot\mathbf{x} = \mathbf{x}$, somit gilt $\mathbf{x} = \mathbf{A}^{-1}\cdot\mathbf{b}$.

Für das zuvor verwendete Beispiel

$$2x_1 + 3x_2 - 5x_3 = 13,$$
$$x_1 - 3x_2 + 8x_3 = -13,$$
$$2x_1 - 2x_2 + 4x_3 = -6,$$

ermitteln wir mit dem Taschenrechner die Lösung wie folgt:



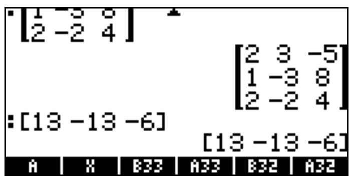
Dies ist mit dem zuvor ermittelten Ergebnis identisch.

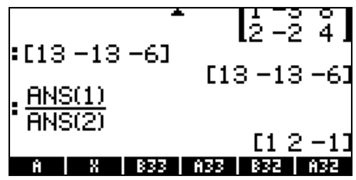
Lösung durch „Division“ von Matrizen

Obwohl die Division für Matrizen nicht definiert ist, können wir mithilfe der Taste [÷] des Taschenrechners Vektor $\mathbf{b}$ durch Matrix $\mathbf{A}$ „dividieren“, um in der Matrixgleichung $\mathbf{A}\cdot\mathbf{x} = \mathbf{b}$ eine Lösung für $\mathbf{x}$ zu finden. Es handelt sich hier um eine willkürliche Erweiterung der algebraischen Division von Matrizen, d. h., aufgrund von $\mathbf{A}\cdot\mathbf{x} = \mathbf{b}$ wagen wir zu schreiben $\mathbf{x} = \mathbf{b}/\mathbf{A}$ (Mathematiker würden schaudern!). Dies wird selbstverständlich als $(1/\mathbf{A})\cdot\mathbf{b} = \mathbf{A}^{-1}\cdot\mathbf{b}$ interpretiert und entspricht der Verwendung der Inversen von $\mathbf{A}$ im vorherigen Abschnitt. Das Verfahren für die „Division“ von **b** durch **A** wird unten für

$$\begin{aligned} 2x_1 + 3x_2 - 5x_3 &= 13, \\ x_1 - 3x_2 + 8x_3 &= -13, \\ 2x_1 - 2x_2 + 4x_3 &= -6, \end{aligned}$$

veranschaulicht. In den folgenden Bildschirmabbildungen wird das Verfahren dargestellt:





Es handelt sich um dieselbe Lösung, die oben mit der inversen Matrix ermittelt wurde.

Lösen mehrerer Gruppen von Gleichungen mit derselben Koeffizientenmatrix

Angenommen Sie möchten die folgenden drei Gruppen von Gleichungen lösen:

X + 2Y + 3Z = 14,	2X + 4Y + 6Z = 9,	2X + 4Y + 6Z = -2,
3X - 2Y + Z = 2,	3X - 2Y + Z = -5,	3X - 2Y + Z = 2,
4X + 2Y - Z = 5,	4X + 2Y - Z = 19,	4X + 2Y -Z = 12.

Die drei Gleichungssysteme können als eine einzige Matrixgleichung dargestellt werden: **A**·**X** = **B**, mit

$$\mathbf{A} = \begin{bmatrix} 1 & 2 & 3 \\ 3 & -2 & 1 \\ 4 & 2 & -1 \end{bmatrix}, \quad \mathbf{X} = \begin{bmatrix} X_{(1)} & X_{(2)} & X_{(3)} \\ Y_{(1)} & Y_{(2)} & Y_{(3)} \\ Z_{(1)} & Z_{(2)} & Z_{(3)} \end{bmatrix},$$

$$\mathbf{B} = \begin{bmatrix} 14 & 9 & -2 \\ 2 & -5 & 2 \\ 5 & 19 & 12 \end{bmatrix}.$$

Die Indizes in den Variablennamen X, Y und Z geben an, auf welches Gleichungssystem sie sich beziehen. Zur Lösung dieses erweiterten Systems verwenden wir im RPN-Modus folgendes Verfahren:

```
[[14,9,-2],[2,-5,2],[5,19,12]] ENTER
[[1,2,3],[3,-2,1],[4,2,-1]] ENTER ÷
```

Das Ergebnis dieser Operation lautet:

$$\mathbf{X} = \begin{bmatrix} 1 & 2 & 2 \\ 2 & 5 & 1 \\ 3 & -1 & -2 \end{bmatrix}.$$

Gauß- und Gauß-Jordan-Elimination

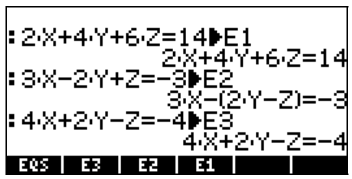
Bei der Gauß-Elimination wird eine quadratische Koeffizientenmatrix, die einem System mit n linearen Gleichungen und n Unbekannten angehört, über mehrere Zeilenoperationen zu einer oberen Dreiecksmatrix (*Staffelform*) reduziert. Dieses Verfahren wird als *Vorwärtssubstitution* bezeichnet. Aufgrund der Reduzierung der Koeffizientenmatrix zu einer oberen Dreiecksmatrix kann mit einem als *Rückwärtssubstitution* bezeichneten Verfahren, bei dem jeweils nur eine Gleichung bearbeitet wird, eine Lösung für alle *n* Unbekannten ermittelt werden.

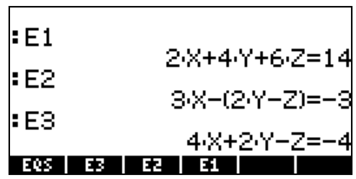
Beispiel für die Gauß-Elimination mit Gleichungen

Zur Veranschaulichung der Gauß-Elimination verwenden wir folgendes System mit 3 Gleichungen und 3 Unbekannten:

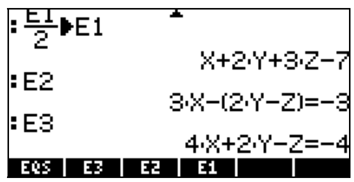
$$2X + 4Y + 6Z = 14,$$
$$3X - 2Y + Z = -3,$$
$$4X + 2Y - Z = -4.$$

Wir speichern diese Gleichungen mit dem Taschenrechner in den Variablen E1, E2 bzw. E3, wie unten dargestellt. Für Backup-Zwecke wurde außerdem eine Liste mit den drei Gleichungen erstellt und in der Variablen EQS gespeichert. Falls eine fehlerhafte Eingabe erfolgt, sind die Gleichungen somit dennoch für den Benutzer verfügbar.

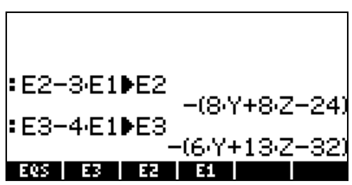


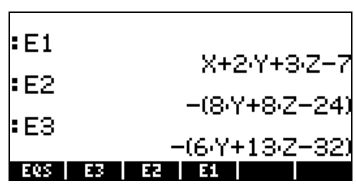


Zu Beginn der Vorwärtssubstitution dividieren wir die erste Gleichung (E1) durch 2, speichern sie in E1 und zeigen die drei Gleichungen erneut an:

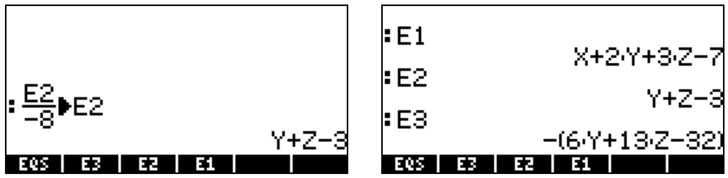


Anschließend ersetzen wir die zweite Gleichung E2 durch (Gleichung 2-3×Gleichung 1, also E1-3×E2) und die dritte Gleichung durch (Gleichung 3-4×Gleichung 1) und erhalten

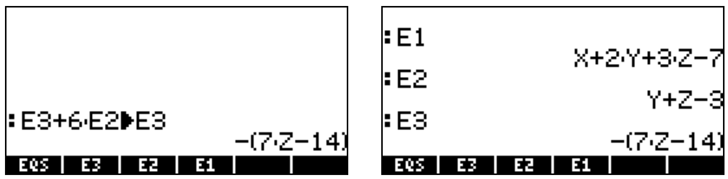




Dann dividieren wir die zweite Gleichung durch -8 und erhalten



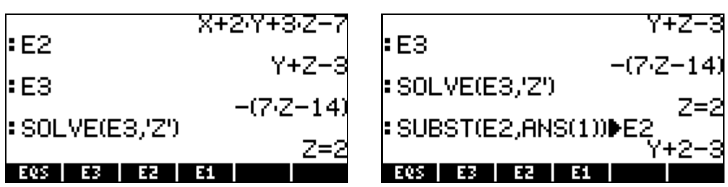
Anschließend ersetzen wir die dritte Gleichung E3 durch (Gleichung 3+6×Gleichung 2, also E2+6×E3) und erhalten



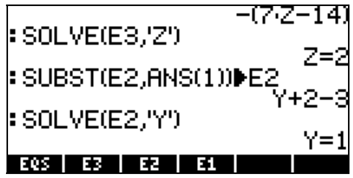
Beachten Sie, dass der Taschenrechner beim Ausführen einer Linearkombination von Gleichungen das Ergebnis in einen Ausdruck auf der linken Seite der Gleichung ändert, d. h. Ausdruck = 0. Die letzte Gruppe von Gleichungen wird somit als folgende äquivalente Gruppe von Gleichungen interpretiert:

$$X + 2Y + 3Z = 7,$$
$$Y + Z = 3,$$
$$-7Z = -14.$$

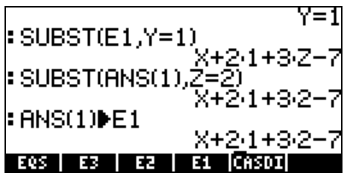
Bei der Gauß-Elimination besteht die Rückwärtssubstitution darin, dass die Werte der Unbekannten ermittelt werden, wobei mit der letzten Gleichung begonnen und der Vorgang mit den jeweils oberen Gleichungen fortgesetzt wird. Wir ermitteln daher zuerst Z:

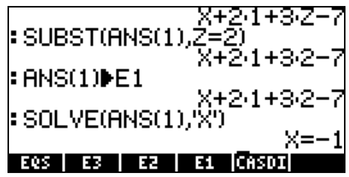


Dann setzen wir in Gleichung 2 (E2) für Z=2 ein und ermitteln Y in E2:



Anschließend setzen wir in E1 für Z=2 und für Y=1 ein und ermitteln X in E1:





Die Lösung lautet somit X = -1, Y = 1, Z = 2.

Beispiel für die Gauß-Elimination mit Matrizen

Das im obigen Beispiel verwendete Gleichungssystem kann als Matrixgleichung **A**·**x** = **b** dargestellt werden, wenn wir schreiben:

$$\mathbf{A} = \begin{pmatrix} 2 & 4 & 6 \\ 3 & -2 & 1 \\ 4 & 2 & -1 \end{pmatrix}, \quad \mathbf{x} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 14 \\ -3 \\ -4 \end{bmatrix}.$$

Um mithilfe der Gauß-Elimination eine Lösung für die Matrix des Gleichungssystems zu erhalten, erstellen wir zunächst eine **A** entsprechende, so genannte <u>*erweiterte Matrix*</u>, also

$$\mathbf{A}_{aug} = \left(\begin{array}{ccc|c} 2 & 4 & 6 & 14 \\ 3 & -2 & 1 & -3 \\ 4 & 2 & -1 & -4 \end{array}\right)$$

Bei Matrix **A**$_{aug}$ handelt es sich um die ursprüngliche Matrix **A** mit einer neuen Zeile, die den Elementen von Vektor **b** entspricht und rechts von der äußersten rechten Spalte von **A** eingefügt (d. h. erweitert) wird.

Nachdem die erweiterte Matrix gebildet wurde, können wir mit ihr Zeilenoperationen durchführen, mit denen die ursprüngliche Matrix A zu einer oberen Dreiecksmatrix reduziert wird. Hierfür verwenden wir den RPN-Modus (MODE +/- OK), wobei Systemflag 117 auf Soft MENU gesetzt ist. Verwenden Sie für den Taschenrechner folgende Tastenkombinationen. Geben Sie zunächst die erweiterte Matrix ein, und erstellen Sie im Stack eine Kopie der Matrix. (Dieser Schritt ist nicht erforderlich, dient jedoch der Absicherung, damit Sie über eine Kopie der erweiterten Matrix verfügen, falls Ihnen bei der durchzuführenden Vorwärtssubstitution ein Fehler unterläuft.):

`[[2,4,6,14],[3,-2,1,-3],[4,2,-1,-4]]` ENTER ENTER

Speichern Sie die erweiterte Matrix in der Variablen AAUG: ' ALPHA ALPHA A A U G ALPHA STO▶

Wenn eine Kopie der erweiterten Matrix im Stack vorhanden ist, drücken Sie ← MTH MATRX ROW, um das Menü ROW zu aktivieren. Wenden Sie anschließend die folgenden Zeilenoperationen auf die erweiterte Matrix an:
Multiplizieren Sie Zeile 1 mit ½: 2 1/x 1 RCI

Multiplizieren Sie Zeile 1 mit -3, und fügen Sie sie Zeile 2 hinzu, indem Sie diese ersetzen: 3 +/- SPC 1 SPC 2 RCIJ

Multiplizieren Sie Zeile 1 mit -4, und fügen Sie sie Zeile 3 hinzu, indem Sie diese ersetzen: 4 +/- SPC 1 SPC 3 RCIJ

Multiplizieren Sie Zeile 2 mit -1/8: 8 +/- 1/x 2 RCI
Multiplizieren Sie Zeile 2 mit 6, und fügen Sie sie Zeile 3 hinzu, indem Sie diese ersetzen: 6 SPC 2 SPC 3 RCIJ

Wenn Sie diese Operationen manuell durchführen, gehen Sie folgendermaßen vor:

$$\mathbf{A}_{aug} = \left(\begin{array}{ccc|c} 2 & 4 & 6 & 14 \\ 3 & -2 & 1 & -3 \\ 4 & 2 & -1 & -4 \end{array}\right) \cong \left(\begin{array}{ccc|c} 1 & 2 & 3 & 7 \\ 3 & -2 & 1 & -3 \\ 4 & 2 & -1 & -4 \end{array}\right)$$

$$\mathbf{A}_{aug} \cong \left(\begin{array}{ccc|c} 1 & 2 & 3 & 7 \\ 0 & -8 & -8 & -24 \\ 0 & -6 & -13 & -32 \end{array}\right) \cong \left(\begin{array}{ccc|c} 1 & 2 & 3 & 7 \\ 0 & 1 & 1 & 3 \\ 0 & -6 & -13 & -32 \end{array}\right)$$

$$\mathbf{A}_{aug} \cong \left(\begin{array}{ccc|c} 1 & 2 & 3 & 7 \\ 0 & 1 & 1 & 3 \\ 0 & 0 & -7 & -14 \end{array}\right)$$

Das Symbol ≅ („entspricht“) gibt an, dass der folgende Ausdruck mit der vorherigen Matrix, auf die einige Zeilenoperationen (bzw. Spaltenoperationen) angewendet wurden, äquivalent ist.

Die resultierende Matrix ist eine obere Dreiecksmatrix und äquivalent mit der Gruppe von Gleichungen

X +2Y+3Z = 7,
Y+ Z = 3,
-7Z = -14,

Diese können nun wie im vorherigen Beispiel durch Rückwärtssubstitution einzeln nacheinander gelöst werden.

Gauß-Jordan-Elimination mit Matrizen

Bei der Gauß-Jordan-Elimination werden die Zeilenoperationen in der aus der Rückwärtssubstitution resultierenden oberen Dreiecksmatrix fortgesetzt, bis anstelle der ursprünglichen Matrix **A** eine Einheitsmatrix gebildet wurde. Beispielsweise können wir im gerade dargestellten Fall die Zeilenoperationen wie folgt fortsetzen:

Multiplizieren Sie Zeile 3 mit -1/7: `7` `+/-` `1/x` `3`

Multiplizieren Sie Zeile 3 mit -1, und fügen Sie sie Zeile 2 hinzu, indem Sie diese ersetzen: 1 +/- SPC 3 SPC 2 RCIJ

Multiplizieren Sie Zeile 3 mit -3, und fügen Sie sie Zeile 1 hinzu, indem Sie diese ersetzen: 3 +/- SPC 3 SPC 1 RCIJ

Multiplizieren Sie Zeile 2 mit -2, und fügen Sie sie Zeile 1 hinzu, indem Sie diese ersetzen: 2 +/- SPC 2 SPC 1 RCIJ

Wenn Sie diesen Vorgang manuell durchführen, ergeben sich folgende Schritte:

$$\mathbf{A}_{aug} = \left(\begin{array}{ccc|c} 1 & 2 & 3 & 7 \\ 0 & 1 & 1 & 3 \\ 0 & 0 & -7 & -14 \end{array}\right) \cong \left(\begin{array}{ccc|c} 1 & 2 & 3 & 7 \\ 0 & 1 & 1 & 3 \\ 0 & 0 & 1 & 2 \end{array}\right) \cong \left(\begin{array}{ccc|c} 1 & 2 & 3 & 7 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 2 \end{array}\right)$$

$$A_{aug} \cong \left(\begin{array}{ccc|c} 1 & 2 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 2 \end{array}\right) \cong \left(\begin{array}{ccc|c} 1 & 0 & 0 & -1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 2 \end{array}\right).$$

Pivotisierung

Wenn Sie die Zeilenoperationen in den oben dargestellten Beispielen sorgfältig untersuchen, werden Sie feststellen, dass durch viele dieser Operationen eine Zeile durch das entsprechende Element in der Hauptdiagonalen dividiert wird. Dieses Element wird als Pivot-Element bezeichnet. In zahlreichen Fällen kann das Pivot-Element den Wert Null annehmen, sodass die Zeile nicht durch ihr Pivot-Element dividiert werden kann. Zur Vereinfachung der numerischen Lösung eines Gleichungssystems mit der Gauß- oder Gauß-Jordan-Elimination empfiehlt es sich außerdem, als Pivot-Element das Element mit dem größten absoluten Wert in einer Spalte zu verwenden. In diesen Fällen vertauschen wir Zeilen vor der Anwendung der Zeilenoperationen. Dieses Vertauschen von Zeilen wird als Teilpivotisierung bezeichnet. Zum Befolgen dieser Empfehlung müssen häufig Zeilen in der erweiterten Matrix vertauscht werden, wenn eine Gauß- oder Gauß-Jordan-Elimination durchgeführt wird.

Bei der Pivotisierung während einer Matrixelimination können Sie die numerische Lösung noch weiter vereinfachen, indem Sie das Element mit dem größten absoluten Wert in der betreffenden Spalte bzw. Zeile als Pivot-Element auswählen. Dies erfordert möglicherweise, dass bei einigen Pivotisierungsoperationen nicht nur Zeilen, sondern auch Spalten vertauscht werden. Wenn bei der Pivotisierung Zeilen- und Spaltentausch zulässig ist, wird dieses Verfahren als Totalpivotisierung bezeichnet.

Beim Vertauschen von Zeilen und Spalten bei der Teil- bzw. Totalpivotisierung ist es erforderlich, die Tauschvorgänge aufzuzeichnen, da hierbei die Anordnung der Unbekannten in der Lösung geändert wird. Eine Möglichkeit zum Aufzeichnen der Spaltentauschvorgänge bei der Teil- bzw. Totalpivotisierung besteht darin, zu Beginn des Verfahrens eine *Permutationsmatrix* $\mathbf{P} = \mathbf{I}_{n\times n}$ zu erstellen. Jeder in der erweiterten Matrix $\mathbf{A}_{aug}$ erforderliche Zeilen- oder Spaltentausch wird auch in der Permutationsmatrix als Zeilen- bzw. Spaltentausch registriert. Wenn die Lösung ermittelt wurde, multiplizieren wir die Permutationsmatrix mit dem Vektor der Unbekannten **x**, um die richtige Anordnung der Unbekannten in der Lösung zu erhalten. Mit anderen Worten, die endgültige Lösung ist durch $\mathbf{P}\cdot\mathbf{x} = \mathbf{b}'$ definiert, wobei $\mathbf{b}'$ die letzte Spalte in der erweiterten Matrix darstellt, nachdem die Lösung ermittelt wurde.

Beispiel für Gauß-Jordan-Elimination mit Totalpivotisierung

Die Totalpivotisierung wird anhand eines Beispiels veranschaulicht. Lösen Sie das folgende Gleichungssystem unter Verwendung von Totalpivotisierung und Gauß-Jordan-Elimination:

$$X + 2Y + 3Z = 2,$$
$$2X + 3Z = -1,$$
$$8X + 16Y - Z = 41.$$

Die erweiterte Matrix und die Permutationsmatrix lauten wie folgt:

$$\mathbf{A}_{aug} = \begin{bmatrix} 1 & 2 & 3 & 2 \\ 2 & 0 & 3 & -1 \\ 8 & 16 & -1 & 41 \end{bmatrix}, \quad \mathbf{P} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Speichern Sie die erweiterte Matrix in der Variablen AAUG, und drücken Sie dann ‚ AAUG, um die erweiterte Matrix in den Stack zu kopieren. Wir möchten, dass der Befehl CSWP (Spalten vertauschen) verfügbar bleibt, für den wir Folgendes eingeben: ‚ CAT ALPHA ALPHA C S ALPHA (CSWP suchen), OK. Sie erhalten eine Fehlermeldung. Drücken Sie $, und ignorieren Sie die Meldung.

Machen Sie anschließend das Menü ROW verfügbar, indem Sie „ MATRICES CREAT ROW drücken.

Nun können wir mit der Gauß-Jordan-Elimination mit Totalpivotisierung beginnen. Wir müssen die Permutationsmatrix per Hand aufzeichnen, tragen Sie also die oben dargestellte Matrix **P** in Ihr Notizbuch ein.

Zunächst überprüfen wir das Pivot-Element a_{11}. Wir stellen fest, dass das Element mit dem größten absoluten Wert in der ersten Zeile und ersten Spalte der Wert a_{31} = 8 ist. Da diese Zahl als Pivot-Element verwendet werden soll, vertauschen wir die Zeilen 1 und 3 mit dem Befehl 1 SPC 3 NXT RSWP. Die erweiterte Matrix und die Permutationsmatrix lauten nun:

$$\begin{bmatrix} 8 & 16 & -1 & 41 \\ 2 & 0 & 3 & -1 \\ 1 & 2 & 3 & 2 \end{bmatrix} \quad \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Beim Überprüfen des Pivot-Elements an Position (1,1) stellen wir fest, dass 16 besser als Pivot-Element geeignet ist als 8. Daher führen wir mit folgendem Befehl einen Spaltentausch durch: 1 SPC 2 ‚ CAT OK RSWP. Die erweiterte Matrix und die Permutationsmatrix lauten nun:

$$\begin{bmatrix} 16 & 8 & -1 & 41 \\ 0 & 2 & 3 & -1 \\ 2 & 1 & 3 & 2 \end{bmatrix} \quad \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

Der größte mögliche Wert befindet sich jetzt an Position (1,1), d. h., wir haben an Position (1,1) eine Totalpivotisierung durchgeführt. Anschließend dividieren wir durch das Pivot-Element:
[1][6][1/x][1][NXT] [RCI]. Die Permutationsmatrix bleibt unverändert, doch die erweiterte Matrix lautet nun:

$$\begin{bmatrix} 1 & 1/2 & -1/16 & 41/16 \\ 0 & 2 & 3 & -1 \\ 2 & 1 & 3 & 2 \end{bmatrix} \qquad \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

Als Nächstes entfernen wir die 2 aus Position (3,2):
[2][+/-][SPC][1][SPC][3][RCIJ]

$$\begin{bmatrix} 1 & 1/2 & -1/16 & 41/16 \\ 0 & 2 & 3 & -1 \\ 0 & 0 & 25/8 & -25/8 \end{bmatrix} \qquad \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

Nachdem wir die Elemente von Spalte 1 unter dem Pivot-Element mit Nullen aufgefüllt haben, überprüfen wir das Pivot-Element an Position (2,2). Wir stellen fest, dass die Zahl 3 an Position (2,3) besser als Pivot-Element geeignet ist und vertauschen daher Spalte 2 und 3 durch die Eingabe: [2][SPC][3] [→] _CAT [OK]

$$\begin{bmatrix} 1 & -1/16 & 1/2 & 41/16 \\ 0 & 3 & 2 & -1 \\ 0 & 25/8 & 0 & -25/82 \end{bmatrix} \qquad \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Bei Überprüfung des Pivot-Elements an Position (2,2) stellen wir fest, dass der Wert 25/8 an Position (3,2) größer als 3 ist. Wir vertauschen daher Spalte 2 und 3 durch die Eingabe: [2][SPC][3] [NXT][RSWP]

$$\begin{bmatrix} 1 & -1/16 & 1/2 & 41/16 \\ 0 & 25/8 & 0 & -25/8 \\ 0 & 3 & 2 & -1 \end{bmatrix} \qquad \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

Nun können wir Spalte 2 durch das Pivot-Element 25/8 dividieren, indem wir ['][8][÷][2][5][▶][SPC][2] [NXT] [RCI] eingeben.

$$\begin{bmatrix} 1 & -1/16 & 1/2 & 41/16 \\ 0 & 1 & 0 & -1 \\ 0 & 3 & 2 & -1 \end{bmatrix} \quad \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

Anschließend entfernen wir die 3 aus Position (3,2) durch folgende Eingabe: [3][+/-][SPC][2][SPC][3][RCIJ]

$$\begin{bmatrix} 1 & -1/16 & 1/2 & 41/16 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 2 & 2 \end{bmatrix} \quad \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

Nachdem wir die Stellen unter dem Pivot-Element mit Nullen aufgefüllt haben, überprüfen wir das Pivot-Element an Position (3,3). Der aktuelle Wert 2 ist größer als ½ oder 0, daher lassen wir ihn unverändert. Wir dividieren die ganze dritte Reihe durch 2, um das Pivot-Element in 1 umzuwandeln, indem wir folgende Eingabe verwenden: [2][1/x][3][RCI]

$$\begin{bmatrix} 1 & -1/16 & 1/2 & 41/16 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & 1 \end{bmatrix} \quad \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

Anschließend entfernen wir ½ an Position (1,3) durch die Eingabe: [2] [1/x] [+/-][SPC][3][SPC][1][RCIJ]

$$\begin{bmatrix} 1 & -1/16 & 0 & 33/16 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & 1 \end{bmatrix} \quad \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

Schließlich entfernen wir -1/16 aus Position (1,2) durch folgende Eingabe: [1][6] [1/x] [SPC] [2][SPC][1][RCIJ]

$$\begin{bmatrix} 1 & 0 & 0 & 2 \\ 0 & 1 & 0 & -1 \end{bmatrix} \quad \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

0 0 1 1 1 0 0

Nun verfügen wir über eine Einheitsmatrix in dem der ursprünglichen Koeffizientenmatrix A entsprechenden Abschnitt der erweiterten Matrix und können mithilfe des in Permutationsmatrix **P** codierten Zeilen- und Spaltentausches die Lösung ermitteln. Wir bestimmen den Vektor der Unbekannten **x**, den Vektor der geänderten Unabhängigen **b**' und die Permutationsmatrix **P** wie folgt:

$$\mathbf{x} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}, \quad \mathbf{b'} = \begin{bmatrix} 2 \\ -1 \\ 1 \end{bmatrix}, \quad \mathbf{P} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}.$$

Die Lösung lautet **P**·**x**=**b**′ oder

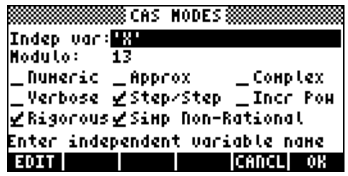
$$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 3 \\ -1 \\ 1 \end{bmatrix}.$$

Dies ergibt

$$\begin{bmatrix} Y \\ Z \\ X \end{bmatrix} = \begin{bmatrix} 3 \\ -1 \\ 1 \end{bmatrix}.$$

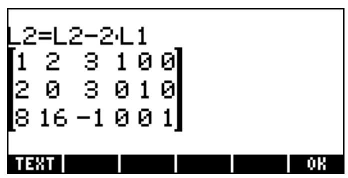
Schrittweises Verfahren für den Taschenrechner zum Lösen linearer Gleichungssysteme

Bei dem gerade erläuterten Beispiel handelt es sich natürlich um ein schrittweises, vom Benutzer durchgeführtes Verfahren zur Lösung linearer Gleichungssysteme mit Totalpivotisierung für die Gauß-Jordan-Elimination. Sie können das vom Taschenrechner verwendete schrittweise Verfahren zum Lösen eines Gleichungssystems ohne Benutzereingaben anzeigen, indem Sie in einem CAS MODES-Fenster des Taschenrechners die Option Step/Step wie folgt auswählen.



Verwenden Sie dann für dieses Beispiel im RPN-Modus folgende Eingabe:
`[2,-1,41]` ENTER `[[1,2,3],[2,0,3],[8,16,-1]]` ENTER ÷

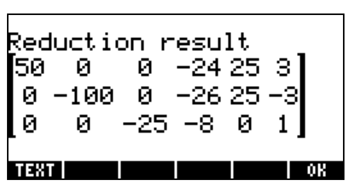
Der Taschenrechner zeigt eine erweiterte Matrix an, die aus der Koeffizientenmatrix **A** und der Einheitsmatrix **I** besteht, während gleichzeitig die nächste Berechnung angezeigt wird.



L2 = L2-2·L1 bedeutet „Reihe 2 (L2) durch die Operation L2-2 L1 ersetzen". Bei der manuellen Ausführung dieser Operation würde dies folgender Eingabe entsprechen: 2 +/- SPC 1 SPC 1 RCI. Drücken Sie OK, und beachten Sie die auf dem Bildschirm des Taschenrechners angezeigten Operationen. Es wird die Ausführung der folgenden Operationen angezeigt:

L3=L3-8·L1, L1 = 2·L1--1·L2, L1=25·L1--3·L3, L2 = 25·L2-3·L3,

und schließlich die Meldung, dass das Ergebnis der Reduzierung („Reduction result") angezeigt wird.



Wenn Sie OK drücken, gibt der Taschenrechner das Endergebnis [1 2 –1] aus.

Schrittweises Berechnen der inversen Matrix

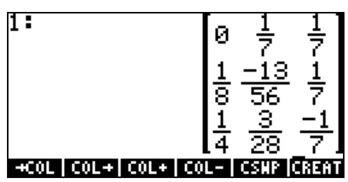
Die Berechnung einer inversen Matrix kann als Berechnung der Lösung eines erweiterten Systems [**A** | **I**] betrachtet werden. Beispielsweise würden wir für Matrix **A** aus dem vorherigen Beispiel die erweiterte Matrix wie folgt schreiben:

$$\mathbf{A}_{aug(I)} = \left[\begin{array}{ccc|ccc} 1 & 2 & 3 & 1 & 0 & 0 \\ 3 & -2 & 1 & 0 & 1 & 0 \\ 4 & 2 & -1 & 0 & 0 & 1 \end{array}\right].$$

Um die Zwischenschritte bei der Berechnung sowie die Inverse anzuzeigen, geben Sie einfach Matrix **A** aus dem vorherigen Beispiel ein, und drücken Sie 1/x, während im CAS-MODES-Fenster des Taschenrechners die Option Step/Step aktiviert ist. Verwenden Sie folgende Eingabe:

```
[[ 1,2,3],[3,-2,1],[4,2,-1]] ENTER 1/x
```

Nach Ausführung der einzelnen Schritte lautet die ausgegebene Lösung:



Der Taschenrechner hat nicht eigentlich eine Gauß-Jordan-Elimination mit Totalpivotisierung angezeigt, sondern ein Verfahren zum Berechnen der Inversen einer Matrix mithilfe einer Gauß-Jordan-Elimination ohne Pivotisierung. Dieses Verfahren zum Berechnen der Inversen beruht auf der erweiterten Matrix $(\mathbf{A}_{aug})_{n\times n} = [\mathbf{A}_{n\times n} \,|\mathbf{I}_{n\times n}]$.

Der Taschenrechner zeigt die Schritte bis zu dem Punkt an, an dem die linke Seite der erweiterten Matrix in eine Diagonalmatrix umgewandelt wurde. Nun besteht der letzte Schritt im Dividieren jeder Zeile durch das entsprechende Pivot-Element der Hauptdiagonalen. Mit anderen Worten, der Taschenrechner hat $(\mathbf{A}_{aug})_{n\times n}$ = [$\mathbf{A}_{n\times n}$ |$\mathbf{I}_{n\times n}$] in [$\mathbf{I}$ |$\mathbf{A}^{-1}$], konvertiert.

Inverse Matrizen und Determinanten

Beachten Sie, dass alle Elemente in der oben berechneten inversen Matrix durch den Wert 56 oder einen seiner Faktoren (28, 7, 8, 4 oder 1) dividiert werden. Wenn Sie die Determinante von Matrix **A** berechnen, erhalten Sie *det*(**A**) = 56.

Wir können $\mathbf{A}^{-1} = \mathbf{C}/det(\mathbf{A})$ schreiben, wobei **C** folgende Matrix darstellt:

$$\mathbf{C} = \begin{bmatrix} 0 & 8 & 8 \\ 7 & -13 & 8 \\ 14 & 6 & -8 \end{bmatrix}.$$

Bei dem Ergebnis $(\mathbf{A}^{-1})_{n\times n} = \mathbf{C}_{n\times n}/det(\mathbf{A}_{n\times n})$ handelt es sich um ein generisches Ergebnis, das auf eine beliebige nichtsinguläre Matrix **A** zutrifft. Auf der Grundlage des Gauß-Jordan-Algorithmus kann eine allgemeine Darstellung der Elemente von C geschrieben werden.

Aufgrund der oben skizzierten Gleichung $\mathbf{A}^{-1} = \mathbf{C}/\det(\mathbf{A})$ ist die inverse Matrix $\mathbf{A}^{-1}$ nicht definiert, wenn *det*(**A**) = 0. Die Bedingung *det*(**A**) = 0 definiert somit auch eine singuläre Matrix.

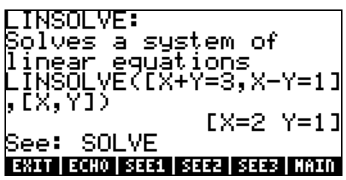
Losung linearer Gleichungssysteme mit den Taschenrechnerfunktionen

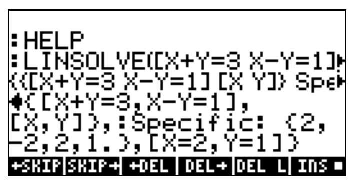
Die einfachste Möglichkeit zum Lösen eines linearen Gleichungssystems $\mathbf{A}\cdot\mathbf{x} = \mathbf{b}$ mit dem Taschenrechner besteht darin, **b** einzugeben, **A** einzugeben und anschließend die Divisionsfunktion / zu verwenden. Wenn das lineare Gleichungssystem überbestimmt oder unterbestimmt ist, kann mit der Funktion LSQ (Least SQares, kleinste Quadrate) eine „Lösung" ermittelt werden. Mit den Funktionen des Menüs MATRICES' LINEAR SYSTEMS, das über ← MATRICES aufgerufen werden kann (setzen Sie Systemflag 117 auf CHOOSE boxes), bietet der Taschenrechner jedoch andere Möglichkeiten zum Lösen linearer Gleichungssysteme.

Die Funktionen dieses Menüs lauten LINSOLVE, REF, rref, RREF und SYST2MAT.

Funktion LINSOLVE

Als Argumente der Funktion LINSOLVE werden ein Feld von Gleichungen und ein Vektor verwendet, der die Namen der Unbekannten enthält. Die Funktion ermittelt die Lösung linearer Gleichungssysteme. In den folgenden Fenstern wird der Eintrag der Hilfefunktion (siehe Kapitel 1) für LINSOLVE und das zugehörige, im Eintrag aufgeführte Beispiel dargestellt. Im rechten Fenster wird das unter Verwendung des Befehlszeileneditors (drücken Sie zum Aktivieren) erzeugte Ergebnis angezeigt:





Es folgt ein weiteres Beispiel im ALG-Modus. Geben Sie Folgendes ein:

```
LINSOLVE([X-2*Y+Z=-8,2*X+Y-2*Z=6,5*X-2*Y+Z=-12],
                    [X,Y,Z])
```

Die Lösung lautet: `[X=-1,Y=2,Z = -3]`.

Für die Funktion LINSOLVE werden symbolische Ausdrücke verwendet. Die Funktionen REF, rref und RREF werden für die erweiterte Matrix bei der Gauß-Elimination verwendet.

Funktionen REF, rref und RREF

Die obere Dreiecksmatrix, zu der die erweiterte Matrix bei der Vorwärtssubstitution im Rahmen der Gauß-Elimination reduziert wird, wird als Staffelform bezeichnet. Die Funktion REF (Reduce to Echelon Form, zu Staffelform reduzieren) erzeugt eine solche Matrix, wenn die erweiterte Matrix auf Ebene 1 des Stacks vorhanden ist.

Gegeben sei die erweiterte Matrix

$$\mathbf{A}_{aug} = \left[\begin{array}{ccc|c} 1 & -2 & 1 & 0 \\ 2 & 1 & -2 & -3 \\ 5 & -2 & 1 & 12 \end{array}\right].$$

Sie stellt ein lineares Gleichungssystem **A**·**x** = **b** dar, mit

A = `[[1,-2,1],[2,1,-2],[5,-2,1]],`

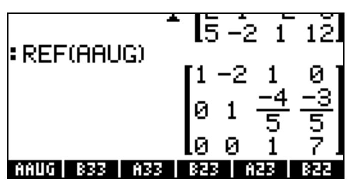
und

b = `[[0],[-3],[12]].`

Geben Sie die erweiterte Matrix ein, und speichern Sie sie im ALG-Modus in der Variablen AAUG:

`[[1,-2,1,0],[2,1,-2,-3][5,-2,1,12] ▶ AAUG`

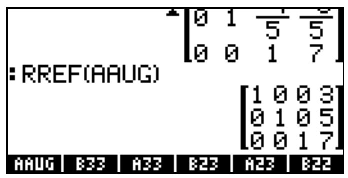
Die Anwendung der Funktion REF erzeugt die Ausgabe:



Das Ergebnis ist die obere Dreieckskoeffizientenmatrix (Staffelform), die aus der Vorwärtssubstitution bei der Gauß-Elimination resultiert.

Die als Ergebnis der Gauß-Jordan-Elimination gebildete Diagonalmatrix wird als zeilenreduzierte Staffelform (bzw. reduzierte Staffelform) bezeichnet. Funktion RREF (Row-Reduced Echelon Form, zeilenreduzierte Staffelform): Durch Aufruf dieser Funktion wird eine zeilenreduzierte Staffelform erzeugt, sodass die Koeffizientenmatrix zu einer Einheitsmatrix reduziert wird. Die zusätzliche Spalte der erweiterten Matrix enthält die Lösung des Gleichungssystems.

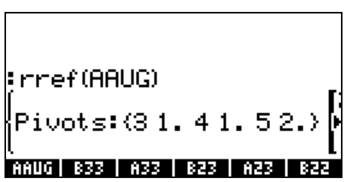
Als Beispiel wird das Ergebnis der Anwendung der Funktion RREF auf die Matrix AAUG im ALG-Modus dargestellt:

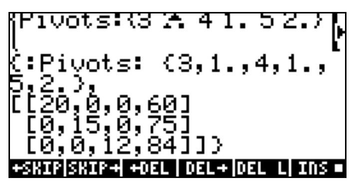


Das Ergebnis ist die durch Gauß-Jordan-Elimination ohne Pivotisierung gebildete, endgültige erweiterte Matrix.

Mit der Funktion rref erhalten Sie eine zeilenreduzierte Staffelform für eine erweiterte Matrix. Diese Funktion erzeugt eine Liste der Pivot-Elemente und eine äquivalente Matrix in zeilenreduzierter Staffelform, sodass die Koeffizientenmatrix zu einer Diagonalmatrix reduziert wird.

Beispielsweise erzeugt die Funktion rref für die Matrix AAUG folgendes Ergebnis:





Die Ausgabe im zweiten Fenster oben erhalten Sie durch Aktivieren des Befehlszeileneditors (drücken Sie ▼). Das Ergebnis enthält die Pivot-Elemente 3, 1, 4, 1, 5 und 2 sowie eine reduzierte Diagonalmatrix.

Funktion SYST2MAT

Mit dieser Funktion wird ein lineares Gleichungssystem in die äquivalente erweiterte Matrix konvertiert. Das folgende Beispiel erhalten Sie über die Hilfefunktion des Taschenrechners:

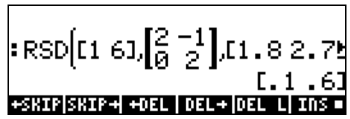
Das Ergebnis ist die dem Gleichungssystem entsprechende erweiterte Matrix:

$$X+Y = 0$$
$$X-Y = 2$$

Restfehler bei Lösungen linearer Gleichungssysteme (Funktion RSD)

Mit der Funktion RSD werden die ReSiDuen bzw. Restfehler bei der Lösung der Matrixgleichung $\mathbf{A}\cdot\mathbf{x}=\mathbf{b}$ berechnet, die ein System von n linearen Gleichungen mit n Unbekannten darstellt. Wir können die Lösung dieses Systems als Lösung der Matrixgleichung $f(\mathbf{x}) = \mathbf{b} - \mathbf{A}\cdot\mathbf{x} = 0$ betrachten. Angenommen, wir erzeugen mit einer numerischen Methode als erste Näherung die Lösung $\mathbf{x}(0)$. Wir berechnen $f(\mathbf{x}(0)) = \mathbf{b} - \mathbf{A}\cdot\mathbf{x}(0) = \mathbf{e} \neq 0$. $\mathbf{e}$ ist somit ein Vektor der Residuen der Funktion für den Vektor $\mathbf{x} = \mathbf{x}(0)$.

Als Argumente der Funktion RSD sind $\mathbf{b}$, $\mathbf{A}$ und $\mathbf{x}(0)$ erforderlich. Der zurückgegebene Vektor lautet $\mathbf{e} = \mathbf{b} - \mathbf{A}\cdot\mathbf{x}(0)$. Wenn beispielsweise $\mathbf{A}$ = `[[2,-1][0,2]]`, $\mathbf{x}(0)$ = `[1.8,2.7]` und $\mathbf{b}$ = `[1,6]`, können wir den Vektor der Residuen wie folgt ermitteln:



Das Ergebnis lautet **e** = **b** - **A**·**x**(0) = `[ 0.1 0.6 ]`.

Hinweis: Wenn wir die Korrektur der Werte von **x**(0) durch den Vektor Δ**x** = **x** – **x** (0) darstellen, können wir für Δ**x** eine neue Matrixgleichung **A**·Δ**x** = **e** erstellen. Durch das Ermitteln von Δ**x** finden wir mit **x** = **x**(0) + Δ**x** die tatsächliche Lösung des ursprünglichen Gleichungssystems.

Eigenwerte und Eigenvektoren

Für eine quadratische Matrix **A** können wir die Eigenwertgleichung **A**·**x** = λ·**x** erstellen, wobei die der Gleichung entsprechenden Werte von λ als <u>Eigenwerte von Matrix **A**</u> bezeichnet werden. Wir können für jeden Wert von λ in der Gleichung Werte von **x** ermitteln, die der Eigenwertgleichung entsprechen. Diese Werte von x werden als <u>Eigenvektoren von Matrix **A**</u> bezeichnet. Die Eigenwertgleichung kann auch als (**A** – λ·**I**)**x** = 0 geschrieben werden.

Diese Gleichung besitzt nur dann eine nicht triviale Lösung, wenn die Matrix (**A** - λ·**I**) singulär ist, d. h., wenn det(**A** - λ·**I**) = 0.

Die letzte Gleichung erzeugt eine algebraische Gleichung mit einem Polynom der Ordnung *n* für eine quadratische Matrix $\mathbf{A}_{n\times n}$. Die resultierende Gleichung wird als <u>charakteristisches Polynom</u> der Matrix **A** bezeichnet. Die Lösung des charakteristischen Polynoms ergibt die Eigenwerte der Matrix.

Der Taschenrechner enthält mehrere Funktionen, mit denen Sie Informationen über Eigenwerte und Eigenvektoren einer quadratischen Matrix erhalten. Einige dieser Funktionen befinden sich im Menü MATRICES/EIGEN, das mit ← MATRICES aktiviert wird.

Funktion PCAR

Mit der Funktion PCAR wird unter Verwendung der Werte der Variablen VX (eine für CAS reservierte Variable, die in der Regel gleich „X“ ist) ein charakteristisches Polynom einer quadratischen Matrix erzeugt. Geben Sie beispielsweise im ALG-Modus folgende Matrix ein, und ermitteln Sie mit PCAR die charakteristische Gleichung: `[[1,5,-3],[2,-1,4],[3,5,2]]`

Unter Verwendung der Variablen λ zur Darstellung der Eigenwerte ist dieses charakteristische Polynom als λ^3-2λ^2-22λ +21=0 zu interpretieren.

Funktion EGVL

Mit der Funktion EGVL (EiGenVaLues, Eigenwerte) werden die Eigenwerte einer quadratischen Matrix erzeugt. Die Eigenwerte der unten dargestellten Matrix werden z. B. im ALG-Modus mit der Funktion EGVL berechnet:

Eigenwerte λ = [-√10, √10].

Hinweis: In einigen Fällen können Sie möglicherweise keine „exakte“ Lösung für das charakteristische Polynom ermitteln und erhalten bei Verwendung der Funktion EGVL als Ergebnis eine leere Liste. Wenn dieser Fall eintritt, ändern Sie den Berechnungsmodus in CAS in den Näherungsmodus und wiederholen die Berechnung.

Beispielsweise wird bei der folgenden Übung im exakten Modus als Ergebnis eine leere Liste ausgegeben.

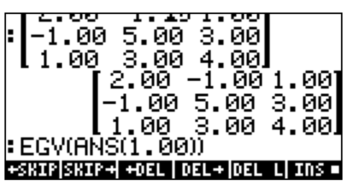
Ändern Sie den Modus in den Näherungsmodus, und wiederholen Sie die Eingabe. Sie erhalten folgende Eigenwerte:

`[(1,38;2,22), (1,38;-2,22), (-1,76;0)].`

Funktion EGV

Mit der Funktion EGV (EiGenwerte und EigenVektoren) werden die Eigenwerte und Eigenvektoren einer quadratischen Matrix erzeugt. Die Eigenvektoren werden als die Spalten einer Matrix zurückgegeben, während die entsprechenden Eigenwerte die Komponenten eines Vektors darstellen.

Beispielsweise werden die Eigenvektoren und Eigenwerte der unten aufgeführten Matrix im ALG-Modus durch Anwendung der Funktion EGV ermittelt:



In der Ergebnisliste werden die Eigenwerte als Spalten der Matrix angezeigt. Um die Eigenwerte anzuzeigen, können wir den Befehl GET(ANS(1),2) verwenden, d. h. das zweite Element in der Liste des vorherigen Ergebnisses abrufen.Die Eigenwerte lauten:

Gesamt:

λ_1 = 0,29; $\mathbf{x}_1$ = [1,00;0,79;–0,91]^T;
λ_2 = 3,16; $\mathbf{x}_2$ = [1,00;-0,51; 0,65]^T;
λ_3 = 7,54; $\mathbf{x}_1$ = [-0,03; 1,00; 0,84]^T.

Hinweis: Eine symmetrische Matrix erzeugt alle echten Eigenwerte, und ihre Eigenvektoren sind zueinander orthogonal. Für das gerade erläuterte Beispiel können Sie überprüfen, ob $\mathbf{x}_1 \bullet \mathbf{x}_2 = 0$, $\mathbf{x}_1 \bullet \mathbf{x}_3 = 0$ und $\mathbf{x}_2 \bullet \mathbf{x}_3 = 0$.

Funktion JORDAN

Die Funktion JORDAN ist für die Diagonalisierung oder Jordan-Zerlegung einer Matrix konzipiert. Mit der Funktion JORDAN werden im RPN-Modus für eine quadratische Matrix **A** vier Ausgaben erzeugt:

- Das Minimalpolynom von Matrix **A** (Ebene 4 des Stacks)
- Das charakteristische Polynom von Matrix **A** (Ebene 3 des Stacks)
- Eine Liste der jedem Eigenwert von Matrix **A** entsprechenden Eigenvektoren (Ebene 2 des Stacks)
- Ein Vektor mit den Eigenvektoren von Matrix **A** (Ebene 4 des Stacks)

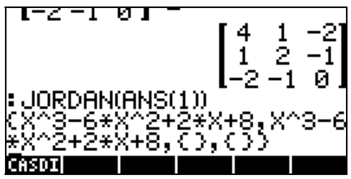
Führen Sie im RPN-Modus beispielsweise diese Übung aus:

```
[[4,1,-2],[1,2,-1],[-2,-1,0]]   JORDAN
```

Die Ausgabe lautet wie folgt:

4: 'X^3+-6*x^2+2*X+8'
3: 'X^3+-6*x^2+2*X+8'
2: { }
1: { }

Dieselbe Übung wird im ALG-Modus wie in den folgenden Bildschirmabbildungen dargestellt:



Funktion MAD

Obwohl diese Funktion nicht im Menü EIGEN zur Verfügung steht, stellt sie auch Informationen über die Eigenwerte einer Matrix bereit. Die Funktion MAD ist im Untermenü MATRICES OPERATIONS (⬑ MATRICES) verfügbar und zum Erzeugen der adjungierten Matrix einer Matrix konzipiert.

Im RPN-Modus erzeugt die Funktion MAD mehrere Eigenschaften einer quadratischen Matrix:

- die Determinante (Ebene 4 des Stacks)
- die formale Inverse (Ebene 3 des Stacks)
- die Matrixkoeffizienten des durch ($\mathbf{x}\cdot\mathbf{I}$-$\mathbf{A}$) $\cdot$p($\mathbf{x}$)=m($\mathbf{x}$)$\cdot\mathbf{I}$ definierten Polynoms p($\mathbf{x}$) (Ebene 2 des Stacks)
- das charakteristische Polynom der Matrix (Ebene 1 des Stacks)

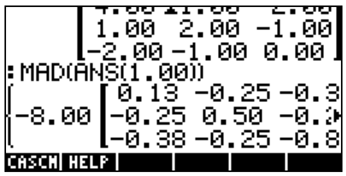
Beachten Sie, dass die Form der Gleichung ($\mathbf{x}\cdot\mathbf{I}$-$\mathbf{A}$)$\cdot$p($\mathbf{x}$)=m($\mathbf{x}$)$\cdot\mathbf{I}$ der Eigenwertgleichung $\mathbf{A}\cdot\mathbf{x} = \lambda\cdot\mathbf{x}$ entspricht.

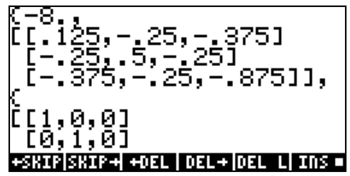
Geben Sie als Übungsbeispiel im RPN-Modus Folgendes ein:

```
[[4,1,-2] [1,2,-1][-2,-1,0]] MAD
```

Das Ergebnis lautet:
4: -8.
3: [[0,13 –0,25 –0,38][-0,25 0,50 –0,25][-0,38 –0,25 –0,88]]
2: {[[1 0 0][0 1 0][0 0 1]] [[-2 1 –2][1 –4 –1][-2 –1 –6] [[-1 2 3][2 –4 2][3 2 7]]}
1: 'X^3+-6*x^2+2*X+8'

Dasselbe Beispiel wird im ALG-Modus wie folgt angezeigt:





Matrixfaktorisierung

Die Faktorisierung bzw. Zerlegung einer Matrix besteht im Ermitteln von Matrizen, die durch Multiplikation die Ausgangsmatrix ergeben. Wir stellen die Matrixzerlegung durch die Funktionen im Matrixmenü FACT dar. Dieses Menü wird mit ← MATRICES aufgerufen.

Die Funktionen dieses Menüs lauten: LQ, LU, QR,SCHUR, SVD und SVL.

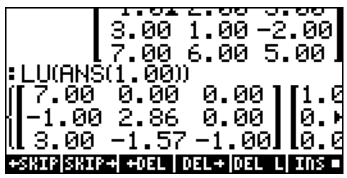
Die Funktion LU

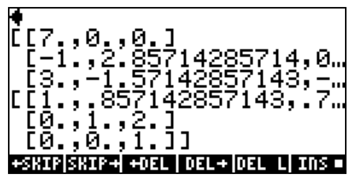
Der Eingabewert für die Funktion LU ist eine quadratische Matrix **A.** Die Ausgabe ist eine untere Dreiecksmatrix **L**, eine obere Dreiecksmatrix **U** und eine Permutationsmatrix **P** auf Ebene 3, 2 bzw. 1 des Stacks. Die Ergebnisse

für **L**, **U** und **P** stimmen mit der Gleichung **P**·**A** = **L**·**U** überein. Beim Aufruf der Funktion LU führt der Taschenrechner mithilfe einer Teilpivotisierung eine LU-Zerlegung von **A** nach dem Crout-Algorithmus durch.
Beispielsweise ergibt die folgende Eingabe im RPN-Modus:
`[[-1,2,5][3,1,-2][7,6,5]] LU`
die Werte:

3:[[7 0 0][-1 2,86 0][3 –1,57 –1]]
2:[[1 0,86 0,71][0 1 2][0 0 1]]
1:[[0 0 1][1 0 0][0 1 0]]

Im ALG-Modus wird dasselbe Beispiel wie folgt angezeigt:





Orthogonalmatrizen und Singulärwertzerlegung

Eine quadratische Matrix ist orthogonal, wenn ihre Spalten Einheitsvektoren darstellen, die zueinander orthogonal sind. Die Matrix **U** = [$\mathbf{v}_1$ $\mathbf{v}_2$ … $\mathbf{v}_n$] mit den Spaltenvektoren $\mathbf{v}_i$, i = 1, 2, …, n und $\mathbf{v}_i \bullet \mathbf{v}_j = \delta_{ij}$, wobei δ_{ij} die Kronecker-Deltafunktion darstellt, ist **U** daher eine Orthogonalmatrix. Aus diesen Bedingungen folgt außerdem, dass $\mathbf{U} \cdot \mathbf{U}^T = \mathbf{I}$.

Die Singulärwertzerlegung (SVD) einer rechteckigen Matrix $\mathbf{A}_{m\times n}$ erfolgt daher durch Bestimmung der Matrizen **U**, **S** und **V**, sodass $\mathbf{A}_{m\times n} = \mathbf{U}_{m\times m} \cdot \mathbf{S}_{m\times n} \cdot \mathbf{V}^T_{n\times n}$, wobei **U** und **V** Orthogonalmatrizen sind und **S** eine Diagonalmatrix ist. Die diagonalen Elemente von **S** werden als Singulärwerte von **A** bezeichnet und sind in der Regel so angeordnet, dass für *i = 1, 2, …, n-1* gilt, dass $s_i \geq s_{i+1}$. Die Spalten [$\mathbf{u}_j$] von **U** und [$\mathbf{v}_j$] von **V** sind die entsprechenden Singulärvektoren.

Funktion SVD

Im RPN-Modus ist der Eingabewert für die Funktion SVD (Singular Value Decomposition, Singulärwertzerlegung) eine Matrix $\mathbf{A}_{n\times m}$. Die Funktion gibt auf den Ebenen 3, 2 bzw. 1 des Stacks die Matrizen $\mathbf{U}_{n\times n}$, $\mathbf{V}_{m\times m}$ und einen

Vektor **s** zurück. Die Dimension des Vektors **s** ist gleich dem geringsten Wert von n bzw. m. Die Matrizen **U** und **V** entsprechen der bereits erläuterten Definition für die Singulärwertzerlegung, während der Vektor **s** die Hauptdiagonale der bereits verwendeten Matrix **S** darstellt.

Beispielsweise ergibt die folgende Eingabe im RPN-Modus:
`[[5,4,-1],[2,-3,5],[7,2,8]] SVD`

3: [[-0,27 0,81 –0,53][-0,37 –0,59 –0,72][-0,89 3,09E-3 0,46]]
2: [[-0,68 –0,14 –0,72][0,42 0,73 –0,54][-0,60 0,67 0,44]]
1: [12,15 6,88 1,42]

Funktion SVL

Die Funktion SVL (Singular VaLues, Singulärwerte) gibt die Singulärwerte einer Matrix $\mathbf{A}_{n\times m}$ als Vektor **s** zurück, dessen Dimension gleich dem geringsten Wert von n bzw. m ist. Beispielsweise ergibt
`[[5,4,-1],[2,-3,5],[7,2,8]] SVL` im RPN-Modus
[12,15 6,88 1,42].

Funktion SCHUR

Im RPN-Modus erzeugt die Funktion SCHUR die *Schur-Zerlegung* einer quadratischen Matrix **A** und ergibt die Matrizen **Q** und **T** auf Ebene 2 bzw. 1 des Stacks, sodass $\mathbf{A} = \mathbf{Q}\cdot\mathbf{T}\cdot\mathbf{Q}^T$, wobei **Q** eine Orthogonalmatrix und **T** eine Dreiecksmatrix ist. Beispielsweise ergibt

`[[2,3,-1][5,4,-2][7,5,4]] SCHUR`

im RPN-Modus folgende Ausgabe:
2: [[0,66 –0,29 –0,70][-0,73 –0,01 –0,68][-0,19 –0,96 0,21]]
1: [[-1,03 1,02 3,86][0 5,52 8,23][0 –1,82 5,52]]

Funktion LQ

Die Funktion LQ erzeugt die *LQ-Faktorisierung* einer Matrix $\mathbf{A}_{n\times m}$ und gibt Ebene 3, 2 bzw. 1 des Stacks eine untere Trapezmatrix $\mathbf{L}_{n\times m}$, eine Orthogonalmatrix $\mathbf{Q}_{m\times m}$ und eine Permutationsmatrix $\mathbf{P}_{n\times n}$ zurück. Für die Matrizen **A**, **L**, **Q** und **P** gilt $\mathbf{P}\cdot\mathbf{A} = \mathbf{L}\cdot\mathbf{Q}$. (Eine aus einer n×m-Matrix gebildete Trapezmatrix entspricht einer aus einer n×n-Matrix gebildeten Dreiecksmatrix.) Beispielsweise erzeugt

`[[ 1, -2, 1][ 2, 1, -2][ 5, -2, 1]] LQ`

die Werte
3: [[-5,48 0 0][-1,10 –2,79 0][-1,83 1,43 0,78]]
2: [[-0,27 0,81 –0,18][-0,36 –0,50 –0,79][-0,20 –0,78 –0,59]]
1: [[0 0 1][0 1 0][1 0 0]]

Funktion QR

Die Funktion QR erzeugt im RPN-Modus die *QR-Faktorisierung* einer Matrix $\mathbf{A}_{n\times m}$ und gibt auf Ebene 3, 2 bzw. 1 des Stacks eine Orthogonalmatrix $\mathbf{Q}_{n\times n}$, eine obere Trapezmatrix $\mathbf{R}_{n\times m}$ und eine Permutationsmatrix $\mathbf{P}_{m\times m}$ zurück. Für die Matrizen **A**, **P**, **Q** und **R** gilt $\mathbf{A}\cdot\mathbf{P} = \mathbf{Q}\cdot\mathbf{R}$. Beispielsweise erzeugt

```
[[ 1,-2,1][ 2,1,-2][ 5,-2,1]] QR
```

die Werte
3: [[-0,18 0,39 0,90][-0,37 –0,88 0,30][-0,91 0,28 –0,30]]
2: [[-5,48 –0,37 1,83][0 2,42 –2,20][0 0 –0,90]]
1: [[1 0 0][0 0 1][0 1 0]]

Hinweis: Über die Hilfefunktion des Taschenrechners erhalten Sie Beispiele und Definitionen für sämtliche Funktionen dieses Menüs. Führen Sie diese Übungen im ALG-Modus aus, um die Ergebnisse in diesem Modus anzuzeigen.

Quadratische Formen einer Matrix

Die <u>quadratische Form</u> einer quadratischen Matrix **A** ist ein aus $\mathbf{x}\cdot\mathbf{A}\cdot\mathbf{x}^T$ gebildetes Polynom. Für **A** = [[2,1,–1][5,4,2][3,5,–1]] und **x** = $[X\ Y\ Z]^T$ wird z. B. die entsprechende quadratische Form wie folgt berechnet:

$$\mathbf{x}\cdot\mathbf{A}\cdot\mathbf{x}^T = \begin{bmatrix} X & Y & Z \end{bmatrix} \cdot \begin{bmatrix} 2 & 1 & -1 \\ 5 & 4 & 2 \\ 3 & 5 & -1 \end{bmatrix} \cdot \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

$$= \begin{bmatrix} X & Y & Z \end{bmatrix} \cdot \begin{bmatrix} 2X+Y-Z \\ 5X+4Y+2Z \\ 3X+5Y-Z \end{bmatrix}$$

Endergebnis: $\mathbf{x}\cdot\mathbf{A}\cdot\mathbf{x}^T = 2X^2+4Y^2-Z^2+6XY+2XZ+7ZY$

Das Menü QUADF

Der Taschenrechner HP 49 G enthält das Menü QUADF für Operationen mit QUADratischen Formen. Das Menü QUADF wird mit ↰ MATRICES aufgerufen.

Dieses Menü enthält die Funktionen AXQ, CHOLESKY, GAUSS, QXA und SYLVESTER.

Funktion AXQ

Die Funktion AXQ erzeugt im RPN-Modus unter Verwendung von *n* Variablen eines Vektors auf Ebene 1 des Stacks die einer Matrix $\mathbf{A}_{n\times n}$ auf Ebene 2 des Stacks entsprechende quadratische Form. Die Funktion gibt die quadratische Form auf Ebene 1 des Stacks und den Vektor der Variablen auf Ebene 1 des Stacks zurück. Beispielsweise ergibt

```
[[2,1,-1],[5,4,2],[3,5,-1]] ENTER
['X','Y','Z'] ENTER AXQ
```

folgende Werte:
2: '2*X^2+(6*Y+2*Z)*X+4*Y^2+7*Z*y-Z^2'
1: ['X' 'Y' 'Z']

Funktion QXA

Die Argumente für die Funktion QXA sind eine quadratische Form auf Ebene 2 des Stacks und ein Vektor von Variablen auf Ebene 1 des Stacks. Die Funktion gibt auf Ebene 2 des Stacks die quadratische Matrix **A** zurück, von der die quadratische Form abgeleitet wurde, und auf Ebene 1 des Stacks die Liste der Variablen. Beispielsweise ergibt

```
'X^2+Y^2-Z^2+4*X*Y-16*X*Z' ENTER
['X','Y','Z'] ENTER QXA
```

folgende Werte:

2: [[1 2 –8][2 1 0][-8 0 –1]]
1: ['X' 'Y' 'Z']

Diagonale Darstellung einer quadratischen Form

Für eine symmetrische quadratische Matrix **A** kann die Matrix **A** „diagonalisiert" werden, indem eine Orthogonalmatrix **P** ermittelt wird, für die gilt: $\mathbf{P}^T \cdot \mathbf{A} \cdot \mathbf{P} = \mathbf{D}$, wobei **D** eine Diagonalmatrix ist. Wenn $Q = \mathbf{x} \cdot \mathbf{A} \cdot \mathbf{x}^T$ eine auf **A** basierende quadratische Form ist, kann die quadratische Form Q so dargestellt werden, dass sie mit $Q = \mathbf{x} \cdot \mathbf{A} \cdot \mathbf{x}^T = (\mathbf{P} \cdot \mathbf{y}) \cdot \mathbf{A} \cdot (\mathbf{P} \cdot \mathbf{y})^T = \mathbf{y} \cdot (\mathbf{P}^T \cdot \mathbf{A} \cdot \mathbf{P}) \cdot \mathbf{y}^T = \mathbf{y} \cdot \mathbf{D} \cdot \mathbf{y}^T$ nur quadratische Ausdrücke einer Variablen **y** enthält, sodass $\mathbf{x} = \mathbf{P} \cdot \mathbf{y}$.

Funktion SYLVESTER

Die Funktion SYLVESTER nimmt eine symmetrische quadratische Matrix **A** als Argument an und gibt einen Vektor zurück, der die diagonalen Größen einer Diagonalmatrix **D** enthält, sowie eine Matrix **P**, sodass $\mathbf{P}^T \cdot \mathbf{A} \cdot \mathbf{P} = \mathbf{D}$.
Beispielsweise ergibt

```
[[2,1,-1],[1,4,2],[-1,2,-1]] SYLVESTER
```

die Werte
2: [1/2 2/7 -23/7]
1: [[2 1 –1][0 7/2 5/2][0 0 1]]

Funktion GAUSS

Die Funktion GAUSS gibt die diagonale Darstellung einer quadratischen Form $Q = \mathbf{x} \cdot \mathbf{A} \cdot \mathbf{x}^T$ zurück und nimmt als Argumente die quadratische Form auf Ebene 2 des Stacks und den Vektor der Variablen auf Ebene 1 des Stacks an. Der Aufruf dieser Funktion führt zu folgenden Ergebnissen:

- Ein Feld von Koeffizienten, die die diagonalen Größen von **D** darstellen (Ebene 4 des Stacks)
- Eine Matrix **P**, sodass $\mathbf{A} = \mathbf{P}^T \cdot \mathbf{D} \cdot \mathbf{P}$ (Ebene 3 des Stacks)
- Die diagonalisierte quadratische Form (Ebene 2 des Stacks)
- Die Liste der Variablen (Ebene 1 des Stacks)

Beispielsweise erzeugt

```
'X^2+Y^2-Z^2+4*X*Y-16*X*Z' ENTER
['X','Y','Z'] ENTER GAUSS
```

folgende Werte:
4: [1 –0,333 20,333]

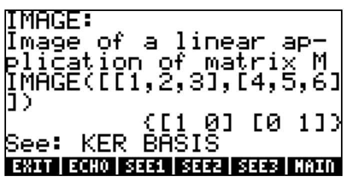
3: [[1 2 –8][0 –3 16][0 0 1]]
2: '61/3*Z^2+ -1/3*(16*Z+-3*Y)^2+(-8*z+2*Y+X)^2'
1: ['X' 'Y' 'Z']

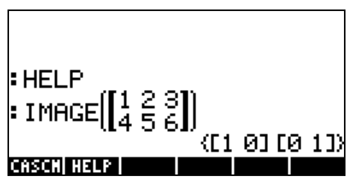
LINEAR APPLICATIONS

Das Menü LINEAR APPLICATIONS wird über MATRICES aufgerufen.

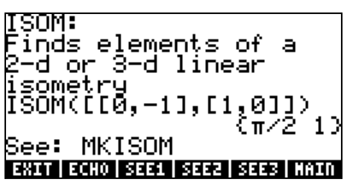
Unten sind die Informationen über die Funktionen dieses Menüs dargestellt, die Sie mit der Hilfefunktion des Taschenrechners aufrufen können. Die Abbildungen stellen den entsprechenden Eintrag der Hilfefunktion und die zugehörigen Beispiele dar.

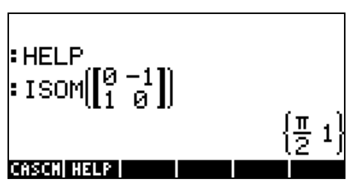
Funktion IMAGE



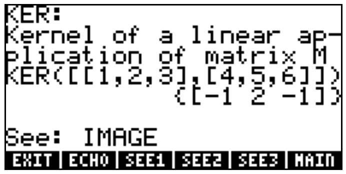


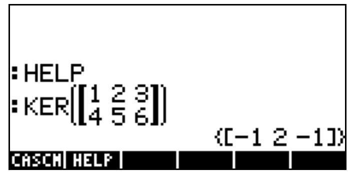
Funktion ISOM



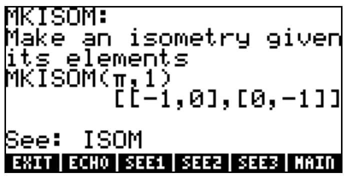


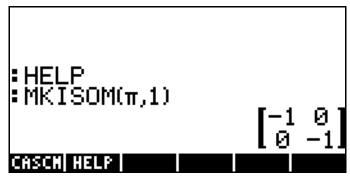
Funktion KER





Funktion MKISOM



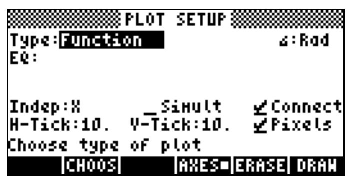


Kapitel 12
Grafik

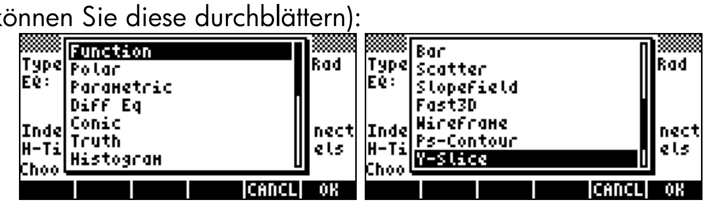
In diesem Kapitel werden einige der Grafikfunktionen des Taschenrechners vorgestellt. Wir stellen Grafiken von Funktionen kartesischer Koordinaten und Polarkoordinaten vor, von parametrischen Diagrammen, Punktdiagrammen, Balkendiagrammen und einer Vielzahl von dreidimensionalen Grafiken.

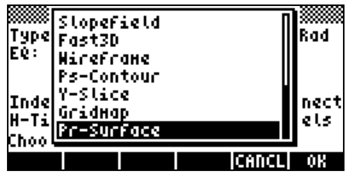
Grafikoptionen des Taschenrechners

Über die Tastenkombination ↰ 2D/3D (F4) gelangen Sie zur Liste der im Taschenrechner verfügbaren Grafikformate. Bitte beachten Sie, dass Sie diese beiden Tasten gleichzeitig drücken müssen, um eine Grafikfunktion einzuschalten, wenn Sie im RPN-Modus arbeiten. Wenn Sie die 2D/3D-Funktion eingeschaltet haben, erscheint auf dem Taschenrechnerdisplay das Fenster PLOT SETUP, darin befindet sich auch das Feld TYPE (siehe Abb. unten).



Gleich neben dem Feld TYPE erscheint normalerweise die Option *Function* hervorgehoben. Das ist der voreingestellte Standard-Grafiktyp des Taschenrechners. Um die Liste der verfügbaren Grafiktypen anzuzeigen, drücken Sie den Softkey CHOOS. Dann erscheint ein Dropdown-Menü mit folgenden Optionen (mit den Pfeiltasten „nach oben" bzw. „nach unten" können Sie diese durchblättern):





Diese Grafikoptionen werden im Folgenden kurz beschrieben.

Function: für Gleichungen der Form y = f(x) in ebenen kartesischen Koordinaten.
Polar: für Gleichungen der Form r = f(θ) in Polarkoordinaten in der Ebene.
Parametric: zur Darstellung von Gleichungen der Form x = x(t), y = y(t) in der Ebene.
Diff Eq: zur Darstellung numerischer Lösungen linearer Differentialgleichungen.
Conic: zur Darstellung von Kegelschnitt-Gleichungen (Kreise, Ellipsen, Hyperbeln und Parabeln).
Truth: zur Darstellung von Ungleichungen in der Ebene.
Histogram: zur Darstellung von Frequenzhistogrammen (Statistikanwendungen).
Bar: zur Darstellung einfacher Balkendiagramme.
Scatter: zur Darstellung von Punktdiagrammen einzelner Datensätze (Statistikanwendungen).
Slopefield: zur Darstellung von Steigungsfeldern einer Funktion f(x,y) = 0.
Fast3D: zur Darstellung von Kurvenoberflächen im Raum.
Wireframe: zur Darstellung von Kurvenoberflächen im Raum, dargestellt als Drahtgitter.
Ps-Contour: zur Darstellung von Oberflächenkonturdiagrammen.
Y- Slice: zur Darstellung der Schnittansicht einer Funktion f(x,y).
Gridmap: zur Darstellung realer und imaginärer Teilverläufe einer komplexen Funktion.
Pr-Surface: für parametrische Oberflächen aus x = x(u,v), y = y(u,v), z = z(u,v).

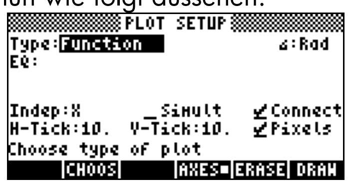
Darstellung eines Ausdrucks der Form *y = f(x)*

In diesem Abschnitt stellen wir ein Beispiel für die Darstellung einer Funktion der Form *y = f(x)* vor. Um mit der Darstellung fortzufahren, muss zunächst die Variable x gelöscht werden, wenn diese im aktuellen Verzeichnis definiert ist

(x soll die unabhängige Variable in der Taschenrechnerfunktion PLOT sein und sollte deshalb nicht vorbelegt sein). Erstellen Sie ein Unterverzeichnis mit der Bezeichnung 'TPLOT' (für Testplot = Testdarstellung) oder einem anderen aussagekräftigen Namen, um folgende Übung durchzuführen.
Als Beispiel stellen wir nun die folgende Funktion dar:

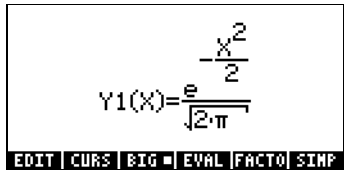
$$f(x) = \frac{1}{\sqrt{2\pi}} \exp(-\frac{x^2}{2})$$

- Gehen Sie zunächst in das Menü PLOT SETUP durch Drücken von ← 2D/3D . Stellen Sie sicher, dass die Option „Function" im Feld TYPE und 'X' als unabhängige Variable (INDEP) gewählt ist. Drücken Sie NXT OK, um zur normalen Anzeige zurückzukehren. Das Fenster PLOT SET UP sollte nun wie folgt aussehen:



- **Hinweis:** Sie werden feststellen, dass eine neue Variable mit der Bezeichnung PPAR bei den Softkeys erscheint. Diese Bezeichnung steht für Plot PARameters (= Parameter darstellen). Um deren Inhalt anzuzeigen, drücken Sie → PPAR . Eine detaillierte Erklärung des Inhalts von PPAR finden Sie weiter hinten in diesem Kapitel. Drücken Sie ◄, um diese Zeile aus dem Stack zu holen.

- Durch Drücken von ← Y= gelangen Sie in das Menü PLOT (gleichzeitig drücken, wenn im RPN-Modus). Drücken Sie ADD, um zum Equation Writer zu gelangen. Dort werden Sie aufgefordert, die rechte Seite einer Gleichung Y1(x) = ▪ auszufüllen. Geben Sie die darzustellende Funktion ein, sodass der Equation Writer Folgendes anzeigt:



- Drücken Sie ENTER, um zum Fenster PLOT SETUP zurückzukehren. Der Ausdruck 'Y1(X) = EXP(-X^2/2)/√(2*π)' wird hervorgehoben. Drücken Sie NXT OK, um zur normalen Anzeige zurückzukehren.

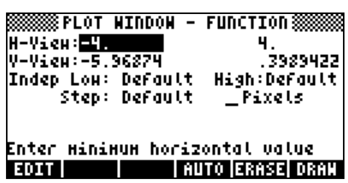
Hinweis: Unter Softkeys werden zwei neue Variablen angezeigt, EQ und Y1. Um den Inhalt von EQ anzuzeigen, drücken Sie ↱ EQ. Der Inhalt von EQ ist einfach die Funktionsbezeichnung 'Y1(X)'. Die Variable EQ wird vom Taschenrechner dazu verwendet, die darzustellende/n Gleichung/en zu speichern.

Um den Inhalt von Y1 anzuzeigen, drücken Sie ↱ Y1. Sie erhalten die Funktion Y1(X) festgelegt als das Programm:

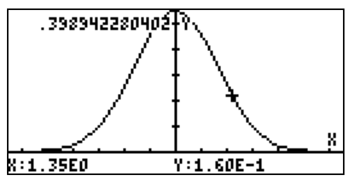
```
<< →X 'EXP(-X^2/2)/√(2*π)' >>.
```

Drücken Sie ◄ zweimal, um den Inhalt aus dem Stack zu entfernen.

- Durch Drücken von ↰ WIN gelangen Sie in das Menü PLOT WINDOW (gleichzeitig drücken, wenn im RPN-Modus). Verwenden Sie einen Bereich von –4 bis 4 für H-VIEW und drücken Sie anschließend AUTO, um V-VIEW automatisch zu erzeugen. Das Fenster PLOT WINDOW sieht folgendermaßen aus:



- Graph darstellen: ERASE DRAW (Warten Sie, bis der Taschenrechner den Graph fertig gestellt hat.)
- Bezeichnungen anzeigen: EDIT NXT LABEL MENU
- Erstes Grafik-Menü wiederherstellen: NXT NXT PICT
- Kurvenverlauf verfolgen: TRACE (X,Y) . Mit den Pfeiltasten „nach links" bzw. „nach rechts" (◄ ►) können Sie sich auf der Kurve hin und her bewegen. Die Koordinaten der Punkte, die sie verfolgen, werden unten auf dem Display angezeigt. Prüfen Sie, dass für x = 1,05; y = 0,231 gilt. Prüfen Sie außerdem, dass für x = -1,48; y = 0,134 gilt. Die folgende Abbildung stellt die Kurve im Verfolgungsmodus dar:



- Um das Menü wiederherzustellen und in das Fenster PLOT WINDOW zurückzukehren, drücken Sie NXT CANCL und anschließend NXT OK.

Hilfreiche Funktionen für Funktionsdarstellungen

Zur Besprechung dieser PLOT-Optionen ändern wir die Funktion ab, um ihr einen wirklichen Beginn aufzuzwingen. (Da die aktuelle Kurve vollständig oberhalb der x-Achse liegt, hat sie keinen wirklichen Anfang.) Drücken Sie ↱ Y1, um den Inhalt der Funktion Y1 im Stack aufzulisten:
<< →X 'EXP(-X^2/2)/ √(2*π) ' >>.
Zur Bearbeitung dieses Ausdrucks verwenden Sie:

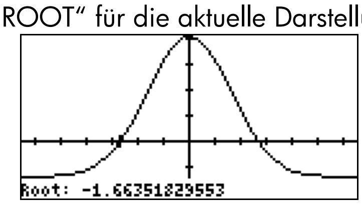
▼	Startet den Zeileneditor
↱ ▼	Bewegt den Cursor an das Zeilenende
◄ ◄ ◄ − 0 . 1	Ändert den Ausdruck
ENTER	Zurück zur Taschenrechnerdisplay.

Anschließend speichern Sie den geänderten Ausdruck in der Variablen y mithilfe von ⬅ Y1 im RPN-Modus oder über ⬅ ANS STO▶ Y1 im ALG-Modus.

Die darzustellende Funktion ist jetzt: $f(x) = \frac{1}{\sqrt{2\pi}} \exp(-\frac{x^2}{2}) - 0.1$

Durch Drücken von ⬅ WIN gelangen Sie in das Menü PLOT WINDOW (gleichzeitig drücken, wenn im RPN-Modus). Behalten Sie einen Bereich von –4 bis 4 für H-VIEW bei und drücken Sie anschließend ▼ AUTO, um V-VIEW zu erzeugen. Zur Darstellung der Kurve drücken Sie ERASE DRAW

- Wenn der Graph dargestellt ist, drücken Sie FCN, um in das *function*-Menü zu gelangen. In diesem Menü erhalten Sie zusätzliche Informationen über die Darstellung, z. B. Schnittpunkte mit der x-Achse, Kurvenbeginn, Steigung von Stützgeraden, Bereich unter einer Kurve, usw.
- Wenn Sie z. B. den Kurvenbeginn auf der linken Seite der Kurve suchen, bewegen Sie den Cursor in die Nähe dieses Punkts und drücken Sie ROOT. Sie erhalten das Ergebnis: ROOT: -1,6635…. Drücken Sie NXT, um zum Menü zurückzukehren. Dies ist das Ergebnis von „ROOT" für die aktuelle Darstellung:



- Wenn Sie den Cursor zur rechten Seite der Kurve hin bewegen, indem Sie die rechte Pfeiltaste (▶) und anschließend ROOT drücken, dann ist das Ergebnis für ROOT: 1,6635... Der Taschenrechner hat vor der Anzeige des Kurvenbeginns angezeigt, dass das Ergebnis über *SIGN REVERSAL (Vorzeichenumkehr)* gefunden wurde. Drücken Sie NXT, um zum Menü zurückzukehren.
- Wenn Sie ISECT drücken, sehen Sie den Schnittpunkt der Kurve mit der x-Achse, welcher im Wesentlichen den Kurvenbeginn markiert.

Platzieren Sie den Cursor genau auf den Kurvenbeginn und drücken Sie ISECT. Sie werden dieselbe Meldung wie zuvor erhalten, und zwar *SIGN REVERSAL (Vorzeichenumkehr)*, bevor das Ergebnis für I-SECT angezeigt wird: 1,6635…. Die Funktion ISECT ist dafür vorgesehen, den Schnittpunkt zweier beliebiger Kurven zu ermitteln, der dem Cursor am nähesten ist. In diesem Fall, in dem es lediglich um eine Kurve geht, nämlich Y1(X), ist der gesuchte Schnittpunkt f(x) der mit der x-Achse. Der Cursor muss jedoch direkt am Kurvenanfang platziert werden, um dasselbe Ergebnis zu erzielen. Drücken Sie NXT, um zum Menü zurückzukehren.

- Platzieren Sie den Cursor auf einen beliebigen Punkt der Kurve und drücken Sie SLOPE. Sie erhalten dann den Wert für die Steigung an diesem Punkt. Zum Beispiel am negativen Anfangspunkt, SLOPE (Steigung): 0,16670…. Drücken Sie NXT, um zum Menü zurückzukehren.
- Um den höchsten Punkt einer Kurve zu ermitteln, positionieren Sie den Cursor in die Nähe des Scheitelpunkts und drücken Sie EXTR Das Ergebnis ist EXTRM: 0.. Drücken Sie NXT, um zum Menü zurückzukehren.
- Weitere verfügbare Softkeys im ersten Menü sind AREA zur Berechnung der Fläche unterhalb der Kurve und SHADE zur Schattierung der Fläche unterhalb der Kurve. Drücken Sie NXT, um weitere Optionen anzuzeigen. Das zweite Menü enthält einen Softkey mit der Bezeichnung VIEW, der die dargestellte Gleichung einige Sekunden lang aufblinken lässt. Drücken Sie VIEW. Alternativ können Sie auch den Softkey NEXQ (NEXt eQuation = nächste Gleichung) drücken, um den Namen der Funktion Y1(x) anzuzeigen. Drücken Sie NXT, um zum Menü zurückzukehren.
- Über den Softkey F(X) erhalten Sie abhängig von der Cursorposition den Wert für f(x). Platzieren Sie den Cursor an eine beliebige Stelle auf der Kurve und drücken Sie F(X). Der Wert wird in der unteren linken Ecke des Displays angezeigt. Drücken Sie NXT, um zum Menü zurückzukehren.
- Positionieren Sie den Cursor an einen beliebigen Punkt der Bewegungsbahn und drücken Sie TANL, um die Gleichung der Tangente zu dieser Kurve an diesem Punkt zu erhalten. Die Gleichung

wird in der unteren linken Ecke des Displays angezeigt. Drücken Sie NXT, um zum Menü zurückzukehren.

- Wenn Sie F' drücken, stellt der Taschenrechner die abgeleitete Funktion f'(x) = df/dx sowie auch die ursprüngliche Funktion f(x) dar. Beachten Sie, dass sich die beiden Kurven an zwei Punkten schneiden. Bewegen Sie den Cursor in die Nähe des linken Schnittpunkts und drücken Sie FCN und ISECT, um den Wert für I-SECT zu erhalten: (-0,6834…;0,21585). Drücken Sie NXT, um zum Menü zurückzukehren.
- Um das FCN-Menü zu verlassen, drücken Sie PICT (oder NXT PICT).
- Drücken Sie CANCL, um wieder in das Fenster PLOT WINDOW zu gelangen. Drücken Sie anschließend NXT OK, um zur normalen Taschenrechnerdisplay zurückzukehren.

Hinweis: Der Stack zeigt alle ausgeführten Grafik-Operationen entsprechend bezeichnet an.

- Gehen Sie durch Drücken von ← Y= in das Menü PLOT (gleichzeitig drücken, wenn im RPN-Modus). Beachten Sie, dass die hervorgehobenen Felder im PLOT-Menü nun die Ableitungen von Y1(X) enthalten. Drücken Sie NXT OK, um zur normalen Anzeige zurückzukehren.
- Drücken Sie → EQ, um den Inhalt von EQ zu prüfen. Sie werden feststellen, dass darin an Stelle eines Ausdrucks eine Liste enthalten ist. Die Liste enthält als Elemente Ausdrücke für die Ableitung von Y1(X) und Y1(X) selbst. Ursprünglich enthielt EQ nur Y1(x). Nachdem wir F' unter FCN gedrückt haben, hat der Taschenrechner automatisch die Ableitung von Y1(x) zur Gleichungsliste in EQ hinzugefügt.

Eine Grafik zur späteren Verwendung speichern

Wenn Sie Ihre Grafik in einer Variablen speichern möchten, rufen Sie das Menü PICTURE auf, indem Sie auf ◄ drücken. Drücken Sie anschließend EDIT NXT NXT PICT→. Damit wird das aktuelle Bild als Grafikobjekt erfasst. Um zum Stack zurückzukehren, drücken Sie PICT CANCL.

In Ebene 1 des Stacks sehen Sie ein Grafikobjekt, dasmit `Graphic 131 x 64` bezeichnet ist. Dieses kann unter einem beliebigen Variablennamen gespeichert werden, z. B. PIC1.

Um die Abbildung erneut anzuzeigen, rufen Sie den Inhalt von PIC1 aus dem Stack ab. Der Stack enthält folgende Zeile: `Graphic 131 × 64`. Um die Grafik anzuzeigen, drücken Sie auf ◄, um das Fenster PICTURE aufzurufen.

Löschen Sie das aktuelle Bild, EDIT NXT ERASE.

Bewegen Sie den Cursor durch Drücken der Tasten ◄ und ▲ in die linke obere Ecke des Displays.

Um die Abbildung in der Ebene 1 des Stacks anzuzeigen, drücken Sie NXT REPL .

Um zur normalen Taschenrechnerfunktion zurückzukehren, drücken Sie PICT CANCL.

Hinweis: Um keine Druckflächen zu verschwenden, werden die aus den Anweisungen zum Erstellen von Grafiken resultierenden Grafiken nicht abgebildet. Der Benutzer kann diese jedoch zu Übungszwecken selbst erstellen.

Grafiken transzendenter Funktionen

In diesem Abschnitt werden wir anhand einiger Grafikfunktionen des Taschenrechners das typische Verhalten natürlicher Logarithmus-, Exponential-, Winkel- und Hyperbelfunktionen aufzeigen. In diesem Kapitel werden keine Grafiken angezeigt, Sie können diese jedoch auf Ihrem Taschenrechner sehen.

Grafik für ln(X)

Wenn Sie im RPN-Modus sind, drücken Sie gleichzeitig die linke Umschalttaste ↰ und die Taste 2D/3D (F4), um das Fenster PLOT SETUP anzuzeigen. Das Feld mit der Bezeichnung **`Type`** ist dann hervorgehoben. Wenn die Option `Function` noch nicht gewählt ist, drücken Sie den Softkey

CHOOS und wählen mit den Auf- und Abwärtstasten die Option `Function` und drücken Sie anschließend OK, um die Auswahl abzuschließen. Stellen Sie sicher, dass das Feld Indep: die Variable 'X' enthält. Wenn das nicht der Fall ist, drücken Sie die Pfeiltaste „nach unten" zweimal, bis das Feld `Indep` hervorgehoben ist. Drücken Sie anschließend den Softkey EDIT und ändern Sie den Wert der unabhängigen Variablen in 'X'. Drücken Sie anschließend OK. Drücken Sie NXT OK, um zur normalen Anzeige zurückzukehren.

Ändern Sie als Nächstes die Größe des PLOT-Fensters. Wenn Sie im RPN-Modus sind, drücken Sie gleichzeitig die linke Umschalttaste ← und die Taste Y= (F1), um das Fenster PLOT-FUNCTION anzuzeigen. Wenn in diesem Fenster eine Gleichung hervorgehoben ist, drücken Sie DEL wie erforderlich, um das Fenster vollständig zu leeren. Wenn das Fenster PLOT-FUNCTION leer ist, erscheint folgende Aufforderung: **`No Equ., Press ADD`**. Drücken Sie den Softkey ADD . Dadurch startet der Equation Writer mit dem Ausdruck Y1(X)=◄ . Geben Sie LN(X) ein. Drücken Sie ENTER, um wieder in das Fenster PLOT-FUNCTION zu gelangen. Drücken Sie NXT OK, um zur normalen Anzeige zurückzukehren.

Wenn Sie im RPN-Modus sind, drücken Sie gleichzeitig die linke Umschalttaste ← und die Taste WIN (F2), um das Fenster PLOT WINDOW - FUNCTION anzuzeigen. Normalerweise wird dann im Display der horizontale (H-View) und vertikale (`V-View`) Bereich angezeigt. H-View: -6.5 6.5, V-View: -3.1 3.2

Dies sind die jeweiligen voreingestellten Standardwerte für die x- und y-Bereiche des aktuellen Fensters für die Grafikanzeige. Ändern Sie den Wert für H-VIEW folgendermaßen: H-View: -1 10 und zwar über 1 +/- OK 1 0 OK. Drücken Sie anschließend den Softkey AUTO, um den Taschenrechner den entsprechenden Vertikalbereich ermitteln zu lassen. Nach einigen Sekunden wird dieser Bereich im Fenster PLOT WINDOW-FUNCTION angezeigt. Jetzt können Sie die Grafik für ln(X) erstellen. Drücken Sie ERASE DRAW, um die natürliche Logarithmusfunktion darzustellen.

Um diese Grafik mit Bezeichnungen zu versehen, drücken Sie EDIT NXT LABEL. Drücken Sie MENU, um die Menübezeichnungen zu entfernen und ein Vollbild der Grafik anzuzeigen. Drücken Sie NXT, um zum aktuellen Grafikmenü zurückzukehren. Drücken Sie NXT PICT, um zum ursprünglichen Grafikmenü zurückzukehren.

Um die Punktkoordinaten der Kurve zu ermitteln, drücken Sie TRACE (der Cursor bewegt sich dann zu einem Punkt auf der Kurve, der etwa in der Mitte des horizontalen Bereichs liegt). Drücken Sie anschließend (X,Y), um die Koordinaten der aktuellen Cursorposition anzuzeigen. Diese Koordinaten erscheinen am unteren Bildschirmrand. Mit den Pfeiltasten „nach links" bzw. „nach rechts" können Sie den Cursor entlang der Kurve bewegen. Wenn Sie den Cursor entlang der Kurve bewegen, werden die Koordinaten der Kurve am unteren Bildschirmrand angezeigt. Prüfen Sie dies anhand der Daten Y:1,00E0, X:2,72E0. Das ist der Punkt (*e,1*), wenn *ln(e) = 1* ist. Drücken Sie NXT, um zum Grafikmenü zurückzukehren.

Wenn Sie jetzt FCN ROOT drücken, gelangen Sie zum Schnittpunkt der Kurve mit der x-Achse. Der Taschenrechner gibt den Wert `Root` (Kurvenbeginn) zurück: `1`, und bestätigt, dass *ln(1) = 0* ist. Drücken Sie NXT NXT PICT CANCL, um zum Fenster PLOT WINDOW – FUNCTION zurückzukehren. Drücken Sie ENTER, um zur normalen Anzeige zurückzukehren. Sie werden feststellen, dass der Kurvenbeginn aus dem Grafikmenü in den Stack des Taschenrechners kopiert worden ist.

Hinweis: Wenn Sie VAR drücken, enthält die Variablenliste neue Variablen mit den Bezeichnungen X und Y1 . Drücken Sie ↱ Y1 , um den Inhalt dieser Variablen anzuzeigen. Sie erhalten das Programm << → X 'LN(X)' >> und Sie werden feststellen, dass dies dasselbe Programm ist wie das, welches aus der Definition der Funktion 'Y1(X) = LN(X)' mithilfe von ↰ DEF resultiert. Das passiert im Grunde genommen auch, wenn Sie mittels ADD eine Funktion im Fenster PLOT – FUNCTION hinzufügen (dem Fenster, das Sie durch gleichzeitiges Drücken von ↰ Y= im RPN-Modus erreichen), d. h. die Funktion wird definiert und der Variablenliste hinzugefügt.

Drücken Sie als Nächstes ↱ X , um den Inhalt dieser Variablen anzuzeigen. Der Wert 10,275 wird im Stack abgelegt. Dieser Wert wird durch unsere Auswahl des horizontalen Anzeigebereichs festgelegt. Wir haben einen Bereich zwischen -1 und 10 für X gewählt. Zum Erstellen der Grafik erzeugt der Taschenrechner Werte innerhalb dieses Bereichs mithilfe konstanter Schritte und die erzeugten Werte werden beim Zeichnen des Graphen nach und nach in der Variablen X abgelegt. Im horizontalen Bereich (-1,10) scheint die verwendete Schrittweite 0,275 zu sein. Wenn der Wert für X größer als der Maximalwert des Bereichs wird (in diesem Fall, wenn X = 10,275 ist), dann endet die Darstellung des Graphs. Der letzte Wert für X, der für die Grafik berücksichtigt wird, wird in der Variablen X festgehalten. Löschen Sie X und Y1, bevor Sie fortfahren.

Grafik einer Exponentialfunktion

Laden Sie zunächst die Funktion *exp(X)* durch Drücken der linken Umschalttaste ← und der Taste Y= (EEX), um zum Fenster PLOT-FUNCTION zu gelangen, und zwar gleichzeitig, wenn Sie im RPN-Modus sind. Drücken Sie DEL, um die Funktion LN(X) zu löschen, wenn Sie Y1 nicht schon wie oben vorgeschlagen gelöscht haben. Drücken Sie ADD und anschließend ← e^x ALPHA X ENTER, um EXP(X) einzugeben und zum Fenster PLOT-FUNCTION zurückzukehren. Drücken Sie NXT OK, um zur normalen Anzeige zurückzukehren.

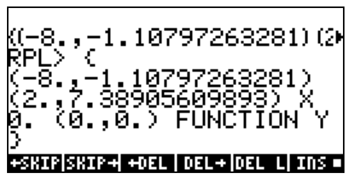
Wenn Sie im RPN-Modus sind, drücken Sie gleichzeitig die linke Umschalttaste ← und die Taste WIN (F2), um das Fenster PLOT WINDOW - FUNCTION anzuzeigen. Ändern Sie den Wert für H-VIEW folgendermaßen: H-View: -8 2

und zwar über 8 +/- OK 2 OK. Drücken Sie anschließend AUTO. Nachdem der vertikale Bereich berechnet ist, drücken Sie ERASE DRAW, um die Exponentialfunktion darzustellen.

Um diese Grafik mit Bezeichnungen zu versehen, drücken Sie EDIT NXT LABEL. Drücken Sie MENU, um die Menübezeichnungen zu entfernen und ein Vollbild der Grafik zu erhalten. Drücken Sie NXT NXT PICT CANCL, um wieder in das Fenster PLOT WINDOW – FUNCTION zu gelangen. Drücken Sie ENTER, um zur normalen Anzeige zurückzukehren.

Die Variable PPAR

Drücken Sie VAR, um bei Bedarf zum Variablenmenü zurückzukehren. In Ihrem Variablenmenü sollte sich eine Variable mit der Bezeichnung PPAR befinden. Drücken Sie ↱ PPAR, um den Inhalt dieser Variablen in den Stack zu laden. Drücken Sie die Pfeiltaste „nach unten", um den Stack-Editor zu starten; und verwenden Sie die Pfeiltasten, um den vollständigen Inhalt von PPAR anzuzeigen. Auf dem Bildschirm erscheinen die folgenden Werte:



PPAR steht für *Plot PARameters* (=Parameter darstellen) und der Inhalt enthält zwei geordnete Paare reeller Zahlen, (-8.;-1,10797263281) und (2.;7,38905609893),

welche jeweils die *Koordinaten der unteren linken und der oberen rechten Ecke* der Darstellung angeben. Als Nächstes listet PPAR den *Name der unabhängigen Variablen,* X, gefolgt von einer Zahl, die das *Inkrement der unabhängigen Variable* bei der Generierung der Darstellung festlegt. Der hier angezeigte Wert ist der voreingestellte Wert, Null (0.), mit dem das Inkrement in X festgelegt wird, was in der grafischen Darstellung 1 Pixel entspricht. Das nächste PPAR-Element ist eine *Liste, die zuerst die Koordinaten des Schnittpunkts der Achsen enthält*, d. h. (0.,0.), *gefolgt von einer Liste, welche die jeweiligen Skalenstrich-Erläuterungen* für die x- und y-Achse festlegt {# 10d # 10d}. Anschließend ist in PPAR die *Darstellungsart* zu finden, die erzeugt werden soll, d. h. FUNCTION, und schließlich die *Bezeichnung der y-Achse*, d. h. Y.

Die Variable PPAR, sofern nicht vorhanden, wird jedes Mal erzeugt, wenn eine Darstellung erzeugt wird. Der Inhalt der Funktion ändert sich entsprechend der Darstellungsart und der im PLOT-Fenster gewählten Optionen (das Fenster, das durch gleichzeitiges Betätigen der Tasten ↰ und WIN (F2) angezeigt wird).

Umkehrfunktionen und deren grafische Darstellung

Nehmen wir an, $y = f(x)$. Wenn wir nun eine Funktion $y = g(x)$ *finden, bei der* $g(f(x)) = x$, dann können wir sagen, dass $g(x)$ die *Umkehrfunktion* von $f(x)$ ist. In der Regel wird die Schreibweise $g(x) = f^{-1}(x)$ verwendet, um eine Umkehrfunktion anzuzeigen. Mit dieser Schreibweise können wir dann Folgendes schreiben: Wenn $y = f(x)$, dann ist $x = f^{-1}(y)$. Also $f(f^{-1}(x)) = x$ und $f^{-1}(f(x)) = x$.

Wie schon zuvor erwähnt, sind die Funktionen ln(x) und exp(x) die Umkehrung der jeweils anderen, d. h. ln(exp(x)) = x und exp(ln(x)) = x. Das kann mit dem Taschenrechner geprüft werden, indem folgende Ausdrücke in den Equation Writer eingegeben und berechnet werden : LN(EXP(X)) und EXP(LN(X)). Beide Ergebnisse sollten X sein.

Wenn eine Funktion $f(x)$ und deren Umkehrung $f^{-1}(x)$ gleichzeitig auf denselben Achsen dargestellt werden, spiegeln sich deren Graphen gegenseitig an der Linie $y = x$. Diese Tatsache kann mit dem Taschenrechner für die Funktionen LN(X) und EXP(X) folgendermaßen überprüft werden:

Wenn Sie im RPN-Modus sind, drücken Sie gleichzeitig [←] Y=. Die Funktion Y1(X) = EXP(X) sollte im Fenster PLOT - FUNCTION noch aus der vorherigen Übung zur Verfügung stehen. Drücken Sie ADD und geben Sie die Funktion Y2(X) = LN(X) ein. Laden Sie außerdem die Funktion Y3(X) = X. Drücken Sie NXT OK, um zur normalen Anzeige zurückzukehren.

Wenn Sie im RPN-Modus sind, drücken Sie gleichzeitig [←] WIN und ändern Sie den Bereich von H-VIEW folgendermaßen: H-View: -8 8

Drücken Sie AUTO, um den vertikalen Bereich zu erzeugen. Drücken Sie ERASE DRAW, um die Grafik y = ln(x), y = exp(x) und y =x zu erzeugen, und zwar gleichzeitig, wenn Sie im RPN-Modus sind.

Sie werden feststellen, dass nur die Kurve für y = exp(x) deutlich sichtbar ist. Bei der AUTO-Auswahl des vertikalen Bereichs ist ein Fehler unterlaufen. Folgendes ist passiert: Wenn Sie AUTO im Fenster PLOT FUNCTION –

WINDOW drücken, dann erzeugt der Taschenrechner den vertikalen Bereich entsprechend der ersten Funktion in der Liste der darzustellenden Funktionen. Und in diesem Fall ist dies die Funktion Y1(X) = EXP(X). Sie müssen den vertikalen Bereich nun selbst eingeben, damit die anderen beiden Funktionen in derselben Darstellung angezeigt werden.

Drücken Sie CANCL, um wieder in das Fenster PLOT FUNCTION – WINDOW zu gelangen. Ändern Sie die vertikalen und horizontalen Bereiche folgendermaßen: H-View: -8 8, V-View: -4 4

Durch Auswählen dieser Bereiche stellen Sie sicher, dass der Maßstab der Grafik vertikal zu horizontal 1:1 bleibt. Drücken Sie ERASE DRAW zur Darstellung der natürlichen Logarithmus-, Exponential- und *y* = *x* Funktionen. Die Grafik wird zeigen, dass LN(X) und EXP(X) Spiegelungen ihrer selbst an der Linie y = X sind. Drücken Sie CANCL, um zum Fenster PLOT WINDOW – FUNCTION zurückzukehren. Drücken Sie ENTER, um zur normalen Anzeige zurückzukehren.

Zusammenfassung der Funktionsdarstellungen

In diesem Abschnitt erhalten Sie Informationen bezüglich der Fenster PLOT SETUP, PLOT-FUNCTION und PLOT WINDOW, welche Sie durch Drücken der linken Umschalttaste zusammen mit den Softkeys F1 bis F4 erreichen. Auf Basis der oben gezeigten Beispiele zum Erstellen von Grafiken gehen Sie bei der FUNCTION sdarstellung (d. h. die Darstellung einer oder mehrerer Funktionen der Form Y = F(X)) folgendermaßen vor:

Gleichzeitig ← 2D/3D drücken, wenn Sie im RPN-Modus sind: Zugriff auf das Fenster PLOT SETUP. Falls erforderlich, ändern Sie `TYPE` in `FUNCTION` und geben Sie anschließend die Bezeichnung der unabhängigen Variable ein.

Einstellungen:

- Wenn Sie `_Simult` aktivieren, bedeutet das im Fall von zwei oder mehreren Darstellungen in derselben Grafik, dass diese beimErstellen der Grafik gleichzeitig dargestellt werden.
- Wenn Sie `_Connect` aktivieren, bedeutet dies, dass die Kurve eine durchgehende Kurve ist, und keine Aneinanderreihung einzelner Punkte.

- Wenn Sie `_Pixels` aktivieren, bedeutet dies, dass die durch `H-Tick` und `V-Tick` angezeigten Markierungen durch entsprechend viele Pixel getrennt werden.
- Der voreingestellte Wert für `H-Tick` und `V-Tick` ist 10.

Softkey-Menüoptionen:

- Mit EDIT können Sie Wertfunktionen in einem ausgewählten Feld ändern.
- Mit CHOOS können Sie die Art der Darstellung auswählen, wenn das Feld `Type:` hervorgehoben ist. Für diese Übung wird dieses Feld auf FUNCTION gesetzt.

> **Hinweis**: Die Softkeys EDIT und CHOOS stehen nicht gleichzeitig zur Verfügung. Es steht entweder der eine oder der andere zur Auswahl, je nachdem, welches Eingabefeld markiert ist.

- Drücken Sie den Softkey AXES, um die Darstellung von Achsen in der Grafik zu aktivieren bzw. zu deaktivieren. Wenn die Option 'plot axes' (Achsen darstellen) aktiviert ist, erscheint ein kleines Quadrat im Tastenfeld: AXES■ . Wenn dieses Quadrat nicht vorhanden ist, werden die Achsen in der Grafik nicht dargestellt.
- Mithilfe von ERASE löschen Sie die aktuell im Grafik-Display angezeigten Graphen.
- Mit DRAW erzeugen Sie einen Graph entsprechend des aktuellen Inhalts von PPAR für die Gleichungen, die im Fenster PLOT-FUNCTION aufgelistet sind.
- Drücken Sie NXT, um zur zweiten Softkey-Ebene dieses Bildschirms zu gelangen.
- Mit RESET setzen Sie alle ausgewählten Felder auf ihren Vorgabewert zurück.
- Mit CANCL machen Sie alle Änderungen am Fenster PLOT SETUP rückgängig und kehren zum normalen Taschenrechneranzeige zurück.
- Drücken Sie OK, um alle Änderungen im Fenster PLOT SETUP zu speichern und zum normalen Taschenrechneranzeige zurückzukehren.

Gleichzeitig ← Y= drücken, wenn Sie im RPN-Modus sind: Rufen Sie das Fenster PLOT auf (in diesem Fall wird es mit PLOT –FUNCTION bezeichnet).

Softkey-Menüoptionen:

- Mit EDIT können hervorgehobene Gleichungen geändert werden.
- Mit ADD können neue Gleichungen der Darstellung hinzugefügt werden.

> **Hinweis**: Mit ADD bzw. EDIT wird der Equation Writer EQW gestartet, mit dem Sie neue Gleichungen eingeben bzw. vorhandene Gleichungen ändern können.

- Mit DEL können hervorgehobene Gleichungen entfernt werden.
- Mit CHOOS können Gleichungen hinzugefügt werden, die bereits in Ihrem Variablenmenü definiert worden sind, die aber nicht im Fenster PLOT – FUNCTION angezeigt werden.
- Mithilfe von ERASE löschen Sie Kurven, die aktuell im Grafikdisplay angezeigt werden.
- Mit DRAW erzeugen Sie eine Kurve entsprechend des aktuellen Inhalts von PPAR für die Gleichungen, die im Fenster PLOT-FUNCTION aufgelistet sind.
- Drücken Sie NXT, um die zweite Menüliste zu aktivieren.
- Mit MOVE↓ und MOVE↑ kann eine ausgewählte Gleichung eins nach oben bzw. eins nach unten verschoben werden.
- Falls gewünscht, können mit CLEAR alle Gleichungen, die derzeit im Fenster PLOT – FUNCTION aktiviert sind, aus dem Fenster gelöscht werden. Der Taschenrechner fordert Sie auf, den Löschvorgnag zu bestätigen. Wählen Sie YES und drücken Sie OK, um mit dem Löschen aller Funktionen fortzufahren. Wählen Sie NO und drücken Sie OK, um die Option CLEAR zu deaktivieren.
- Drücken Sie anschließend OK, um zur normalen Taschenrechner-Anzeige zurückzukehren.

Gleichzeitig ← WIN drücken, wenn Sie im RPN-Modus sind: Zugriff auf das Fenster PLOT WINDOW .

Einstellungen:

- Geben Sie im Darstellungsfenster die obere und untere Begrenzung für die horizontale Ansicht (H-View) und die vertikale Ansicht (V-View) ein. Oder,
- geben Sie die untere und obere Begrenzung für die horizontale Ansicht (H-View), ein und drücken Sie AUTO, während der Cursor in einem der V-

View-Felder steht, um den Bereich der vertikalen Ansicht (V-View) automatisch zu erzeugen. Oder,

- geben Sie die untere und obere Begrenzung für die vertikale Ansicht (V-View) ein und drücken Sie AUTO, während der Cursor in einem der H-View-Felder steht, um den Bereich der horizontalen Ansicht (H-View) automatisch zu erzeugen.
- Der Taschenrechner verwendet den Bereich der horizontalen Ansicht (H-View) dazu, Datenwerte für die Grafik zu erzeugen, es sei denn, Sie ändern die Optionen Indep Low, (Indep) High bzw. (Indep) Step. Diese Werte bestimmen jeweils die Mindest-, Maximal- und Inkrementwerte der unabhängigen Variablen, die in der Darstellung verwendet werden. Wenn die Felder Indep Low, (Indep) High und (Indep) Step auf den Wert Default eingestellt sind, verwendet der Taschenrechner die Mindest- und Maximalwerte, die in der H-VIEW festgelegt sind.

- Wenn _Pixels aktiviert ist, bedeutet dies, dass die Inkremente der unabhängigen Variablen in Pixel angegeben sind und nicht als Grafikkoordinaten.

Softkey-Menüoptionen:

- Mit EDIT können alle Einträge im Fenster bearbeitet werden.
- Verwenden Sie AUTO gemäß den Erläuterungen unter *Einstellungen*.
- Mithilfe von ERASE löschen Sie Kurven, die aktuell im Grafik-Display angezeigt werden.
- Mit DRAW erzeugen Sie eine Kurve entsprechend des aktuellen Inhalts von PPAR für die Gleichungen, die im Fenster PLOT-FUNCTION aufgelistet sind.
- Drücken Sie NXT, um die zweite Menüliste zu aktivieren.
- Mit RESET setzen Sie das ausgewählte Feld (d. h. dort, wo sich der Cursor befindet) auf seinen Vorgabewert zurück.
- Mit CALC gelangen Sie in den Stack des Taschenrechners, um Rechenoperationen durchzuführen, die zur Ermittlung von Werten notwendig sein können, die für eine der Optionen in diesem Fenster erforderlich sein könnten. Wenn Sie auf den Stack des Taschenrechners zugreifen können, sind auch die Softkeys CANCL und OK verfügbar.

- Verwenden Sie CANCL, wenn Sie die aktuelle Rechnung beenden und zum Bildschirm PLOT WINDOW zurückkehren möchten. Oder,
- Verwenden Sie OK, um die Ergebnisse Ihrer Berechnung zu übernehmen und zur Maske PLOT WINDOW zurückzukehren.
- Mit TYPES erhalten Sie Informationen über die Objektarten, die im ausgewählten Optionsfeld verwendet werden können.
- Mit CANCL machen Sie alle Änderungen im Fenster PLOT WINDOW rückgängig und kehren zum normalen Taschenrechneranzeige zurück.
- Drücken Sie OK, um alle Änderungen im Fenster PLOT WINDOW anzunehmen und zum normalen Taschenrechnerdisplayzurückzukehren.

Gleichzeitig ← GRAPH drücken, wenn Sie im RPN-Modus sind: Stellt die Kurve auf Basis der Einstellungen dar, die in der Variablen PPAR und den aktuellen, im Fenster PLOT – FUNCTION definierten Funktionen gespeichert sind. Wenn im Grafik-Display bereits eine andere Kurve als die, die Sie gerade darstellen, vorhanden ist, wird diese mit der neuen Darstellung überschrieben. Wenn dies nicht das gewünschte Ergebnis ist, sollten Sie die in den Fenstern PLOT SETUP, PLOT-FUNCTION und PLOT WINDOW verfügbaren Softkeys ERASE DRAW verwenden.

Darstellung von Winkel- und Hyperbelfunktionen

Die oben zur Darstellung von LN(X) und EXP(X) beschriebenen Verfahren können – einzeln oder auch gleichzeitig – angewandt werden, um beliebige Funktionen der Form y = f(x) darzustellen. Wir überlassen es dem Leser, zur Übung Darstellungen von Winkel- und Hyperbelfunktionen und deren Umkehrfunktionen zu erstellen. Die folgende Tabelle gibt Werte vor, die jeweils für die vertikalen und horizontalen Bereiche verwendet werden können. Sie können die Funktion Y=X bei der gleichzeitigen Darstellung einer Funktion und ihrer Umkehrfunktion einschließen, um ihre Spiegelung über die Linie Y = X nachzuprüfen.

	Bereich H-View		Bereich V-View:	
Funktion:	**Minimum**	**Maximum**	**Minimum**	**Maximum**
SIN(X)	-3.15	3.15	AUTO	
ASIN(X)	-1.2	1.2	AUTO	

SIN & ASIN	-3.2	3.2	-1.6	1.6
COS(X)	-3.15	3.15	AUTO	
ACOS(X)	-1.2	1.2	AUTO	
COS & ACOS	-3.2	3.2	-1.6	1.6
TAN(X)	-3.15	3.15	-10	10
ATAN(X)	-10	10	-1.8	1.8
TAN & ATAN	-2	-2	-2	-2
SINH(X)	-2	2	AUTO	
ASINH(X)	-5	5	AUTO	
SINH & ASINH	-5	5	-5	5
COSH(X)	-2	2	AUTO	
ACOSH(X)	-1	5	AUTO	
COS & ACOS	-5	5	-1	5
TANH(X)	-5	5	AUTO	
ATANH(X)	-1.2	1.2	AUTO	
TAN & ATAN	-5	5	-2.5	2.5

Eine Wertetabelle für Funktionen erstellen

Mit den Tastenkombinationen ← TBLSET (F5) und ← TABLE (F6), im RPN-Modus gleichzeitig gedrückt, kann der Benutzer eine Wertetabelle für Funktionen erstellen. Wir erstellen zum Beispiel eine Tabelle für die Funktion Y(X) = X/(X+10) im Bereich -5 < X < 5, und zwar gemäß folgender Anleitung:

- Wir erzeugen Werte der Funktion f(x), die oben definiert ist, für Werte von x von –5 bis 5 und in Schritten von 0,5. Zunächst muss sichergestellt werden, dass der Grafiktyp im Fenster PLOT SETUP (gleichzeitig ← 2D/3D drücken, wenn im RPN-Modus) auf **FUNCTION** eingestellt ist. Das Feld neben der Option *Type* ist hervorgehoben. Wenn dieses Feld nicht bereits auf **FUNCTION** eingestellt ist, drücken Sie den Softkey CHOOS und wählen die Option **FUNCTION** und drücken Sie OK.
- Drücken Sie anschließend ▼, um das Feld vor der Option EQ hervorzuheben, und geben Sie den Funktionsausdruck: 'X/(X+10)' ein.

- Um die im Fenster PLOT SETUP vorgenommenen Änderungen zu übernehmen, drücken Sie NXT OK. Damit kehren Sie zur normalen Anzeige zurück.
- Der nächste Schritt ist das Öffnen des Fensters für das Einrichten der Tabelle mithilfe der Tastenkombination ← TBLSET (d. h. Softkey F5) – gleichzeitig, wenn Sie im RPN-Modus sind. Damit rufen Sie einen Bildschirm auf, in dem Sie den Anfangswert (*Start*) und die Schritte *(Step)* auswählen können. Geben Sie Folgendes ein: 5 +/- OK 0 • 5 OK 0 • 5 OK (d. h. Zoomfaktor = 0,5). Betätigen Sie den Softkey ✓CHK, sodass vor der Option *Small Font* (kleine Schriftart) bei Bedarf ein Häkchen erscheint. Drücken Sie anschließend OK. Damit kehren Sie zur normalen Anzeige zurück.

Die Variable TPAR

Wenn Sie das Einrichten der Tabelle abgeschlossen haben, erzeugt Ihr Taschenrechner eine Variable mit der Bezeichnung TPAR (Table PARameters = Tabellenparameter), in der die für die zu erzeugende Tabelle relevanten Daten gespeichert werden. Um den Inhalt dieser Variablen anzuzeigen, drücken Sie → TPAR.

- Um die Tabelle anzuzeigen, drücken Sie ← TABLE (d. h., Softkey F6) – gleichzeitig, wenn im RPN-Modus. Dadurch wird eine Tabelle für die Werte *x = -5; -4,5,…* mit den entsprechenden Werten für f(x) erzeugt, die standardmäßig als Y1 aufgeführt sind. Mit den Pfeiltasten „nach oben“ und „nach unten“ können Sie sich in der Tabelle bewegen. Sie werden feststellen, dass für die unabhängige Variable x kein Endwert festgelegt wurde. Deshalb geht die Tabelle über den vorher empfohlenen Maximalwert für x, nämlich x = 5, hinaus.

Wenn die Tabelle angezeigt wird, sind unter anderem folgende Optionen verfügbar: ZOOM, BIG und DEFN.

- Wenn DEFN gewählt wurde, wird die Definition der unabhängigen Variable angezeigt.

- Mit der Taste BIG wird die Schriftart in der Tabelle von klein auf groß und umgekehrt geändert. Probieren Sie es aus.
- Wenn die Taste ZOOM gedrückt wird, erscheint ein Menü mit den folgenden Optionen: *In*, *Out*, *Decimal*, *Integer* und *Trig*. Führen Sie folgende Übungen aus:

  - Wenn die Option *In* hervorgehoben ist, drücken Sie OK. Die Tabelle wird daraufhin erweitert und die x-Inkremente betragen jetzt 0,25 und nicht mehr 0,5. Der Taschenrechner multipliziert einfach das ursprüngliche Inkrement von 0,5 mit dem Zoomfaktor 0,5, um das neue Inkrement von 0,25 zu generieren. Daher ist die *Zoom In*-Option nützlich, wenn Sie eine höhere Auflösung für die x-Werte in Ihrer Tabelle erzielen möchten.
  - Um die Auflösung zusätzlich um den Faktor 0,5 zu erhöhen, drücken Sie ZOOM erneut, wählen *In* erneut und drücken anschließend OK. Das Inkrement für x beträgt nun 0,0125.
  - Um das vorherige x-Inkrement wiederherzustellen, drücken Sie ZOOM ▲ OK und wählen Sie anschließend die Option *Un-zoom*. Das x-Inkrement wird auf 0,25 erhöht.
  - Um das ursprüngliche x-Inkrement von 0,5 wiederherzustellen, können Sie erneut die Option *Un-zoom* wählen oder die Option *zoom out* aktivieren, indem Sie ZOOM OK drücken.
  - Mit der ZOOM -Option „Decimal“ (Dezimal) erzeugen Sie x-Inkremente von 0,10.
  - Mit der ZOOM -Option „Integer“ (Ganzzahl) erzeugen Sie x-Inkremente von 1.
  - Mit der Option „Trig“ erzeugen Sie Inkremente aus Bruchteilen von π, diese ist deshalb für die Darstellung von Winkelfunktionen hilfreich.
  - Um zur normalen Taschenrechner-Anzeige zurückzukehren, drücken Sie ENTER.

Darstellungen in Polarkoordinaten

Zunächst ist es sinnvoll, die in den vorhergehenden Beispielen verwendeten Variablen zu löschen (z. B., X, EQ, Y1, PPAR), und zwar mit der Funktion PURGE (TOOL PURGE). Damit werden alle Parameter, die mit Grafiken verbunden sind, gelöscht. Drücken Sie [VAR], um zu prüfen, ob tatsächlich alle Variablen entfernt wurden.

Versuchen Sie, die Funktion f(θ) = 2(1-sin(θ)) wie folgt darzustellen:

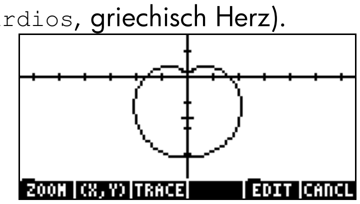
- Stellen Sie zunächst sicher, dass das Winkelmaß Ihres Taschenrechners auf Radianten eingestellt ist.
- Drücken Sie [←] 2D/3D , gleichzeitig wenn im RPN-Modus, um zum Fenster PLOT SETUP zu gelangen.
- Ändern Sie `TYPE` in `Polar`, indem Sie CHOOS [▼] OK drücken.
- Drücken Sie [▼] und geben Sie Folgendes ein:

['] [2] [×] [←] () [1] [−] [SIN] [ALPHA] [→] [T] OK.

- Der Cursor befindet sich jetzt im Feld `Indep`. Drücken Sie ['] [ALPHA] [→] [T] OK, um die unabhängige Variable auf θ zu setzen.
- Drücken Sie [NXT] OK, um zur normalen Anzeige zurückzukehren.
- Drücken Sie [←] WIN gleichzeitig, wenn im RPN-Modus, um zum PLOT-Fenster zu gelangen (in diesem Fall wird es als PLOT –POLAR bezeichnet).
- Ändern Sie den Bereich von H-VIEW auf –8 bis 8 durch Drücken von [8] [+/-] OK [8] OK und den Bereich von V-VIEW auf -6 bis 2 durch Drücken von [6] [+/-] OK [2] OK.

> **Hinweis**: H-VIEW und V-VIEW legen lediglich die Skala des Anzeigefensters fest und deren Bereich bezieht sich nicht auf den Bereich der Werte der unabhängigen Variablen in diesem Fall.

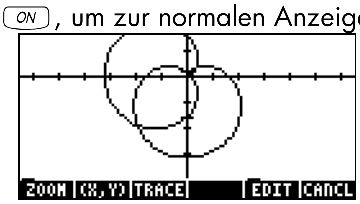
- Setzen Sie den Wert `Indep Low` auf 0 und den Wert High auf 6,28 (≈ 2π) durch Drücken von: [0] OK [6] [.] [2] [8] OK.
- Drücken Sie ERASE DRAW, um die Funktion in Polarkoordinaten darzustellen. Das Ergebnis ist eine herzförmige Kurve. Diese Kurve wird als *Kardioide* bezeichnet (von `cardios`, griechisch Herz).



- Drücken Sie EDIT NXT LABEL MENU um den Graphen mit Bezeichnern anzuzeigen. Drücken Sie NXT, um zum Menü zurückzukehren. Drücken Sie NXT PICT, um zum ursprünglichen Grafikmenü zurückzukehren.
- Drücken Sie TRACE (X,Y) , um die Kurve zu verfolgen. Die am unteren Rand des Displays angezeigten Daten sind der Winkel θ und der Radius r, wobei letzterer mit dem Buchstaben Y bezeichnet ist (Vorgabebezeichnung abhängiger Variablen).
- Drücken Sie NXT CANCL, um wieder zum Fenster PLOT WINDOW zu gelangen. Drücken Sie NXT OK, um zur normalen Anzeige zurückzukehren.

In dieser Übung haben Sie die Gleichung eingegeben, die direkt im Fenster PLOT SETUP dargestellt werden soll. Sie können darzustellende Gleichungen auch im Fenster PLOT eingeben, und zwar durch gleichzeitiges Drücken von ← Y= im RPN-Modus. Wenn Sie zum Beispiel ← Y= drücken, nachdem Sie die vorherige Übung abgeschlossen haben, dann wird die Gleichung '2*(1-SIN(θ))' hervorgehoben. Angenommen, Sie möchten zusammen mit der vorherigen Gleichung auch die Funktion '2*(1-COS(θ))' darstellen.

- Drücken Sie ADD und drücken Sie 2 × ← () 1 − COS ALPHA → T ENTER, um die neue Gleichung einzugeben.
- Drücken Sie ERASE DRAW, um die beiden Gleichungen in derselben Grafik darzustellen. Das Ergebnis sind zwei sich überschneidende *Kardioiden.* Drücken Sie CANCL ON, um zur normalen Anzeige zurückzukehren.



Darstellung von Kegelschnitt-Kurven

Die allgemeinste Form einer Kegelschnitt-Kurve in der x/y-Ebene ist: $Ax^2+By^2+Cxy+Dx+Ey+F = 0$. Wir erkennen auch die Kegelschnitt-Gleichungen, die in der Normalform für folgende geometrische Abbildungen gegeben sind:

- Kreis : $(x\text{-}x_o)^2+(y\text{-}y_o)^2 = r^2$
- Ellipse : $(x\text{-}x_o)^2/a^2 + (y\text{-}y_o)^2/b^2 = 1$
- Parabel: $(y\text{-}b)^2 = K(x\text{-}a)$ or $(x\text{-}a)^2 = K(y\text{-}b)$
- Hyperbel: $(x\text{-}x_o)^2/a^2 + (y\text{-}y_o)^2/b^2 = 1$ or $xy = K$,

wobei x_o, y_o, a, b und K konstant sind.

Die Bezeichnung *Kegelschnitt-Kurve* resultiert daraus, dass diese Abbildungen (Kreis, Ellipse, Parabel oder Hyperbel) aus der Verschneidung einer Ebene mit einem Kegel entstehen. Ein Kreis ist beispielsweise die Verschneidung eines Kegels mit einer Ebene, die rechtwinklig zur Hauptachse des Kegels steht.

Mit dem Taschenrechner ist es möglich, eine oder mehrere Kegelschnitt-Kurven darzustellen, und zwar durch Auswahl von „Conic" für die Funktion TYPE im Menü PLOT. Stellen Sie sicher, die Variablen PPAR und EQ zu löschen, bevor Sie fortfahren. Speichern Sie beispielsweise die Gleichungsfolge

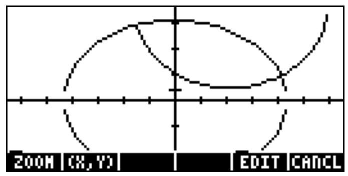
{ '(X-1)^2+(Y-2)^2=3' , 'X^2/4+Y^2/3=1' }

in der Variablen EQ.

Wir erkennen diese als Gleichungen eines Kreises, dessen Mittelpunkt auf (1,2) liegt mit einem Radius √3, und einer Ellipse, deren Mittelpunkt auf (0,0) liegt mit einer Halbachsenlänge a = 2 und b = √3.

- Gehen Sie ins PLOT-Menü, und zwar durch gleichzeitiges Drücken von [↰] [2D/3D] im RPN-Modus, und wählen Sie anschließend `Conic als TYPE` aus. Die Liste der Gleichungen erscheint im Feld EQ.
- Stellen Sie sicher, dass die unabhängige Variable (`Indep`) auf 'X' und die abhängige Variable (`Depnd`) auf 'Y' gesetzt ist.
- Drücken Sie [NXT] [OK], um zur normalen Anzeige zurückzukehren.
- Durch Drücken von [↰] [WIN] gelangen Sie in das Menü PLOT WINDOW (gleichzeitig drücken, wenn in RPN-Modus).
- Ändern Sie den Bereich für H-VIEW auf -3 bis 3 durch Drücken von [3] [+/-] [OK] [3] [OK]. Ändern Sie auch den Bereich für V-VIEW auf -1,5 bis 2 durch Drücken von [1] [.] [5] [+/-] [OK] [2] [OK].

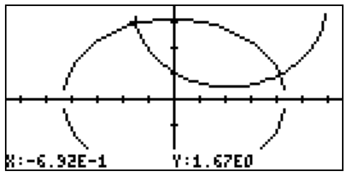
- Ändern Sie die Felder *Indep Low*: und *High:* auf “Default” (Vorgabewert) durch Drücken von NXT RESET, während diese beiden Felder hervorgehoben sind. Wählen Sie die Option *Reset value* (Wert zurückstellen), nachdem Sie RESET gedrückt haben. Drücken Sie OK, um das Zurückstellen der Werte abzuschließen. Drücken Sie NXT, um zum Hauptmenü zurückzukehren.
- Stellen Sie den Graph dar: ERASE DRAW.

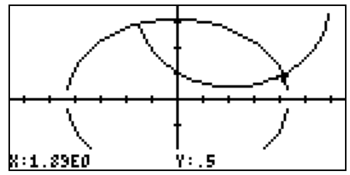


Hinweis: Die Bereiche <u>`H-VIEW`</u> und <u>`V-VIEW`</u> wurden ausgewählt, um die Überschneidung der beiden Kurven anzuzeigen. Es gibt keine allgemeine Regel für die Auswahl der Bereiche, sie sollten lediglich basierend auf den Kurven ausgewählt werden. Bei den oben gezeigten Gleichungen ist beispielsweise bekannt, dass der Kreis auf x von -3+1 = -2 bis 3+1 = 4 geht, und auf y von -3+2=-1 bis 3+2=5. Zudem reicht die Ellipse, deren Zentrum am Ursprung (0,0) liegt, von -2 bis 2 auf x, und von -√3 bis √3 auf y.

Beachten Sie, dass die äußersten Bereiche des Kreises und der Ellipse jeweils nicht dargestellt werden. Das gilt für alle Kreise und Ellipsen, die unter der Einstellung `Conic` als `TYPE` dargestellt werden.

- Zur Anzeige der Bezeichnungen: EDIT NXT LABEL MENU
- Zur Wiederherstellung des Menüs: NXT NXT PICT
- Um die Koordinaten von Schnittpunkten zu ermitteln, drücken Sie den Softkey (X,Y) und bewegen Sie den Cursor mithilfe der Pfeiltasten so nah wie möglich an diese Punkte heran. Die Koordinaten des Cursors werden im Display angezeigt. Der linke Schnittpunkt befindet sich beispielsweise in der Nähe von (-0,692; 1,67) und der rechte Schnittpunkt liegt etwa bei (1,89;0,5).





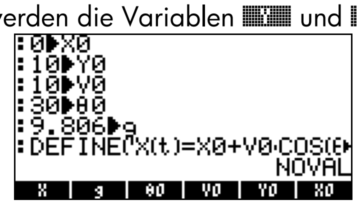
- Um das Menü wieder herzustellen und in das Fenster PLOT zurückzukehren, drücken Sie NXT CANCL.
- Um zur normalen Taschenrechner-Anzeige zurückzukehren, drücken Sie NXT OK.

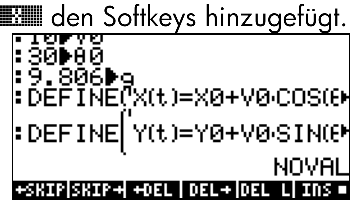
Parametrische Diagramme

Parametrische Diagramme in der Ebene sind Diagramme, deren Koordinaten durch ein System von Gleichungen x = x(t) und y = y(t) erzeugt werden, und bei denen t als Parameter bekannt ist. Ein Beispiel für eine derartige Grafik ist die Flugbahn eines Projektils, $x(t) = x_0 + v_0 \cdot \mathrm{COS}\ \theta_0 \cdot t$, $y(t) = y_0 + v_0 \cdot \sin\ \theta_0 \cdot t - ½ \cdot g \cdot t^2$. Um Gleichungen wie diese darzustellen, die konstante Werte x_0, y_0, v_0 und θ_0 beinhalten, müssen die Werte dieser Parameter in Variablen gespeichert werden. In diesem Beispiel erstellen Sie ein Unterverzeichnis namens 'PROJM' für PROJectile Motion (=Projektilbewegung). Speichern Sie in diesem Unterverzeichnis die folgenden Variablen: X0 = 0, Y0 = 10, V0 = 10 , θ0 = 30 und g = 9,806. Stellen Sie sicher, dass das Winkelmaß des Taschenrechners auf DEG eingestellt ist. Definieren Sie anschließend die Funktionen (über ← DEF):

X(t) = X0 + V0*COS(θ0)*t
Y(t) = Y0 + V0*SIN(θ0)*t – 0.5*g*t^2

Damit werden die Variablen Y und X den Softkeys hinzugefügt.



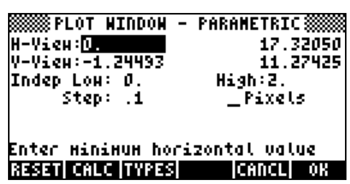


Um die Grafik selbst zu erstellen, gehen wir folgendermaßen vor:

- Drücken Sie ← 2D/3D , gleichzeitig wenn im RPN-Modus, um zum Fenster PLOT SETUP zu gelangen.
- Ändern Sie `TYPE` auf `Parametric` durch Drücken von CHOOS ▼ ▼ OK.
- Drücken Sie ▼ und geben 'X(t) + i*Y(t)' OK ein, um das parametrische Diagramm als das einer komplexen Größe zu definieren. (Die realen und imaginären Teile der komplexen Größe entsprechen den x- und y-Koordinaten der Kurve.)
- Der Cursor ist jetzt im Feld `Indep`. Drücken Sie ' ALPHA ← T OK, um die unabhängige Variable auf *t* zu ändern.
- Drücken Sie NXT OK, um zur normalen Anzeige zurückzukehren.
- Drücken Sie ← WIN gleichzeitig, wenn im RPN-Modus, um zum PLOT-Fenster zu gelangen (in diesem Fall heißt es PLOT –PARAMETRIC). Anstatt zuerst die horizontale und vertikale Ansicht zu ändern, so wie dies bei den anderen Darstellungsarten gemacht worden ist, legen wir nun zuerst die unteren und oberen Werte der unabhängigen Variablen fest, und zwar so:
- Wählen Sie das Feld `Indep Low` durch Drücken von ▼ ▼ . Ändern Sie den Wert auf 0 OK. Dann ändern Sie den Wert von `High` auf 2 OK. Geben Sie 0 • 1 OK als den `Step` –Wert ein (d. h. Schrittgröße = 0,1).

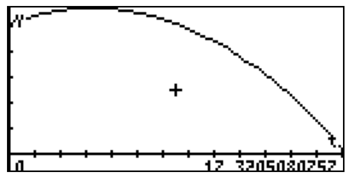
Hinweis: Durch diese Einstellungen zeigen wir, dass der Parameter t den Wert t = 0; 0,1; 0,2, … usw. annimmt bis der Wert 2,0 erreicht ist.

- Drücken Sie AUTO. Damit wird ein automatischer Wert für den Bereich H-VIEW und V-VIEW erzeugt, und dieser basiert auf den Werten der unabhängigen Variable t und den verwendeten Definitionen für X(t) und Y(t). Das Ergebnis ist Folgendes:



- Drücken Sie ERASE DRAW, um das parametrische Diagramm zu zeichnen.

- Drücken Sie EDIT NXT LABEL MENU, um die Grafik mit Bezeichnern anzuzeigen. Die Fensterparameter sind so, dass nur die Hälfte der Bezeichner der x-Achse zu sehen ist.



- Drücken Sie NXT, um zum Menü zurückzukehren. Drücken Sie NXT PICT, um zum ursprünglichen Grafikmenü zurückzukehren.
- Drücken Sie TRACE (X,Y), um die Koordinaten eines beliebigen Punktes auf der Grafik zu bestimmen. Mit ▶ und ◀ bewegen Sie den Cursor entlang der Kurve. Am unteren Rand des Displays wird der Wert des Parameters t und die Koordinaten des Cursors als (X,Y) angezeigt.
- Drücken Sie NXT CANCL, um wieder zum Fenster PLOT WINDOW zu gelangen. Dann drücken Sie ON oder NXT OK, um zur normalen Taschenrechnerdisplay zurückzukehren.

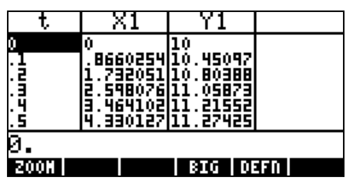
Bei Prüfung Ihrer Softkeys sehen Sie, dass Sie nun folgende Variablen haben: t, EQ, PPAR, Y, X, g, θ0, V0, Y0, X0. Die Variablen t, EQ, und PPAR werden vom Taschenrechner erzeugt, um die aktuellen Werte der Parameter zu speichern: von t, der darzustellenden Gleichung, EQ (der 'X(t) + I∗Y(t)' enthält) und der Darstellungsparameter. Die anderen Variablen enthalten die Werte von Konstanten, die in den Definitionen von X(t) und Y(t) verwendet werden.

Sie können verschiedene Werte in den Variablen speichern und neue parametrische Diagramme der in diesem Beispiel verwendeten Projektil-Gleichungen erzeugen. Wenn Sie den Inhalt des aktuellen Bildes löschen möchten bevor Sie ein neues Diagramm erstellen, müssen Sie in das Fenster PLOT, PLOT WINDOW oder PLOT SETUP wechseln, und zwar durch Drücken von ↰ Y=, ↰ WIN oder ↰ 2D/3D (im RPN-Modus müssen die beiden Tasten gleichzeitig gedrückt werden). Dann drücken Sie ERASE DRAW. Drücken Sie CANCL, um wieder zum Fenster PLOT, PLOT WINDOW oder PLOT SETUP zurückzukehren. Drücken Sie ON oder NXT OK, um zur normalen Taschenrechnerdisplay zurückzukehren.

Erzeugen einer Tabelle für parametrische Gleichungen

In einem früheren Beispiel haben wir eine Wertetabelle (X,Y) für einen Ausdruck der Form Y=f(X) erzeugt, d. h. eine Grafik des Typs Function. In diesem Abschnitt zeigen wir, wie eine Tabelle für ein parametrisches Diagramm erzeugt wird. Zu diesem Zweck nutzen wir die parametrischen Gleichungen, die im Beispiel oben definiert sind.

- Zunächst gehen wir in das Fenster TABLE SETUP durch Drücken von ← TBLSET, und zwar gleichzeitig im RPN-Modus. Wir setzen den Anfangswert der unabhängigen Variablen auf 0.0 und den Schrittgrößenwert (Step) auf 0,1. Drücken Sie OK.
- Erzeugen Sie die Tabelle durch gleichzeitiges Drücken von ← TABLE (wenn im RPN-Modus). Diese Tabelle hat drei Spalten: für den Parameter t und die Koordinaten der entsprechenden Punkte. Für diese Tabelle werden die Koordinaten mit X1 und Y1 bezeichnet.



t	X1	Y1	
0	0	10	
.1	.8660254	10.45097	
.2	1.732051	10.80388	
.3	2.598076	11.05873	
.4	3.464102	11.21552	
.5	4.330127	11.27425	

- Mit den Pfeiltasten ◀ ▶ ▲ ▼ können Sie sich in der Tabelle bewegen.
- Drücken Sie ON ,um zur normalen Anzeige zurückzukehren.

Dieser Vorgang zur Erzeugung einer Tabelle für die aktuelle Darstellungsart kann auch für andere Darstellungsarten angewendet werden.

Lösungsdarstellung für einfache Differentialgleichungen

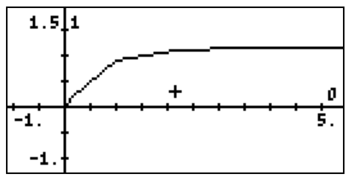
Einfache Differentialgleichungen können durch Auswahl von `Diff Eq` im Feld `TYPE` des Fensters PLOT SETUP folgendermaßen dargestellt werden: Nehmen wir an, wir wollten x(t) aus der Differentialgleichung $dx/dt = \exp(-t^2)$

darstellen, und zwar mit den folgenden Ausgangswerten: x = 0 bei t = 0. Mit dem Taschenrechner kann die Lösung der Differentialgleichung der Form Y'(T) = F(T,Y) dargestellt werden. In unserem Fall ist Y→x und T→t, und damit F(T,Y)→f(t,x) = exp(-t^2).

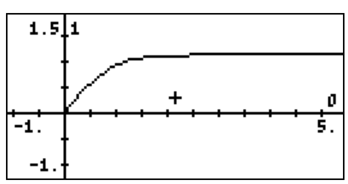
Vor Darstellung der Lösung, x(t), für t = 0 bis 5, löschen wir die Variablen EQ und PPAR.

- Drücken Sie ← 2D/3D , gleichzeitig wenn im RPN-Modus, um zum Fenster PLOT SETUP zu gelangen.
- Ändern Sie `TYPE` auf `Diff Eq`.
- Drücken Sie ▼ und drücken Sie ' ← e^x — ALPHA ← T Y^x 2 OK.
- Der Cursor ist jetzt im Feld `H-Var`. Jetzt sollte `H-Var:0` und `V-Var:1` sein. Das ist der vom Taschenrechner verwendete Code, um die darzustellenden Variablen zu identifizieren. `H-Var:0` bedeutet, dass die unabhängige Variable (die später ausgewählt wird) auf der horizontalen Achse dargestellt wird. Und `V-Var:1` bedeutet, dass die abhängige Variable (voreingestellte Standardbezeichnung 'Y') auf der vertikalen Achse dargestellt wird.
- Drücken Sie ▼ . Der Cursor ist jetzt im Feld `Indep`. Drücken Sie → ' ALPHA ← T OK, um die unabhängige Variable auf *t* zu ändern.
- Drücken Sie NXT OK, um zur normalen Anzeige zurückzukehren.
- Drücken Sie ← WIN gleichzeitig, wenn im RPN-Modus, um zum PLOT-Fenster zu gelangen (in diesem Fall heißt es PLOT WINDOW – DIFF EQ).
- Ändern Sie den Wert für H-VIEW und V-VIEW folgendermaßen: H-VIEW: -1 5, V-VIEW: -1 1,5
- Ändern Sie den Wert `Init` auf 0 und den Endwert auf 5, und zwar mithilfe von: 0 OK 5 OK.
- Die Werte „Step" und „Tol" stellen den Schritt der unabhängigen Variable und die Konvergenztoleranz dar, die von der numerischen Lösung verwendet werden sollen. Wir lassen diese Werte auf den voreingestellten Standardeinstellungen (wenn das Wort *default* nicht im Feld Step: erscheint, setzen wir mit NXT RESET alle Werte auf die voreingestellten Werte zurück. Drücken Sie NXT, um zum Hauptmenü zurückzukehren.) Drücken Sie ▼.

- Der Wert Init-Soln stellt den ursprünglichen Wert der Lösung für das numerische Ergebnis dar. In diesem Fall haben wir die Ausgangsbedingung x(0) = 0, und deshalb müssen wir diesen Wert mithilfe von 0 OK auf 0.0 ändern.
- Drücken Sie ERASE DRAW, um die Lösung für die Differentialgleichung darzustellen.
- Drücken Sie EDIT NXT LABEL MENU, um die Grafik mit Bezeichnern anzuzeigen.



- Drücken Sie NXT, um zum Menü zurückzukehren. Drücken Sie NXT PICT, um zum ursprünglichen Grafikmenü zurückzukehren.
- Wenn wir die Darstellung der Kurve betrachten, werden wir feststellen, dass die Kurve nicht sehr glatt und geschmeidig verläuft. Das liegt daran, dass der Plotter mit Zeitschritten arbeitet, die zu groß sind. Um einen feineren und geschmeidigeren Graph zu erhalten, sollte ein Schritt von 0.1 verwendet werden. Versuchen Sie es mit folgenden Tasten: CANCL ▼▼▼ . 1 OK ERASE DRAW. Die Darstellung wird dann zwar etwas länger dauern bis sie fertig ist, aber die Form wird sicherlich geschmeidiger sein.
- Drücken Sie EDIT NXT LABEL MENU, um die Achsen mit Bezeichnern und Bereichen anzuzeigen. Beachten Sie, dass die Bezeichner für die Achsen als 0 (horizontal) und 1 (vertikal) angezeigt werden. Dies sind die Definitionen für die Achsen gemäß dem Fenster PLOT WINDOW (siehe oben), d. h. H-VAR (t): 0, und V-VAR(x): 1.



- Drücken Sie NXT NXT PICT, um zum Menü und in den PICT-Bereich zurückzukehren.
- Drücken Sie (X,Y) , um die Koordinaten eines beliebigen Punktes auf dem Graphen zu bestimmen. Mit ▶ und ◀ bewegen Sie den Cursor im Grafikbereich. Am unteren Rand des Displays werden die Koordinaten des Cursors als (X,Y) angezeigt. Der Taschenrechner verwendet X und Y als voreingestellte Standardbezeichnung für die horizontale und vertikale Achse.
- Drücken Sie NXT CANCL, um wieder in den Bereich PLOT WINDOW zu gelangen. Drücken Sie anschließend ON, um zur normalen Anzeige zurückzukehren.

Weitere Einzelheiten zur Verwendung von Grafiklösungen für Differentialgleichungen finden Sie in Kapitel 16.

Truth-Plot-Funktion

Die Truth-Plot-Funktion wird verwendet, um zweidimensionale Darstellungen von Bereichen, die bestimmte mathematische Bedingungen erfüllen, zu erstellen, die wahr oder falsch sein können. Nehmen wir beispielsweise an, Sie wollen den Bereich für X^2/36 + Y^2/9 < 1 darstellen, dann gehen Sie folgendermaßen vor:

- Drücken Sie ← 2D/3D , gleichzeitig wenn im RPN-Modus, um zum Fenster PLOT SETUP zu gelangen.
- Ändern Sie TYPE auf Truth.
- Drücken Sie ▼ und geben {'(X^2/36+Y^2/9 < 1)','(X^2/16+Y^2/9 > 1)'} OK ein, um die darzustellende Bedingung zu definieren.
- Der Cursor ist jetzt im Feld `Indep`. Belassen Sie dieses auf 'X', wenn es bereits auf diese Variable eingestellt ist, oder ändern Sie es auf 'X', wenn nicht.
- Drücken Sie NXT OK, um zur normalen Anzeige zurückzukehren.
- Drücken Sie ← WIN gleichzeitig, wenn im RPN-Modus, um zum PLOT-Fenster zu gelangen (in diesem Fall PLOT WINDOW – TRUTH genannt). Belassen wir es auf dem voreingestellten Standardwert für den Fensterbereich: `H-View: -6.5 6.5`, `V-View: -3.1 3.2` (Mit NXT RESET (wählen Sie "Reset all") OK NXT werden diese zurückgestellt).

Hinweis: Wenn der Fensterbereich nicht die voreingestellten Standardeinstellungen zeigt, dann können diese am schnellsten durch Drücken von NXT RESET (wählen Sie *„Reset all")* OK NXT zurückgestellt werden.

- Drücken Sie ERASE DRAW, um den Truth-Plot zu zeichnen. Da der Taschenrechner den gesamten Darstellungsbereich Punkt für Punkt abfragt, dauert es ein paar Minuten bis der Truth-Plot fertiggestellt ist. Die aktuelle Grafik sollte eine schattierte Ellipse erzeugen mit den Halbachsen 6 und 3 (auf x und y), zentriert auf dem Ursprung.
- Drücken Sie EDIT NXT LABEL MENU, um die Grafik mit Bezeichnern anzuzeigen. Die Fensterparameter sind so, dass nur die Hälfte der Bezeichner der x-Achse zu sehen ist. Drücken Sie NXT, um zum Menü zurückzukehren. Drücken Sie NXT PICT, um zum ursprünglichen Grafikmenü zurückzukehren.
- Drücken Sie (X,Y), um die Koordinaten eines beliebigen Punktes auf der Grafik zu bestimmen. Mit den Pfeiltasten bewegen Sie den Cursor im Grafikbereich. Am unteren Rand des Displays werden die Koordinaten des Cursors als (X,Y) angezeigt.
- Drücken Sie NXT CANCEL, um wieder in den Bereich PLOT WINDOW zu gelangen. Dann drücken Sie ON oder NXT OK, um zur normalen Taschenrechnerdisplay zurückzukehren.

Wenn Sie die Bedingungen multiplizieren, dann können Sie mehr als eine Bedingung gleichzeitig darstellen. Wenn Sie beispielsweise die Kurve der Punkte darstellen möchten, bei denen $X^2/36 + Y^2/9 < 1$ ist und $X^2/16 + Y^2/9 > 1$, dann gehen Sie wie folgt vor:

- Drücken Sie ← 2D/3D, gleichzeitig wenn im RPN-Modus, um zum Fenster PLOT SETUP zu gelangen.
- Drücken Sie ▼ und geben '(X^2/36+Y^2/9 < 1)· (X^2/16+Y^2/9 > 1)' OK ein, um die darzustellenden Bedingungen zu definieren.
- Drücken Sie ERASE DRAW, um den Truth-Plot zu zeichnen. Auch hier müssen sie Geduld aufbringen, bis der Taschenrechner den Graphen fertig gestellt hat. Wenn Sie die Darstellung abbrechen möchten, dann drücken Sie einmal ON. Drücken Sie anschließend CANCEL .

Darstellung von Histogrammen, Balkendiagrammen und Punktdiagrammen

Histogramme, Balkendiagramme und Punktdiagramme werden zur Darstellung von Einzeldaten verwendet, die in der reservierten Variable ΣDAT abgelegt sind. Diese Variable wird nicht nur für diese Arten von Grafiken verwendet, sondern auch für viele andere Statistikanwendungen, die wir in Kapitel 18 vorstellen. Tatsächlich ist es so, dass wir die Darstellung des Histogramms auf später verschieben, nämlich bis wir zu diesem Kapitel kommen, da die Darstellung eines Histogramms die Gruppierung von Daten erfordert, und die Durchführung einer Frequenzanalyse vor der eigentlichen Darstellung. In diesem Abschnitt zeigen wir, wie Daten in die Variable ΣDAT geladen werden und wie Balken- und Punktdiagramme dargestellt werden.

Mit den folgenden Daten werden wir Balken- und Punktdiagramme darstellen:

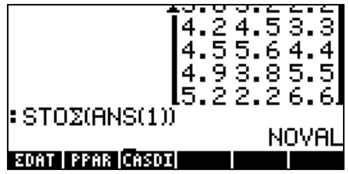
x	y	z
3.1	2.1	1.1
3.6	3.2	2.2
4.2	4.5	3.3
4.5	5.6	4.4
4.9	3.8	5.5
5.2	2.2	6.6

Balkendiagramme

Stellen Sie zunächst sicher, dass das CAS Ihres Taschenrechners im Modus `Exact` ist. Danach geben Sie die obengenannten Daten als Matrix ein, d. h.

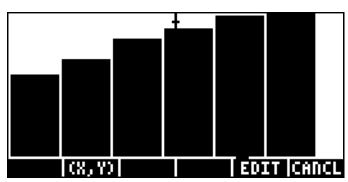
[[3,1;2,1;1,1],[3,6;3,2;2,2],[4,2;4,5;3,3],
[4,5;5,6;4,4],[4,9;3,8;5,5],[5,2;2,2;6,6]] ENTER

und legen Sie diese in ΣDAT ab. Verwenden Sie dazu die Funktion STOΣ (aus dem Funktionskatalog ↱ CAT) Drücken Sie VAR, um zum Variablenmenü zurückzukehren. Im Stack sollte ein Softkey mit der Bezeichnung ΣDAT zur Verfügung stehen. Untenstehende Abbildung zeigt die Speicherung dieser Matrix im ALG-Modus:



Erzeugung der Grafik:

- Drücken Sie ← 2D/3D , gleichzeitig wenn im RPN-Modus, um zum Fenster PLOT SETUP zu gelangen.
- Ändern Sie `TYPE` auf `Bar`.
- Im Feld ΣDAT wird eine Matrix angezeigt. Das ist die Matrix, die wir zuvor unter ΣDAT abgelegt haben.
- Markieren Sie das Feld `Col:` . Mit diesem Feld können Sie die Spalte aus ΣDAT wählen, die dargestellt werden soll. Der voreingestellte Standardwert ist 1. Lassen Sie diesen stehen, um die Spalte 1 in ΣDAT darzustellen..
- Drücken Sie NXT OK, um zur normalen Anzeige zurückzukehren.
- Drücken Sie ← WIN , gleichzeitig wenn im RPN-Modus, um zum Fenster PLOT WINDOW zu gelangen.
- Ändern Sie die V-VIEW folgendermaßen: `V-View: 0 5`.
- Drücken Sie ERASE DRAW, um das Balkendiagramm zu zeichnen.



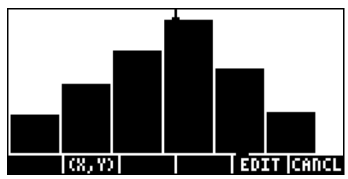
- Drücken Sie CANCL um wieder in das Fenster PLOT WINDOW zu gelangen. Dann drücken Sie ON , oder NXT OK, um zur normalen Taschenrechnerdisplay zurückzukehren.

Die Zahl der darzustellenden Balken bestimmt die Breite der Balken. H- und V-VIEW sind standardmäßig auf 10 voreingestellt. Wir haben V-VIEW geändert, um den maximumwert der Spalte 1 aus ΣDAT besser unterzubringen.

Balkendiagramme sind hilfreich bei der Darstellung kategorischer (d. h. nichtnumerischer) Daten.

Nehmen wir an, wir möchten die Daten aus Spalte 2 der ΣDAT-Matrix darstellen:

- Drücken Sie [←] 2D/3D , gleichzeitig wenn im RPN-Modus, um zum Fenster PLOT SETUP zu gelangen.
- Drücken Sie [▼][▼], um das Feld `Col:` hervorzuheben und drücken Sie 2 OK, und anschließend [NXT] OK.
- Drücken Sie [←] WIN , gleichzeitig wenn im RPN-Modus, um zum Fenster PLOT SETUP zu gelangen.
- Ändern Sie die V-VIEW folgendermaßen: `V-View: 0 6`
- Drücken Sie ERASE DRAW.



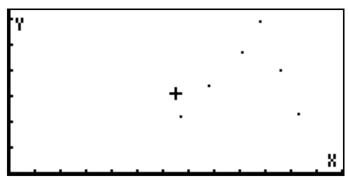
- Drücken Sie CANCL, um zum Fenster PLOT WINDOW zurückzukehren, und anschließend [ON], um zum normalen Taschenrechner-Display zurückzukehren.

Punktdiagramme

Wir verwenden dieselbe ΣDAT-Matrix, um Punktdiagramme zu erzeugen. Zunächst werden wir die Werte von y gegen x darstellen, und dann die von y gegen z, und zwar wie folgt:

- Drücken Sie [←] 2D/3D , gleichzeitig wenn im RPN-Modus, um zum Fenster PLOT SETUP zu gelangen.
- Ändern Sie `TYPE` auf `Scatter`.
- Drücken Sie [▼][▼] um das Feld **Cols**: hervorzuheben. Geben Sie [1] OK [2] OK ein, um die Spalte 1 für X und die Spalte 2 für Y für das Punktdiagramm Y gegen X auszuwählen.

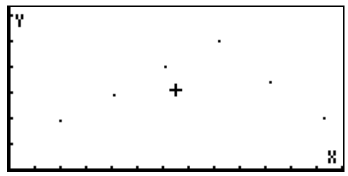
- Drücken Sie NXT OK um zur normalen Anzeige zurückzukehren.
- Drücken Sie ← WIN, gleichzeitig wenn im RPN-Modus, um zum Fenster PLOT WINDOW zu gelangen.
- Ändern Sie den Grafikfensterbereich folgendermaßen: H-View: 0 6, V-View: 0 6.
- Drücken Sie ERASE DRAW um das Balkendiagramm zu zeichnen. Drücken Sie EDIT NXT LABEL MENU um die Grafik ohne Menü im Vollbild und mit Bezeichnern zu sehen (Der Cursor bleibt jedoch in der Mitte der Grafik):



- Drücken Sie NXT NXT PICT um den Bereich EDIT zu verlassen.
- Drücken Sie CANCL um wieder in das Fenster PLOT WINDOW zu gelangen. Drücken Sie anschließend ON oder NXT OK, um zur normalen Taschenrechnerdisplay zurückzukehren.

Zur Darstellung von y gegen z:

- Drücken Sie ← 2D/3D, gleichzeitig wenn im RPN-Modus, um zum Fenster PLOT SETUP zu gelangen.
- Drücken Sie ▼▼ um das Feld **Cols**: hervorzuheben. Geben Sie 3 OK 2 OK ein, um die Spalte 3 für X und die Spalte 2 für Y für den Scatterplot Y gegen X auszuwählen.
- Drücken Sie NXT OK um zur normalen Anzeige zurückzukehren.
- Drücken Sie ← WIN, gleichzeitig wenn im RPN-Modus, um zum Fenster PLOT WINDOW zu gelangen.
- Ändern Sie den Grafikfensterbereich folgendermaßen: H-View: 0 7, V-View: 0 7.
- Drücken Sie ERASE DRAW um das Balkendiagramm zu zeichnen. Drücken Sie EDIT NXT LABEL MENU um die Grafik im Vollbild, ohne Menü und mit Bezeichnern zu sehen.



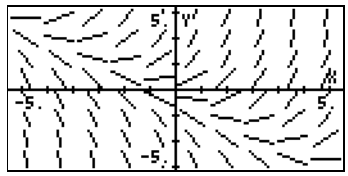
- Drücken Sie NXT NXT PICT, um den Bereich EDIT zu verlassen.
- Drücken Sie CANCL, um wieder in das Fenster PLOT WINDOW zu gelangen. Drücken Sie anschließend ON, oder NXT OK, um zur normalen Taschenrechnerdisplay zurückzukehren.

Steigungsfelder

Steigungsfelder werden dazu verwendet, Lösungen für Differentialgleichungen der Form y' = f(x,y) zu veranschaulichen. Im Grunde genommen zeigt die Grafik Segmente, die tangential zur Lösungskurve liegen, da y' = dy/dx, an jedem beliebigen Punkt (x,y), gemessen die Steigung der Tangente am Punkt (x,y) darstellt.

Um beispielsweise die Lösung zur Differentialgleichung y' = f(x,y) = x+y, zu veranschaulichen, gehen Sie folgendermaßen vor:

- Drücken Sie ← 2D/3D, gleichzeitig wenn im RPN-Modus, um zum Fenster PLOT SETUP zu gelangen.
- Ändern Sie `TYPE` auf `Slopefield`.
- Drücken Sie ▼ und geben Sie 'X+Y' OK ein.
- Stellen Sie sicher, dass bei `Indep:` die Variable 'X' gewählt ist und bei `Depnd:` 'Y' .
- Drücken Sie NXT OK, um zur normalen Anzeige zurückzukehren.
- Drücken Sie ← WIN, gleichzeitig wenn im RPN-Modus, um zum Fenster PLOT WINDOW zu gelangen.
- Ändern Sie den Grafikfensterbereich folgendermaßen: X-Left:-5, X-Right:5, Y-Near:-5, Y-Far: 5
- Drücken Sie ERASE DRAW, um die Steigungsfelddarstellung zu zeichnen. Drücken Sie EDIT NXT LABEL MENU, um die Grafik im Vollbild, ohne Menü und mit Bezeichnern zu sehen.



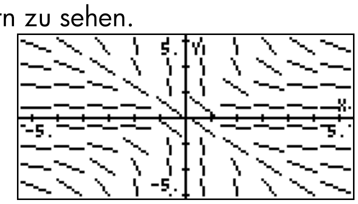
- Drücken Sie NXT NXT PICT, um den Bereich EDIT zu verlassen.

- Drücken Sie CANCL um wieder in das Fenster PLOT WINDOW zu gelangen. Dann drücken Sie ON, oder NXT OK, um zur normalen Taschenrechnerdisplay zurückzukehren.

Wenn Sie die Darstellung des Steigungsfeldes auf Papier bringen, dann können Sie von Hand Linien nachvollziehen, die tangential zu den Liniensegmenten in der Darstellung liegen. Diese Linien bilden Linien von y(x,y) = Konstante für die Lösung von y' = f(x,y). Deshalb sind Steigungsfelder nützliche Werkzeuge für die Veranschaulichung besonders schwieriger Gleichungen.

Versuchen Sie auch ein Steigungsfeld für die Funktion $y' = f(x,y) = -(y/x)^2$ darzustellen, und zwar wie folgt:

- Drücken Sie ← 2D/3D, gleichzeitig wenn im RPN-Modus, um zum Fenster PLOT SETUP zu gelangen.
- Ändern Sie `TYPE` auf `Slopefield`.
- Drücken Sie ▼ und geben '– (Y/X)^2' OK ein.
- Drücken Sie ERASE DRAW um die Steigungsfelddarstellung zu zeichnen. Drücken Sie EDIT NXT LABEL MENU um die Grafik im Vollbild, ohne Menü und mit Bezeichnern zu sehen.



- Drücken Sie NXT NXT PICT um den Bereich EDIT zu verlassen.

- Drücken Sie CANCL um wieder zum Fenster PLOT WINDOW zu gelangen. Dann drücken Sie ON, oder NXT OK, um zur normalen Taschenrechnerdisplay zurückzukehren.

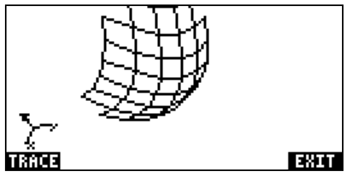
Fast 3D-Darstellung

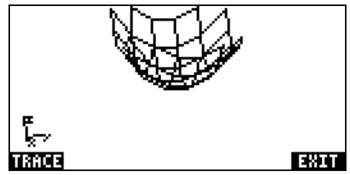
Die Fast 3D-Darstellung wird dazu verwendet, dreidimensionale Oberflächen, die durch Gleichungen der Form z = f(x,y) dargestellt werden, zu veranschaulichen. Wenn Sie beispielsweise $z = f(x,y) = x^2+y^2$ veranschaulichen möchten, dann können Sie folgendermaßen vorgehen:

- Drücken Sie ← 2D/3D, gleichzeitig wenn im RPN-Modus, um zum Fenster PLOT SETUP zu gelangen.
- Ändern Sie TYPE auf Fast3D.
- Drücken Sie ▼ und geben 'X^2+Y^2' OK ein.
- Stellen Sie sicher, dass bei Indep: die Variable 'X' gewählt ist und bei Depnd: 'Y' .
- Drücken Sie NXT OK um zur normalen Anzeige zurückzukehren.
- Drücken Sie ← WIN, gleichzeitig wenn im RPN-Modus, um zum Fenster PLOT WINDOW zu gelangen.
- Behalten Sie den voreingestellten Grafikfensterbereich bei, und zwar: X-Left:-1, X-Right:1, Y-Near:-1, Y-Far: 1, Z-Low: -1, Z-High: 1, Step Indep: 10, Depnd: 8

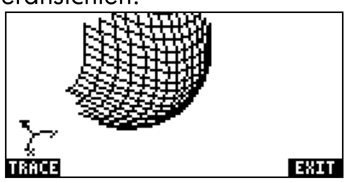
Hinweis: Die Step Indep: und Depnd: -Werte stellen die Zahl der Rasterlinien dar, die in der Grafik verwendet werden. Je größer diese Zahl, desto langsamer geht die Erstellung der Grafik voran. Und doch ist die für die Grafikerzeugung benötigte Zeit relativ kurz. Zunächst behalten wir jedoch die voreingestellten Standardwerte von 10 und 8 für „Step" bei.

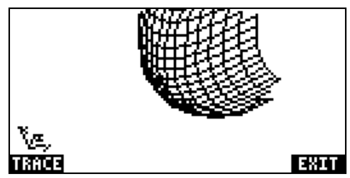
- Drücken Sie ERASE DRAW um die dreidimensionale Oberfläche zu zeichnen. Das Ergebnis ist die Drahtgitterdarstellung einer Oberfläche mit Referenz-Koordinatensystem in der unteren Bildschirmecke links. Mit den Pfeiltasten (◄►▲▼) können Sie die Ausrichtung der Oberfläche verändern. Die Ausrichtung des Referenz-Koordinatensystems ändert sich entsprechend. Versuchen Sie die Oberflächenausrichtung selbst zu ändern. Die folgenden Abbildungen zeigen einige Ansichten der Grafik:





- Wenn Sie fertig sind, drücken Sie **EXIT**.
- Drücken Sie **CANCL** um wieder in das Fenster PLOT WINDOW zu gelangen.
- Ändern Sie die „Step"-Daten folgendermaßen ab: `Step Indep: 20 Depnd: 16`
- Drücken Sie **ERASE** **DRAW** um das Zeichnen der Oberfläche zu verfolgen. Musteransichten:





- Wenn Sie fertig sind, drücken Sie **EXIT**.
- Drücken Sie **CANCL** um wieder zum Fenster PLOT WINDOW zu gelangen.
- Drücken Sie ON, oder NXT **OK**, um zur normalen Taschenrechnerdisplay zurückzukehren.

Versuchen Sie auch, eine Fast 3D-Darstellung für die Oberfläche z = f(x,y) = sin (x^2+y^2) zu erstellen.

- Drücken Sie ← 2D/3D, gleichzeitig wenn im RPN-Modus, um zum Fenster PLOT SETUP zu gelangen.
- Drücken Sie ▼ und geben Sie 'SIN(X^2+Y^2)' **OK** ein.
- Drücken Sie **ERASE** **DRAW** zeichnen diagramm.
- Wenn Sie getan werden, drücken Sie **EXIT**.
- Drücken Sie **CANCL** zu zurückgehen PLOT WINDOW.
- Drücken Sie ON, oder NXT **OK**, zur normalen Rechneranzeige zurückgehen.

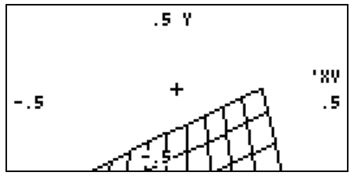
Drahtgitterdarstellung

Drahtgitter-Darstellungen sind Grafiken dreidimensionaler Oberflächen, die folgendermaßen beschrieben werden z = f(x,y). Im Gegensatz zu *Fast-3D-Darstellungen* sind Drahtgitterdarstellungen feststehende Grafiken. Der Benutzer kann den Blickwinkel für die Darstellung wählen, d. h. von welchem Punkt aus die Oberfläche betrachtet wird. Um beispielsweise eine Drahtgitterdarstellung für die Oberfläche z = x + 2y –3 zu erzeugen, gehen Sie wie folgt vor:

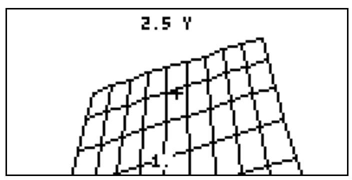
- Drücken Sie ⬅ 2D/3D, gleichzeitig wenn im RPN-Modus, um zum Fenster PLOT SETUP zu gelangen.
- Ändern Sie `TYPE` auf `Wireframe`.
- Drücken Sie ▼ und geben Sie 'X+2*Y-3' OK ein.
- Stellen Sie sicher, dass bei `Indep:` die Variable 'X' gewählt ist und bei `Depnd:` 'Y' .
- Drücken Sie NXT OK um zur normalen Anzeige zurückzukehren.
- Drücken Sie ⬅ WIN, gleichzeitig wenn im RPN-Modus, um zum Fenster PLOT WINDOW zu gelangen.
- Behalten Sie den voreingestellten Grafikfensterbereich bei, und zwar: X-Left:-1, X-Right:1, Y-Near:-1, Y-Far: 1, Z-Low: -1, Z-High: 1, XE:0,YE:-3, ZE:0, Step Indep: 10 Depnd: 8

Die Koordinaten XE, YE und ZE stehen für die „Augenkoordinaten“, d. h. die Koordinaten, von denen aus der Betrachter die Darstellung sieht. Die gezeigten Werte sind die voreingestellten Standardwerte. Die Step Indep: und Depnd: -Werte stellen die Zahl der Rasterlinien dar, die in der Grafik verwendet werden. Je größer diese Zahl desto langsamer geht die Erzeugung der Grafik. Zunächst behalten wir jedoch die voreingestellten Standardwerte von 10 und 8 für „Step“ bei.

- Drücken Sie ERASE DRAW um die dreidimensionale Oberfläche zu zeichnen. Das Ergebnis ist ein Drahtgitterbild der Oberfläche.
- Drücken Sie EDIT NXT LABEL MENU um die Grafik mit Bezeichnern und Bereichen anzuzeigen. Diese Version der Grafik beschränkt sich auf den unteren Teil der Anzeige. Wir können den Blickwinkel ändern, um eine andere Version der Grafik zu sehen.

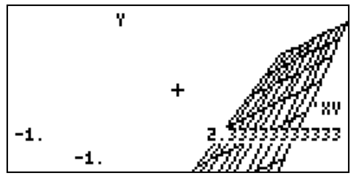


- Drücken Sie NXT NXT PICT CANCL um wieder in den Bereich PLOT WINDOW zu gelangen.
- Ändern Sie die Augenkoordinaten folgendermaßen ab: `XE:0 YE:-3 ZE:3`
- Drücken Sie ERASE DRAW um das Zeichnen der Oberfläche zu verfolgen.
- Drücken Sie EDIT NXT LABEL MENU um die Grafik mit Bezeichnern und Bereichen anzuzeigen.



Diese Version der Grafik beansprucht mehr Raum im Display als die vorherige. Wir können den Blickwinkel noch einmal ändern, um eine andere Version der Grafik zu sehen.

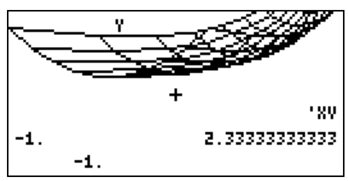
- Drücken Sie NXT NXT PICT CANCL um wieder in den Bereich PLOT WINDOW zu gelangen.
- Ändern Sie die Augenkoordinaten folgendermaßen ab: `XE:3 YE:3 ZE:3`
- Drücken Sie ERASE DRAW um das Zeichnen der Oberfläche zu verfolgen. Dieses Mal befindet sich der Hauptteil der Grafik auf der rechten Seite des Displays.



- Drücken Sie CANCL um wieder in das Fenster PLOT WINDOW zu gelangen.
- Drücken Sie ON, oder NXT OK, um zur normalen Taschenrechnerdisplay zurückzukehren.

Versuchen Sie auch, eine Drahtgitterdarstellung für die Oberfläche z = f(x,y) = sin x^2+y^2 zu erstellen.

- Drücken Sie ↰ 2D/3D , gleichzeitig wenn im RPN-Modus, um zum Fenster PLOT SETUP zu gelangen.
- Drücken Sie ▼ und geben 'X^2+Y^2' OK ein.
- Drücken Sie ERASE DRAW um die Steigungsfelddarstellung zu zeichnen. Drücken Sie EDIT NXT MENU LABEL um die Grafik im Vollbild ohne Menü und mit Bezeichnern zu sehen.

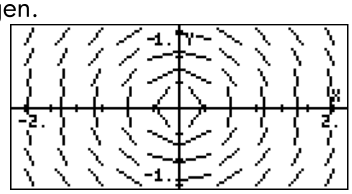


- Drücken Sie NXT NXT PICT um den Bereich EDIT zu verlassen.
- Drücken Sie CANCL um wieder in das Fenster PLOT WINDOW zu gelangen. Drücken Sie anschließend ON, oder NXT OK, um zur normalen Taschenrechnerdisplay zurückzukehren.

Ps-Contour-Darstellungen

Ps-Contour-Darstellungen sind Konturdarstellungen dreidimensionaler Oberflächen, die folgendermaßen beschrieben werden: z = f(x,y). Die erzeugten Konturen sind Projektionen von ebenen Oberflächen z = Konstante auf der xy-Ebene. Um beispielsweise eine PS-Contour-Darstellung für die Oberfläche $z = x^2+y^2$ zu erzeugen, gehen Sie wie folgt vor:

- Drücken Sie ← 2D/3D , gleichzeitig wenn im RPN-Modus, um zum Fenster PLOT SETUP zu gelangen.
- Ändern Sie TYPE auf Ps-Contour.
- Drücken Sie ▼ und geben 'X^2+Y^2' OK ein.
- Stellen Sie sicher, dass bei `Indep:` die Variable 'X' gewählt ist und bei `Depnd:` 'Y' .
- Drücken Sie NXT OK um zur normalen Anzeige zurückzukehren.
- Drücken Sie ← WIN , gleichzeitig wenn im RPN-Modus, um zum Fenster PLOT WINDOW zu gelangen.
- Behalten Sie den voreingestellten Grafikfensterbereich bei, und zwar: X-Left:-2, X-Right:2, Y-Near:-1 Y-Far: 1, Step Indep: 10, Depnd: 8
- Drücken Sie ERASE DRAW um die Konturgrafik zu zeichnen. Dieser Vorgang dauert eine Weile. Haben sie bitte Geduld. Das Ergebnis ist eine Konturdarstellung der Oberfläche. Beachten Sie, dass die Konturen nicht unbedingt durchgehend sind, sie liefern jedoch ein gutes Bild der ebenen Oberflächen der Funktion.
- Drücken Sie EDIT NXT LABEL MENU um die Grafik mit Bezeichnern und Bereichen anzuzeigen.

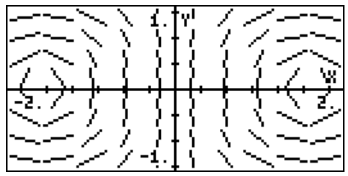


- Drücken Sie NXT NXT PICT CANCL um wieder in den Bereich PLOT WINDOW zu gelangen.
- Drücken Sie ON , oder NXT OK, um zur normalen Taschenrechnerdisplay zurückzukehren.

Versuchen Sie auch, eine Ps-Contour-Darstellung für die Oberfläche $z = f(x,y) = \sin x \cos y$ zu erstellen.

- Drücken Sie ← 2D/3D , gleichzeitig wenn im RPN-Modus, um zum Fenster PLOT SETUP zu gelangen.

- Drücken Sie ˅ und geben Sie 'SIN(X)*COS(Y)' OK ein.
- Drücken Sie ERASE DRAW, um die Steigungsfelddarstellung zu zeichnen. Drücken EDIT NXT LABEL MENU, um die Grafik im Vollbild, ohne Menü und mit Bezeichnern zu sehen.



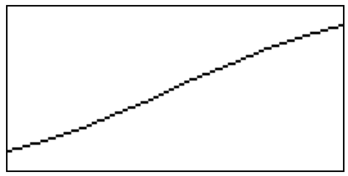
- Drücken Sie NXT NXT PICT um den Bereich EDIT zu verlassen.
- Drücken Sie CANCL um wieder in das Fenster PLOT WINDOW zu gelangen. Drücken Sie anschließend ON, oder NXT OK, um zur normalen Taschenrechnerdisplay zurückzukehren.

Y-Schnitt-Darstellungen

Y-Schnitt-Darstellungen sind animierte Darstellungen von z gegen y für verschiedene Werte von x aus der Funktion z = f(x,y). Um beispielsweise eine Y-Schnitt-Darstellung für die Oberfläche $z = x^3 - xy^3$ zu erzeugen, gehen Sie wie folgt vor:

- Drücken Sie ← 2D/3D, gleichzeitig wenn im RPN-Modus, um zum Fenster PLOT SETUP zu gelangen.
- Ändern Sie TYPE auf Y-Slice.
- Drücken Sie ˅ und geben 'X^3+X*Y^3' OK ein.
- Stellen Sie sicher, dass bei Indep: die Variable 'X' gewählt ist und bei Depnd: 'Y' .
- Drücken Sie NXT OK um zur normalen Anzeige zurückzukehren.
- Drücken Sie ← WIN, gleichzeitig wenn im RPN-Modus, um zum Fenster PLOT WINDOW zu gelangen.
- Behalten Sie den voreingestellten Grafikfensterbereich bei, und zwar: X-Left:-1, X-Right:1, Y-Near:-1, Y-Far: 1, Z-Low:-1, Z-High:1, Step Indep: 10 Depnd: 8
- Drücken Sie ERASE DRAW um die dreidimensionale Oberfläche zu zeichnen. Sie werden sehen, dass der Taschenrechner eine Reihe von Kurven auf dem Bildschirm erstellt, die sofort wieder verschwinden. Wenn der

Taschenrechner mit der Erstellung aller Y-Schnitt-Kurven fertig ist, dann geht er automatisch zur Animation der verschiedenen Kurven über. Eine Kurve sehen Sie unten.



- Drücken Sie ON um die Animation abzubrechen. Drücken Sie CANCL um wieder in das Fenster PLOT WINDOW zu gelangen.
- Drücken Sie ON, oder NXT OK, um zur normalen Taschenrechnerdisplay zurückzukehren.

Versuchen Sie auch, eine Ps-Contour-Darstellung für die Oberfläche z = f(x,y) = (x+y) sin y zu erstellen.

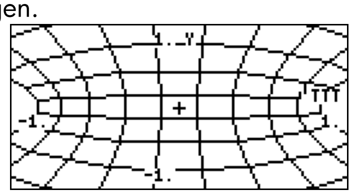
- Drücken Sie ← 2D/3D, gleichzeitig wenn im RPN-Modus, um zum Fenster PLOT SETUP zu gelangen.
- Drücken Sie ▼ und geben Sie '(X+Y)*SIN(Y)' OK ein.
- Drücken Sie ERASE DRAW um die Y-Schnitt-Animation zu erzeugen.
- Drücken Sie ON um die Animation abzubrechen.
- Drücken Sie CANCL um wieder in das Fenster PLOT WINDOW zu gelangen. Drücken Sie anschließend ON, oder NXT OK, um zur normalen Taschenrechnerdisplay zurückzukehren.

Netzbilddarstellungen

*Netzbild*darstellungen erzeugen ein Netz orthogonaler Kurven, die eine Funktion einer komplexen Größe der Form w =f(z) = f(x+iy) beschreiben, wobei z = x+iy eine komplexe Größe ist. Die dargestellten Funktionen entsprechen dem realen und imaginären Teil von w = $\Phi(x,y) + i\Psi(x,y)$, d. h. sie stellen die Kurven $\Phi(x,y)$ =Konstante und $\Psi(x,y)$ = Konstante dar. Um beispielsweise eine Netzbilddarstellung für die Funktion w = sin(z) zu erzeugen, gehen Sie wie folgt vor:

- Drücken Sie ← 2D/3D, gleichzeitig wenn im RPN-Modus, um zum Fenster PLOT SETUP zu gelangen.
- Ändern Sie TYPE auf Gridmap.

- Drücken Sie ▼ und geben Sie 'SIN(X+i*Y)' OK ein.
- Stellen Sie sicher, dass bei Indep: die Variable 'X' gewählt ist und bei Depnd: 'Y' .
- Drücken Sie NXT OK um zur normalen Anzeige zurückzukehren.
- Drücken Sie ← WIN , gleichzeitig wenn im RPN-Modus, um zum Fenster PLOT WINDOW zu gelangen.
- Behalten Sie den voreingestellten Grafikfensterbereich bei, und zwar: X-Left:-1, X-Right:1, Y-Near:-1 Y-Far: 1, XXLeft:-1 XXRight:1, YYNear:-1, yyFar: 1, Step Indep: 10 Depnd: 8
- Drücken Sie ERASE DRAW um die Netzbilddarstellung zu zeichnen. Das Ergebnis ist ein Netz von Funktionen, die den realen und imaginären Teilen einer komplexen Funktion entsprechen.
- Drücken Sie EDIT NXT LABEL MENU um die Grafik mit Bezeichnern und Bereichen anzuzeigen.



- Drücken Sie NXT NXT PICT CANCEL um wieder in den Bereich PLOT WINDOW zu gelangen.
- Drücken Sie ON , oder NXT OK, um zur normalen Taschenrechnerdisplay zurückzukehren.

Andere Funktionen einer komplexen Größe, die für eine Netzbilddarstellung geeignet sind, sind:

(1) SIN((X,Y))	d. h. $F(z) = \sin(z)$	(2)(X,Y)^2	d. h. $F(z) = z^2$
(3) EXP((X,Y))	d. h. $F(z) = e^z$	(4) SINH((X,Y))	d. h. $F(z) = \sinh(z)$
(5) TAN((X,Y))	d. h. $F(z) = \tan(z)$	(6) ATAN((X,Y))	d. h. $F(z) = \tan^{-1}(z)$
(7)(X,Y)^3	d. h. $F(z) = z^3$	(8) 1/(X,Y)	d. h. $F(z) = 1/z$
(9) √ (X,Y)	d. h. $F(z) = z^{1/2}$		

Pr-Oberflächendarstellungen

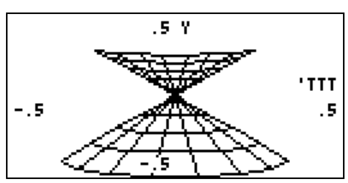
Pr-Oberflächen-Darstellungen (parametrische Oberflächen) werden für dreidimensionale Oberflächen verwendet, deren Koordinaten (x,y,z) mit x =

x(X,Y), y = y(X,Y), z=z(X,Y) beschrieben werden, wobei X und Y unabhängige Parameter sind.

Hinweis: Die Gleichungen x = x(X,Y), y = y(X,Y) und z=z(X,Y) stellen die parametrische Beschreibung einer Oberfläche dar. X und Y sind unabhängige Parameter. In den meisten Lehrbüchern werden u und v als Parameter verwendet und nicht X und Y. Deshalb wird die parametrische Beschreibung einer Oberfläche mit x = x(u,v), y = y(u,v), z=z(u,v) angegeben.

Wenn beispielsweise eine Pr-Oberflächendarstellung für die Oberfläche x = x(X,Y) = X sin Y, y = y(X,Y) = x cos Y, z=z(X,Y)=X, erzeugt werden soll, dann gehen Sie folgendermaßen vor:

- Drücken Sie ← 2D/3D , gleichzeitig wenn im RPN-Modus, um zum Fenster PLOT SETUP zu gelangen.
- Ändern Sie TYPE auf Pr-Surface.
- Drücken Sie ▼ und geben Sie '{X*SIN(Y), X*COS(Y), X}' OK ein.
- Stellen Sie sicher, dass bei Indep: die Variable 'X' gewählt ist und 'Y' bei Depnd: .
- Drücken Sie NXT OK um zur normalen Anzeige zurückzukehren.
- Drücken Sie ← WIN , gleichzeitig wenn im RPN-Modus, um zum Fenster PLOT WINDOW zu gelangen.
- Behalten Sie den voreingestellten Grafikfensterbereich bei, und zwar: X-Left:-1, X-Right:1, Y-Near:-1, Y-Far: 1, Z-Low: -1, Z-High:1, XE: 0, YE:-3, zE:0, Step Indep: 10, Depnd: 8
- Drücken Sie ERASE DRAW um die dreidimensionale Oberfläche zu zeichnen.
- Drücken Sie EDIT NXT LABEL MENU um die Grafik mit Bezeichnern und Bereichen anzuzeigen.



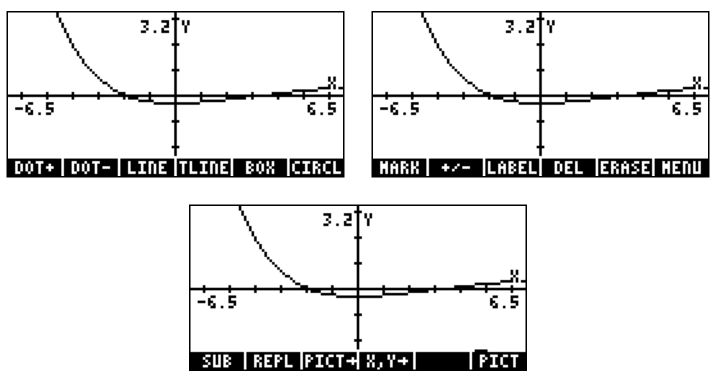
- Drücken Sie NXT NXT PICT CANCL um wieder in den Bereich PLOT WINDOW zu gelangen.
- Drücken Sie ON, oder NXT OK, um zur normalen Taschenrechnerdisplay zurückzukehren.

Die VPAR-Variable

Die Variable VPAR (Volumenparameter) enthält Informationen bezüglich des „volumens“, das benötigt wird, um eine dreidimensionale Grafik zu erzeugen. Deshalb wird sie immer erzeugt, wenn Sie eine dreidimensionale Darstellung wie z.B. Fast3D, Wireframe (Drahtgitter) oder Pr-Surface (Pr-Oberfläche) erstellen.

Interaktives Zeichnen

Immer wenn wir eine zweidimensionale Grafik erzeugen, erscheint im Grafikbildschirm ein Softkey mit der Bezeichnung EDIT . Wenn Sie auf EDIT drücken, erscheint ein Menü, das folgende Optionen bereit hält (bezüglich weiterer Funktionen drücken Sie NXT):



Mit dem Beispiel oben können Sie die Funktionen LABEL, MENU, PICT→, und REPL ausprobieren. Viele der restlichen Funktionen, z.B. DOT+, DOT-, LINE, BOX, CIRCL, MARK, DEL usw. können dazu verwendet werden, Punkte, Linien, Kreise usw. auf dem Grafikbildschirm zu zeichnen, und zwar gemäß der untenstehenden Beschreibung. Damit Sie sehen, wie diese Funktionen verwendet werden, machen wir folgende Übung:

Zunächst gehen wir in den Grafikbildschirm, und zwar gemäß der folgenden Anweisungen:

- Drücken Sie ⬅ 2D/3D, gleichzeitig wenn im RPN-Modus, um zum Fenster PLOT SETUP zu gelangen.
- Ändern Sie `TYPE` auf `Function` falls erforderlich.
- Ändern Sie EQ auf 'X'
- Stellen Sie sicher, dass Indep: auch auf 'X" eingestellt ist.
- Drücken Sie NXT OK um zur normalen Anzeige zurückzukehren.
- Drücken Sie ⬅ WIN gleichzeitig, wenn im RPN-Modus, um zum PLOT-Fenster zu gelangen (in diesem Fall heißt es PLOT –POLAR).
- Ändern Sie den Bereich von H-VIEW auf –10 bis 10 durch Drücken von 1 0 +/- OK 1 0 OK und den Bereich von V-VIEW auf -5 bis 5 durch Drücken von 5 +/- OK 5 OK.
- Drücken Sie ERASE DRAW um die Funktion darzustellen.
- Drücken Sie EDIT NXT LABEL um der Grafik Bezeichner hinzuzufügen. Drücken Sie NXT NXT (oder ⬅ PREV) um zum ursprünglichen EDIT-Menü zurückzukehren.

Jetzt zeigen wir die Verwendung der verschiedenen Zeichenfunktionen am fertigen Grafikbildschirm. Dazu verwendet man den Cursor und die Pfeiltasten (◀ ▶ ▲ ▼) um den Cursor auf dem Grafikbildschirm zu bewegen.

DOT+ und DOT-

Wenn die Funktion DOT+ gewählt ist, dann werden überall dort, wohin der Cursor bewegt wird, Pixel aktiviert, und er hinterlässt so eine Spur. Wenn DOT- gewählt wird, ist das Gegenteil der Fall, d. h. der Cursor löscht die Pixel auf seinem Weg.

Sie können den Cursor beispielsweise mit den ▶ ▲ Tasten irgendwo in die Mitte des ersten Quadranten der xy-Ebene bewegen und dann DOT+ drücken. Der Bezeichner wird ausgewählt (DOT+■). Drücken Sie die ▶ Taste, um eine horizontale Linie zu verfolgen. Jetzt drücken Sie DOT-, um diese Option auszuwählen (DOT-■). Halten Sie die Taste ◀ gedrückt, um zu sehen wie die Linie, die sie eben verfolgt haben, ausradiert wird. Wenn das abgeschlossen ist, dann drücken Sie DOT-, um diese Option abzuwählen.

MARK

Mit diesem Befehl kann der Benutzer einen Markierungspunkt setzen, der vielfältig verwendet werden kann, zum Beispiel als:

- Linienanfang für den Befehl LINE oder TLINE,
- Eckpunkt für den Befehl BOX (Rechteck),
- Mittelpunkt für den Befehl CIRCLE (Kreis).

Wenn nur der Befehl MARK verwendet wird, dann erscheint an der markierten Stelle ein x (Kreuzchen). Drücken Sie NXT MARK zur Demonstration.

LINE

Dieser Befehl wird zum Ziehen einer Linie zwischen zwei Punkten in der Grafik verwendet. Zur Demonstration setzen Sie den Cursor irgendwo in den ersten Quadranten und drücken ← PREV LINE . Auf dem Cursor erscheint eine Markierung, die den Anfang der Linie kennzeichnet. Mit der Taste ▶ bewegen Sie den Cursor von der aktuellen Position aus nach rechts, sagen wir etwa 1 cm, und drücken dann LINE. Dann wird eine Linie zwischen dem Anfangs- und Schlusspunkt gezogen.

Beachten Sie, dass der Cursor am Schlusspunkt dieser Linie immer noch aktiv ist, und der Taschenrechner damit bereit ist, eine neue Linie zu ziehen, und zwar mit diesem Punkt als Anfangspunkt. Drücken Sie ▼ um den Cursor nach unten zu bewegen, sagen wir um etwa 1 cm, und drücken dann noch einmal LINE. Jetzt sollten Sie einen gestreckten Winkel haben, der von einem horizontalen und einem vertikalen Segment gezeichnet ist. Der Cursor ist immer noch aktiv. Um ihn zu deaktivieren, drücken Sie LINE ohne ihn zu bewegen. Der Cursor erhält wieder seine normale Form (ein Kreuz) und die Funktion LINE ist nicht mehr aktiv.

TLINE

(Toggle LINE) Bringen Sie den Cursor zur Demonstration in den zweiten Quadranten. Drücken Sie TLINE. Eine Markierung erscheint am Anfangspunkt der Linie. Bewegen Sie den Cursor mit den Pfeiltasten von diesem Punkt weg und drücken dann TLINE . Eine Linie wird von der derzeitigen Cursorposition zum zuerst gewählten Bezugspunkt gezogen. Nun erscheint eine Linie, die konträr zum Hintergrund ist, d. h. Pixel, die zuvor aktiviert waren, werden jetzt deaktiviert bzw. andersherum. Um die letzte gezogene Line zu entfernen, drücken Sie noch einmal TLINE . Um die Funktion TLINE zu deaktivieren, bewegen Sie den Cursor an den Punkt zurück, an dem TLINE aktiviert worden ist, und drücken dann LINE LINE.

BOX

Dieser Befehl wird zum Zeichnen eines Rechteckes in der Grafik verwendet. Bewegen Sie den Cursor zu einer freien Fläche der Grafik und drücken BOX. Damit wird der Cursor hervorgehoben. Bewegen Sie den Cursor mit den Pfeiltasten in diagonaler Richtung zu einem anderen Punkt. Drücken Sie noch einmal BOX. Es erscheint ein Rechteck, dessen Diagonale die Anfangs- und Schlussposition des Cursors verbindet. Die Anfangsposition des Rechtecks ist immer noch mit einem x markiert. Wenn der Cursor nun zu einer anderen Stelle hin bewegt wird und BOX gedrückt wird, wird ein neues Rechteck mit diesem Punkt als Anfangspunkt erzeugt. Um die Funktion BOX zu deaktivieren, bewegen Sie den Cursor an den Punkt zurück, an dem BOX aktiviert worden ist, und drücken Sie dann LINE LINE.

CIRCL

Dieser Befehl erzeugt einen Kreis. Markieren Sie den Mittelpunkt des Kreises mit dem Befehl MARK und bewegen den Cursor dann zu einem Punkt, der Teil des Kreisumfangs ist. Drücken Sie anschließend CIRCL. Um die Funktion CIRCL zu deaktivieren, bringen Sie den Cursor an die Stelle zurück, an der die Markierung gesetzt wurde, und drücken dann LINE.

Probieren Sie diesen Befehl aus, indem Sie den Cursor in einen freien Teil der Grafik bringen und drücken dann MARK. Bewegen Sie den Cursor dann zu einem anderen Punkt und drücken CIRCL . Es erscheint ein Kreis, dessen Mittelpunkt auf der Markierung MARK liegt, und dessen Umfang die letzte Cursorposition durchläuft.

LABEL

Durch Drücken von LABEL setzen Sie Bezeichner an der x- und y-Achse der aktuellen Grafik. Diese Funktion wurde bereits ausführlich in diesem Kapitel besprochen.

DEL

Dieser Befehl wird dazu verwendet, einen Teil der Grafik, der zwischen zwei Markierungen (MARK) liegt, zu entfernen. Bewegen Sie den Cursor zu einem Punkt in der Grafik und drücken dann MARK. Bewegen Sie den Cursor dann zu einem anderen Punkt und drücken dann noch einmal MARK . Drücken Sie anschließend DEL. Der Teil der Grafik, der in dem Rechteck zwischen den beiden Markierungen liegt, wird gelöscht.

ERASE

Mit der Funktion ERASE löschen Sie das gesamte Grafikfenster. Dieser Befehl steht im PLOT-Menü zur Verfügung, und auch in den PLOT-Fenstern, zu denen man über die Softkeys gelangt.

MENU

Durch Drücken auf MENU werden die Softkey-Bezeichner entfernt, und die Grafik wird ohne diese angezeigt. Zum Wiederherstellen der Bezeichner drücken Sie NXT.

SUB

Mit diesem Befehl können Sie einen Teilbereich eines Grafikobjekts extrahieren. Das extrahierte Objekt wird automatisch im Stack abgelegt. Wählen Sie den Teilbereich aus, den Sie extrahieren möchten, und zwar durch Markieren (mit MARK) eines Punktes der Grafik und eines zweiten Punktes, der diagonal zum ersten Punkt liegt, sodass das entstehende Rechteck den gewünschten Teilbereich der Grafik einschließt. Drücken Sie anschließend SUB. Diese Funktion kann dazu verwendet werden, Teile einer Grafik innerhalb der Grafik zu bewegen.

REPL

Mit diesem Befehl kann der Inhalt eines Grafikobjekts, das sich in Ebene 1 des Stack befindet, an der Cursorposition im Grafikfenster eingefügt werden. Die linke obere Ecke des einzufügenden Grafikobjekts kommt dann an die Cursorposition. Wenn Sie also eine Grafik aus dem Stack in das Grafikfenster einfügen möchten, sodass das gesamte Fenster ausgefüllt wird, sollten Sie sich vergewissern, dass der Cursor ganz in der oberen linken Ecke des Displays steht.

PICT→

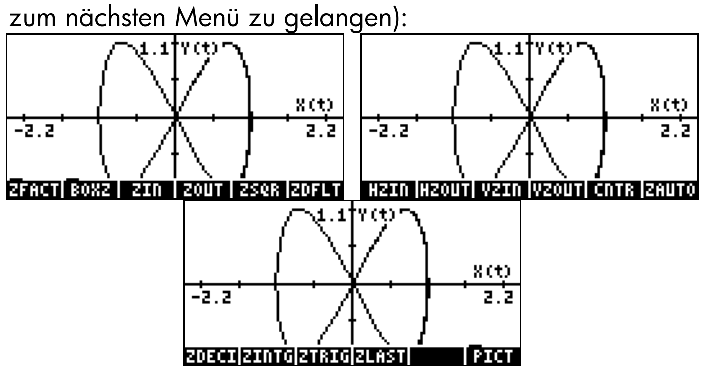
Mit diesem Befehl können Sie eine Kopie der aktuellen Grafik im Grafikfenster als Grafikobjekt im Stack ablegen. Das Grafikobjekt, das im Stack abgelegt wird, kann für jede Art der Verwendung mit einem beliebigen Namen gespeichert werden.

X,Y→

Mit diesem Befehl kopieren Sie die Koordinaten der aktuellen Cursorposition in den Stack, und zwar als Benutzerkoordinaten.

Vergrößern und verkleinern im Grafikfenster (Zoomen)

Immer wenn Sie interaktiv eine zweidimensionale Funktionsgrafik erstellen, können Sie mit dem ersten Softkey, der die Bezeichnung ZOOM trägt, auf Funktionen zugreifen, mit denen Sie Teile der aktuellen Grafik vergrößern und verkleinern können. Das Menü ZOOM enthält folgende Funktionen (Drücken NXT, um zum nächsten Menü zu gelangen):



Wir stellen im Folgenden jede dieser Funktionen vor. Wir müssen lediglich eine Grafik gemäß den Anweisungen in Kapitel 12 erzeugen, oder mit einem der Programme, die an früherer Stelle in diesem Kapitel behandelt worden sind.

ZFACT, ZIN, ZOUT und ZLAST

Nach Drücken von ZFACT erscheint eine Eingabemaske, mit der Sie die aktuellen X- und Y-Faktoren ändern können. Die X- und Y-Faktoren beziehen sich auf die horizontalen und vertikalen benutzerdefinierten Bereichseinheiten der entsprechenden Pixelbereiche. Ändern Sie den H-Faktor auf 8. und drücken Sie OK, dann ändern Sie den V-Faktor auf 2. und drücken OK. Entfernen Sie das Häkchen der Option „✓Recenter on cursor“ (= neu auf Cursor zentrieren) und drücken Sie OK.

Zurück im Grafikdisplay drücken Sie ZIN . Die Grafik wird neu gezeichnet, und zwar mit dem neuen vertikalen und horizontalen Einteilungsfaktor, mit der Cursorposition als Mittelpunkt, während die originale Bildgröße erhalten bleibt (d. h. die ursprüngliche Pixelzahl in beiden Richtungen). Mit den Pfeiltasten können Sie sich so weit die vergrößerte Grafik reicht umherbewegen.

Zum Verkleinern, doch abhängig von den unter ZFACT eingestellten H- und V-Faktoren, drücken Sie ZOOM ZOUT. Die daraus resultierende Grafik enthält mehr Details als die vergrößerte Grafik. Es ist jederzeit möglich, zum jeweils letzten Zoomfenster zurückzukehren, und zwar über ZLAST.

BOXZ

Das Vergrößern und Verkleinern einer bestimmten Grafik kann auch über den Softkey BOXZ erfolgen. Mit BOXZ können Sie einen rechtwinkligen Bereich („Box“) auswählen, den Sie vergrößern möchten. Bewegen Sie den Cursor in eine der Ecken der Box (mithilfe der Pfeiltasten) und drücken ZOOM BOXZ. Dann gehen Sie mit den Pfeiltasten in die gegenüberliegende Ecke der gewünschten Zoombox. Der Cursor umfährt die Zoombox in der Anzeige. Wenn die Zoombox wie gewünscht ausgewählt ist, drücken Sie ZOOM. Der

Taschenrechner vergrößert den Inhalt der Zoombox, sodass dieser den gesamten Bildschirm ausfüllt.

Wenn Sie jetzt ZOUT drücken, dann fährt der Taschenrechner aus der aktuellen Zoombox unter Anwendung der H- und V-Faktoren heraus, sodass es sein kann, dass die Grafik nicht ganz so wiederhergestellt wird, wie sie vor dem Zoomen war.

ZDFLT, ZAUTO

Durch Drücken von ZDFLT wird die aktuelle Darstellung neu gezeichnet, und zwar mithilfe der voreingestellten Standardwerte der x- und y-Bereiche, d. h. –6.5 bis 6.5 für x und –3.1 bis 3.1 für y. Der Befehl ZAUTO wiederum erzeugt ein Zoomfenster unter Anwendung des Bereichs der aktuellen unabhängigen Variable (x), wobei jedoch der Bereich der abhängigen Variable (y) an die Kurve angepasst wird (so wie bei Verwendung der Funktion AUTO in der Eingabemaske im Fenster PLOT WINDOW (gleichzeitig ← WIN drücken wenn im RPN-Modus).

HZIN, HZOUT, VZIN and VZOUT

Mit diesen Funktionen können Sie in das Grafikfenster hinein- und herauszoomen, in horizontaler und vertikaler Richtung entsprechend den aktuellen H- und V-Faktoren.

CNTR

Hiermit wird das Bild vergrößert, wobei das Zentrum des Zoomfensters die aktuelle Cursorposition ist. Die verwendeten Zoomfaktoren sind die aktuellen H- und V-Faktoren.

ZDECI

Vergrößert die Grafik so, dass die Begrenzung des x-Intervalls auf einen Dezimalwert abgerundet wird.

ZINTG

Vergrößert die Grafik so, dass die Pixeleinheiten benutzerdefinierte Einheiten werden. Der Mindestwert des PICT-Fensters ist beispielsweise 131 Pixel.

Wenn Sie nun ZINTG verwenden und der Cursor auf dem Bildschirmmittelpunkt steht, dann wird das Fenster vergrößert, sodass die x-Achse von –64,5 bis 65,5 reicht.

ZSQR

Vergrößert die Grafik so, dass der Darstellungsmaßstab bei 1:1 bleibt, wobei die x-Skala angepasst wird und die y-Skala bestehen bleibt, wenn das Fenster breiter als höher ist. Dies erfordert proportionales Zoomen.

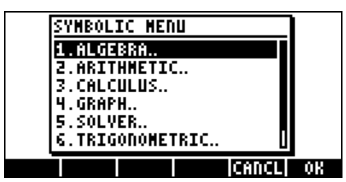
ZTRIG

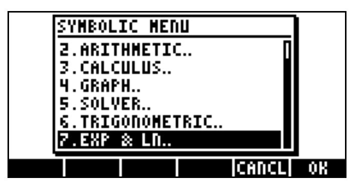
Vergrößert die Grafik so, dass die x-Skala einen Bereich von etwa -3π to $+3\pi$ mit einbezieht, dem bevorzugten Bereich bei trigonometrischen Funktionen.

Hinweis: Keine dieser Funktionen kann programmiert werden. Sie sind nur im interaktiven Gebrauch nützlich. Hier gilt es, den Befehl ZFACT im ZOOM-Menü nicht mit der Funktion ZFACTOR zu verwechseln, die in Gasdynamik- und chemischen Anwendungen verwendet wird (siehe Kapitel 3).

Symbolisches Menü und Grafiken

Das SYMBOLIC MENU (symbolisches Menü) wird durch Drücken der Taste SYMB (Tastenfeld: vierte Taste von links in der vierten Reihe von oben) aktiviert. Dieses Menü enthält eine Liste von Menüs im Zusammenhang mit dem Computer Algebraic System (Algebraisches System des Computers), kurz CAS genannt, und diese sind:



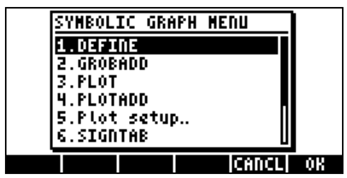


Alle außer einem dieser Menüs sind direkt von der Tastatur erreichbar, durch Drücken der entsprechenden folgenden Tastenkombinationen. Aufgeführt ist auch das Kapitel des Benutzerhandbuchs, in dem das entsprechende Menü beschrieben ist:

ALGEBRA..	↱ ALG (Taste 4)	Kap. 5
ARITHMETIC..	↰ ARITH (Taste 1)	Kap. 5
CALCULUS..	↰ CALC (Taste 4)	Kap. 13
SOLVER..	↰ S.SLV (Taste 7)	Kap. 6
TRIGONOMETRIC..	↱ TRIG (Taste 8)	Kap. 5
EXP&LN..	↰ EXP&LN (Taste 8)	Kap. 5

Das Menü SYMB/GRAPH

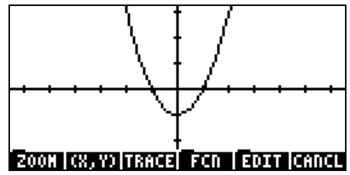
Das Untermenü GRAPH im Menü SYMB enthält folgende Funktionen:



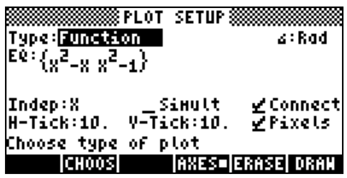
DEFINE: gleicht der Tastenfolge ↰ DEF (Taste 2).
GROBADD: fügt zwei GROBs ein, den ersten über den zweiten (siehe Kapitel 22).
PLOT(function): Zeichnet eine Funktion, ähnlich wie ↰ 2D/3D
PLOTADD(function): fügt diese Funktion der Liste der darzustellenden Funktionen hinzu, ähnlich wie ↰ 2D/3D
Plot setup..: wie ↰ 2D/3D
SIGNTAB(function): Zeichentabelle einer bestimmten Funktion, in der Intervalle positiver und negativer Variationen, Nullpunkte und unendlicher Asymptoten verzeichnet sind.
TABVAL: Wertetabelle für eine Funktion.
TABVAR: Variationstabelle einer Funktion.

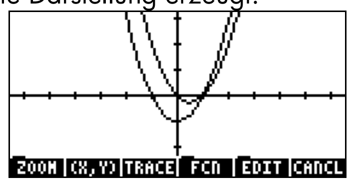
Es folgen einige Beispiele für diese Funktionen.

PLOT(X^2-1) ist ähnlich wie ↰ 2D/3D bei EQ: X^2 -1. Mithilfe von ERASE DRAW wird die Darstellung erzeugt:

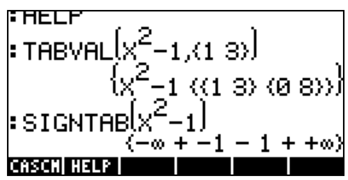


PLOTADD(X^2-X) ist ähnlich wie ⬅ 2D/3D , aber fügt diese Funktion EQ hinzu: X^2 -1. Mithilfe von ERASE DRAW wird die Darstellung erzeugt:

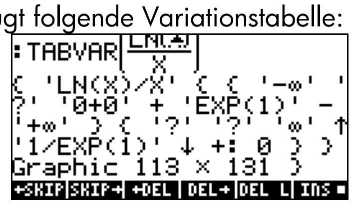




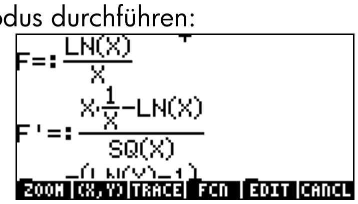
TABVAL(X^2-1,{1, 3}) erzeugt eine Liste von {min max} –Werten der Funktion im Intervall {1,3}, während SIGNTAB(X^2-1) das Vorzeichen der Funktion im Intervall zeigt (-∞,+), wobei f(x) > 0 bei (-∞,-1), f(x) <0, bei (-1,1), und f(x) > 0 bei (1,+ ∞) ist.

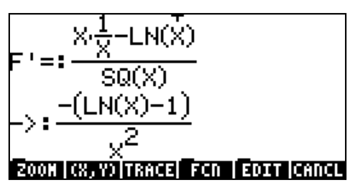


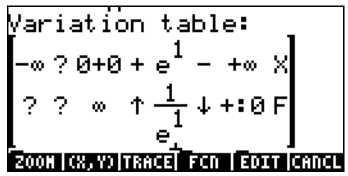
TABVAR(LN(X)/X) erzeugt folgende Variationstabelle:



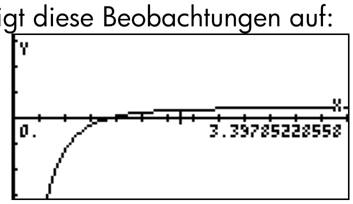
Eine detaillierte Interpretation der Variationstabelle lässt sich einfacher im RPN-Modus durchführen:





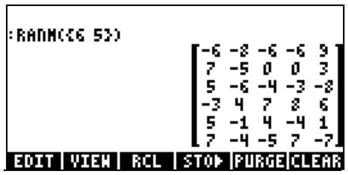


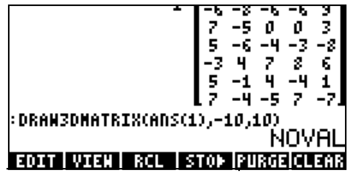
Die Ausgabe erfolgt in einem grafischen Format, wobei die ursprüngliche Funktion F(X) gezeigt wird, die Ableitung F'(X) gleich nach Ableitung und Bereinigung, und schließlich eine Variationstabelle. Die Tabelle besteht aus zwei Reihen, die auf der rechten Seite bezeichnet sind. Die obere Reihe zeigt nun also die Werte für X und die zweite Reihe die Werte für F. Die Fragezeichen kennzeichnen Unsicherheiten bzw. fehlende Definitionen. Beispielsweise ist für X<0 LN(X) nicht definiert, deshalb zeigen die X-Reihen ein Fragezeichen in diesem Intervall. Genau bei Null (0+0) ist F unendlich, da X = e, F = 1/e. F steigt vor Erreichen dieses Wertes, was mit dem Pfeil nach oben angezeigt wird, und sinkt nach diesem Wert (X=e), und wird dann etwas größer als Null (+:0), da X unendlich wird. Die untenstehende Abbildung der Kurvezeigt diese Beobachtungen auf:



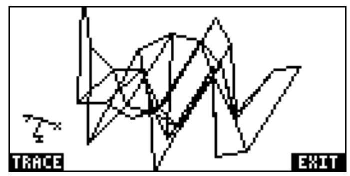
Funktion DRAW3DMATRIX

Diese Funktion nimmt eine n×m Matrix als Argument, **Z**, = [z_{ij}], und einen Minimal- und Maximalwert für die Darstellung. Am besten wählen Sie die Werte v_{min} und v_{max} , sodass diese die Werte aus **Z** enthalten. Der allgemeine Aufruf der Funktion ist deshalb DRAW3DMATRIX(**Z**,v_{min},v_{max}). Um die Verwendung dieser Funktion zu zeigen, erzeugen wir zunächst eine 6×5 Matrix unter Verwendung von RANM({6,5}), und dann rufen wir die Funktion DRAW3DMATRIX auf, gemäß Darstellung unten:





Die Darstellung ähnelt der Fast-3D-Darstellung. Unten sind verschiedene Ansichten der Darstellung abgebildet:



Kapitel 13
Anwendungen der Infinitesimalrechnung

In diesem Kapitel werden Anwendungen der Taschenrechnerfunktionen auf Operationen der Infinitesimalrechnung erläutert, z. B. Grenzwerte, Ableitungen, Integrale, Potenzreihen usw.

Das Menü CALC (Infinitesimalrechnung)

Zahlreiche der in diesem Kapitel dargestellten Funktionen befinden sich im Menü CALC des Taschenrechners, das über die Tastenkombination ← CALC (der Taste 4 zugeordnet) aufgerufen wird. Das Menü CALC enthält die folgenden Einträge:

Bei den ersten vier Optionen dieses Menüs handelt es sich eigentlich um Untermenüs für (1) Ableitungen und Integrale, (2) Grenzwerte und Potenzreihen, (3) Differenzialgleichungen und (4) Grafiken. In diesem Kapitel werden die Funktionen (1) und (2) dargestellt. Differenzialgleichungen, der Inhalt von Menüpunkt (3), werden in Kapitel 16 dargestellt. Grafikfunktionen, der Inhalt von Menüpunkt (4), wurden am Ende von Kapitel 12 dargestellt. Schließlich handelt es sich bei den Menüpunkten 5. DERVX und 6. INTVX um die Funktionen, mit denen Sie eine Ableitung und ein unbestimmtes Integral für eine Funktion der CAS-Standardvariablen (normalerweise 'X') erhalten. Die Funktionen DERVX und INTVX werden später ausführlicher behandelt.

Grenzwerte und Ableitungen

Bei der Differenzialrechnung werden Ableitungen (oder Änderungsraten) von Funktionen und ihren Anwendungen in der mathematischen Analyse behandelt. Die Ableitung einer Funktion ist als der Grenzwert der Differenz einer Funktion definiert, wenn das Inkrement der unabhängigen Variablen gegen Null geht. Grenzwerte werden außerdem verwendet, um die Stetigkeit von Funktionen zu überprüfen.

Funktion lim

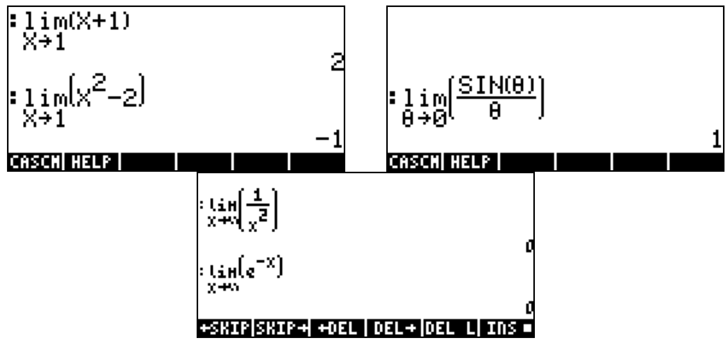
Der Taschenrechner enthält die Funktion *lim* zum Berechnen der Grenzwerte von Funktionen. Bei dieser Funktion wird ein Ausdruck als Eingangswert verwendet, der eine Funktion und ihren Wert darstellt, wobei der Grenzwert zu berechnen ist. Die Funktion *lim* kann über den Befehlskatalog (→ CAT ALPHA ← L) oder über die Option 2. LIMITS & SERIES des Menüs CALC (siehe oben) aufgerufen werden.

Hinweis: Die Funktionen im Menü LIMITS & SERIES sind unten abgebildet:

Mithilfe der Funktion DIVPC werden zwei Polynome dividiert, sodass sich eine Reihenentwicklung ergibt. Die Funktionen DIVPC, SERIES, TAYLOR0 und TAYLOR werden in Reihenentwicklungen von Funktionen verwendet und in diesem Kapitel ausführlicher erläutert.

Die Funktion *lim* wird im ALG-Modus als `lim(f(x),x=a)` eingegeben, um den Grenzwert $\lim_{x \to a} f(x)$ zu berechnen. Geben Sie im RPN-Modus zunächst die Funktion, dann den Ausdruck 'x=a', und schließlich die Funktion *lim* ein. Im Folgenden werden Beispiele für den ALG-Modus sowie einige Grenzwerte gegen Unendlich dargestellt. Die Tastenkombinationen für das erste Beispiel lauten wie folgt (wenn im algebraischen Modus Systemflag 117 auf CHOOSE boxes gesetzt ist):

← CALC 2 OK 2 OK X + 1 → , X → = 1 ENTER



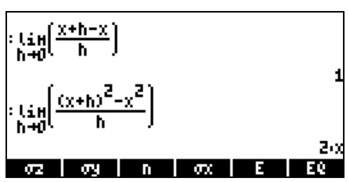
Das Unendlichkeitssymbol ist der Taste 0 zugeordnet. d. h. ., ← ∞ .

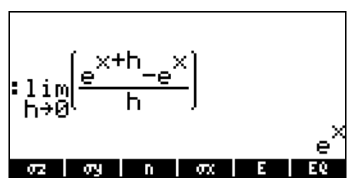
Ableitungen

Die Ableitung einer Funktion f(x) mit x = a ist definiert als der Grenzwert

$$\frac{df}{dx} = f'(x) = \lim_{h->0} \frac{f(x+h) - f(x)}{h}$$

In den folgenden Bildschirmabbildungen werden einige Beispiele für Ableitungen mit diesem Grenzwert dargestellt:

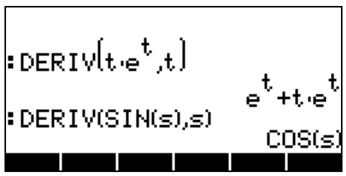


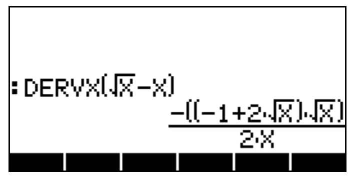


Funktionen DERIV und DERVX

Die Funktion DERIV nimmt als Eingangswert Ableitungen von beliebigen unabhängigen Variablen an, während die Funktion DERVX Ableitungen in Bezug auf die CAS-Standardvariable VX (normalerweise 'X') als Eingangswert annimmt. Während die Funktion DERVX direkt im Menü CALC verfügbar ist, können beide Funktionen über das Menü CALCL (← CALC) im Untermenü DERIV.&INTEG aufgerufen werden.

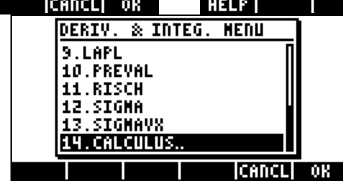
Die Funktion DERIV erfordert eine Funktion, beispielsweise f(t), und eine unabhängige Variable, z. B. t, während für die Funktion DERVX nur eine Funktion von VX erforderlich ist. Im Folgenden werden Beispiele im ALG-Modus dargestellt. Beachten Sie, dass im RPN-Modus die Argumente vor dem Anwenden der Funktion eingegeben werden müssen.





Das Menü DERIV&INTEG

Die in diesem Untermenü verfügbaren Funktionen sind unten aufgelistet:

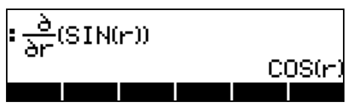


Von diesen Funktionen werden DERIV und DERVX für Ableitungen verwendet. Zu den anderen Funktionen zählen Funktionen für Stammfunktionen und Integrale (IBP, INTVX, PREVAL, RISCH, SIGMA und SIGMAVX), Fourier-Reihen (FOURIER) und Vektorrechnung (CURL, DIV, HESS, LAPL). Zunächst werden die Funktionen DERIV und DERVX erläutert. Die übrigen Funktionen werden entweder weiter unten in diesem Kapitel oder in den folgenden Kapiteln dargestellt.

Berechnen von Ableitungen mit ∂

Das Symbol ist als [→] ∂ (die [COS]-Taste) verfügbar. Dieses Symbol kann zum Eingeben einer Ableitung in den Stack oder in EquationWriter verwendet werden (siehe Kapitel 2). Wenn Sie mithilfe des Symbols eine Ableitung in den Stack eingeben, geben Sie direkt danach die unabhängige Variable und dann zwei Klammern um die abzuleitende Funktion ein. Verwenden Sie daher zum Berechnen der Ableitung d(sin(r),r) im ALG-Modus die Zeichen:
[→] ∂ [ALPHA][←][R][←][()] [SIN][ALPHA][←][R][ENTER]

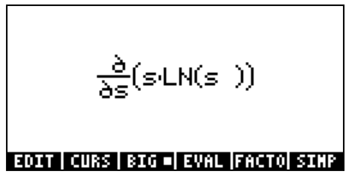
Im RPN-Modus muss dieser Ausdruck in Anführungszeichen eingeschlossen werden, bevor er in den Stack eingegeben wird. Im ALG-Modus lautet das Ergebnis:



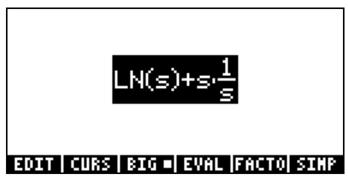
In EquationWriter gibt der Taschenrechner folgenden Ausdruck aus, wenn Sie [→] ∂, drücken:

Der Einfügecursor (◆) befindet sich rechts vom Nenner, bis Sie eine unabhängige Variable eingeben, z.B. s: [ALPHA][←][S]. Drücken Sie anschließend die Nach-Rechts-Taste (▶), um den Cursor an den Platzhalter zwischen den Klammern zu verschieben.

Geben Sie anschließend die abzuleitende Funktion ein, z. B. s*ln(s):



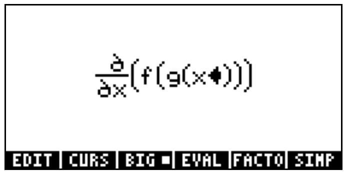
Zum Berechnen der Ableitung in EquationWriter drücken Sie viermal die Nach-Oben-Taste ▲, um den gesamten Ausdruck auszuwählen, und drücken Sie anschließend EVAL. Die Ableitung wird in EquationWriter wie folgt berechnet:

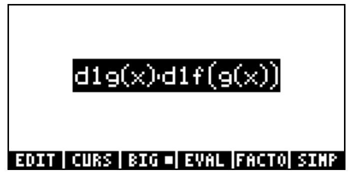


Hinweis: Das Symbol ∂ wird in der Mathematik verwendet, um eine partielle Ableitung anzugeben, d. h. die Ableitung einer Funktion mit mehreren Variablen. Jedoch werden normale und partielle Ableitungen vom Taschenrechner nicht unterschieden und für beide dasselbe Symbol verwendet. Beachten Sie diesen Unterschied, wenn Sie Ergebnisse des Taschenrechners zu Papier bringen.

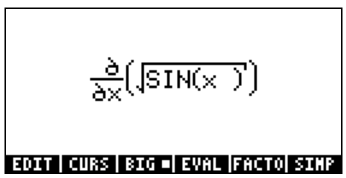
Die Kettenregel

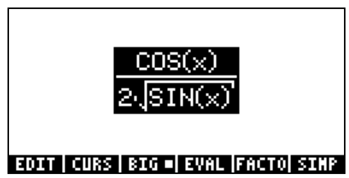
Die Kettenregel für Ableitungen wird auf Ableitungen zusammengesetzter Funktionen angewendet. Ein allgemeiner Ausdruck für die Kettenregel lautet d{f[g(x)]}/dx = (df/dg)·(dg/dx). Bei Verwendung des Taschenrechners lautet diese Formel wie folgt:





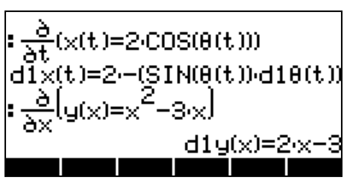
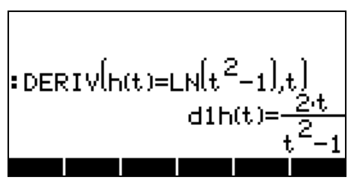
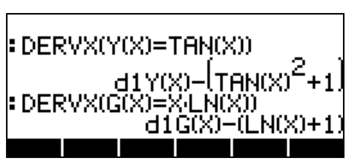
Bei dem Ausdruck d1 vor g(x) und f(g(x)) in der Formel oben handelt es sich um eine Abkürzung, die vom Taschenrechner verwendet wird, um eine erste Ableitung anzugeben, wenn die unabhängige Variable, in diesem Fall x, eindeutig definiert ist. Somit wird das Ergebnis wie in der oben dargestellten Formel für die Kettenregel interpretiert. Es folgt ein weiteres Beispiel für eine Anwendung der Kettenregel:





Ableitungen von Gleichungen

Mit dem Taschenrechner können Sie Ableitungen von Gleichungen berechnen, d. h. Ausdrücke, die auf beiden Seiten des Gleichheitszeichens Ableitungen enthalten. Unten werden einige Beispiele dargestellt:

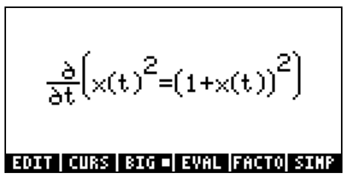




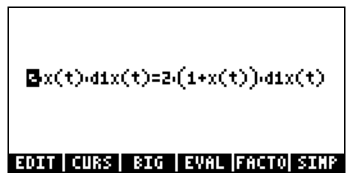
Beachten Sie, dass in den Ausdrücken, bei denen das Ableitungszeichen (∂) oder die Funktion DERIV verwendet wurde, das Gleichheitszeichen in der

Gleichung beibehalten wird, doch nicht in den Fällen, in denen die Funktion DERVX verwendet wurde. In diesen Fällen wurde die Gleichung neu geschrieben, und alle zugehörigen Ausdrücke wurden auf die linke Seite des Gleichheitszeichens verschoben. Außerdem wurde das Gleichheitszeichen entfernt, doch der Ergebnisausdruck ist selbstverständlich gleich Null.

Implizite Ableitungen

Implizite Ableitungen können z. B. in folgenden Ausdrücken verwendet werden:





Anwendung von Ableitungen

Mit Ableitungen können Graphen von Funktionen berechnet und Funktionen einer Variablen optimiert werden (z. B. Suchen der Maxima und Minima). Im Folgenden werden einige Anwendungen von Ableitungen dargestellt.

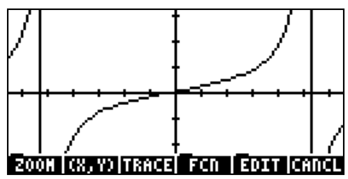
Berechnen der Graphen von Funktionen

In Kapitel 11 wurden einige Funktionen vorgestellt, die auf dem Grafikbildschirm zum Berechnen der Graphen von Funktionen der Form y = f(x) zur Verfügung stehen. Zu diesen Funktionen zählen (X,Y) und TRACE zum Bestimmen von Punkten des Graphen sowie Funktionen in den Menüs ZOOM und FCN. Mithilfe der Funktionen im Menü ZOOM können Sie die Darstellung eines Graphen vergrößern, um ihn detaillierter berechnen zu können. Diese Funktionen werden in Kapitel 12 ausführlich beschrieben. Unter den Funktionen des Menüs FCN können SLOPE, EXTR, F' und TANL zum Bestimmen der Steigung einer Tangente des Graphen, zum Bestimmen der Extremwerte (Minima und Maxima) der Funktion, zum Zeichnen der Ableitung und zum Bestimmen der Gleichung für die Tangente verwendet werden.

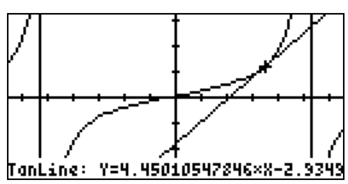
Probieren Sie folgendes Beispiel für die Funktion y = tan(x) aus.

- Drücken Sie im RPN-Modus gleichzeitig ← 2D/3D ,, um das Fenster PLOT SETUP zu öffnen.
- Ändern Sie ggf. mithilfe von [CHOOS] TYPE in FUNCTION.
- Drücken Sie ▼, und geben Sie die Gleichung 'TAN(X) ' ein.
- Stellen Sie sicher, dass die unabhängige Variable auf 'X' gesetzt ist.
- Drücken Sie NXT OK, um zum normalen Display des Taschenrechners zurückzukehren.
- Drücken Sie gleichzeitig ← WIN , um das PLOT-Fenster aufzurufen.
- Ändern Sie den H-VIEW-Bereich auf -2 bis 2 und den V-VIEW-Bereich auf -5 bis 5.
- Drücken Sie ERASE DRAW, um die Funktion in Polarkoordinaten zu zeichnen.

Die resultierende Zeichnung wird wie folgt dargestellt:



- Beachten Sie die vertikalen Linien, die die Asymptoten darstellen. Diese sind nicht Bestandteil des Graphen, sondern zeigen die Punkte an, an denen TAN(X) bei bestimmten Werten von X gegen ± ∞ geht.
- Drücken Sie TRACE (X,Y) und bewegen Sie den Cursor an Punkt X: 1,08E0, Y: 1,86E0. Drücken Sie anschließend NXT FCN SLOPE. Das Ergebnis ist die Steigung: 4,45010547846.
- Drücken Sie NXT NXT TANL. Hierdurch wird die Gleichung der Tangente erstellt und der zugehörige Graph in dieselbe grafische Darstellung gezeichnet. Das Ergebnis wird in der Abbildung unten dargestellt:



- Drücken Sie NXT PICT CANCL ON, um zum normalen Display des Taschenrechners zurückzukehren. Beachten Sie, dass die gewünschte Steigung und Tangente im Stack aufgelistet sind.

Funktion DOMAIN

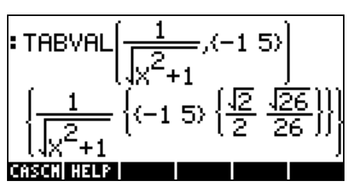
Mit der über den Befehlskatalog (→ _CAT_) verfügbaren Funktion DOMAIN erhalten Sie den Definitionsbereich einer Funktion als eine Liste von Zahlen und Beschreibungen. Beispielsweise gibt

an, dass die Funktion LN(X) zwischen –∞ und 0 nicht definiert ist (?), dass sie jedoch zwischen 0 und +∞ definiert ist (+). Andererseits gibt

an, dass die Funktion weder zwischen –∞ und -1 noch zwischen 1 und +∞ definiert ist. Der Definitionsbereich dieser Funktion lautet daher -1<X<1.

Funktion TABVAL

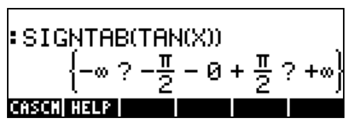
Diese Funktion wird über den Befehlskatalog oder im Menü CALC über das Untermenü GRAPH aufgerufen. Als Argument der Funktion TABVAL ist die Funktion der CAS-Variablen f(X) und eine Liste zweier Zahlen erforderlich, die den relevanten Bereich für die Funktion f(X) darstellen. Die Funktion TABVAL gibt die Eingabewerte plus den Bereich der Funktion zurück, der dem für die Eingabe verwendeten Bereich entspricht. Beispiel:



Durch dieses Ergebnis wird angegeben, dass der Bereich der Funktion $f(X)=\frac{1}{\sqrt{X^2+1}}$, der dem Definitionsbereich D = { -1,5 } entspricht, R = $\left\{\frac{\sqrt{2}}{2},\frac{\sqrt{26}}{26}\right\}$ ist.

Funktion SIGNTAB

Mit der über den Befehlskatalog (→ CAT) aufrufbaren Funktion SIGNTAB erhalten Sie Informationen über das Vorzeichen einer Funktion in ihrem Definitionsbereich. Beispielsweise gibt SIGNTAB für die Funktion TAN(X)



an, dass TAN(X) zwischen –π/2 und 0 negativ und zwischen 0 und π/2 positiv ist. In diesem Fall liefert SIGNTAB in den Intervallen zwischen –∞ und -π/2 und zwischen +π/2 und ∞ keine Informationen (?). Somit liefert SIGNTAB in diesem Fall ausschließlich Informationen über den Hauptdefinitionsbereich von TAN(X), also -π/2 < X < +π/2.

Unten ist ein zweites Beispiel für die Funktion SIGNTAB abgebildet:

In diesem Fall ist die Funktion für X<-1 negativ und für X>-1 positiv.

Funktion TABVAR

Diese Funktion wird über den Befehlskatalog oder im Menü CALC über das Untermenü GRAPH aufgerufen. Als Eingangswert wird die Funktion f(VX) verwendet, wobei VX die CAS-Standardvariable ist. Die Funktion gibt im RPN-Modus Folgendes zurück:

- Ebene 3: die Funktion f(VX).

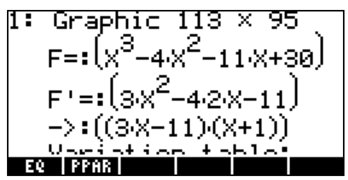
- Zwei Listen. Die erste Liste gibt die Abweichung der Funktion (d. h. die Punkte, an denen die Werte ab- oder zunehmen) in Bezug auf die unabhängige Variable VX an, die zweite Liste die Abweichung der Funktion in Bezug auf die abhängige Variable.

- Ein Grafikobjekt, das anzeigt, wie die Abweichungstabelle berechnet wurde.

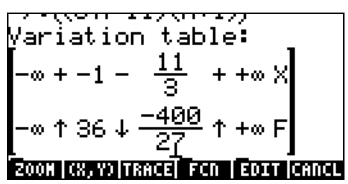
Beispiel: Berechnen Sie mithilfe der Funktion TABVAR die Funktion Y = X^3-4X^2-11X+30. Verwenden Sie im RPN-Modus die folgenden Tastenkombinationen:

`'X^3-4*X^2-11*X+30'` ENTER ↱ CAT ALPHA T (Auswahl von TABVAR) OK

Der Taschenrechner zeigt auf Ebene 1 des Stacks Folgendes an:



Dies ist ein Grafikobjekt. Um das gesamte Ergebnis anzuzeigen, drücken Sie ▼. Die Abweichungstabelle der Funktion wird wie folgt dargestellt:



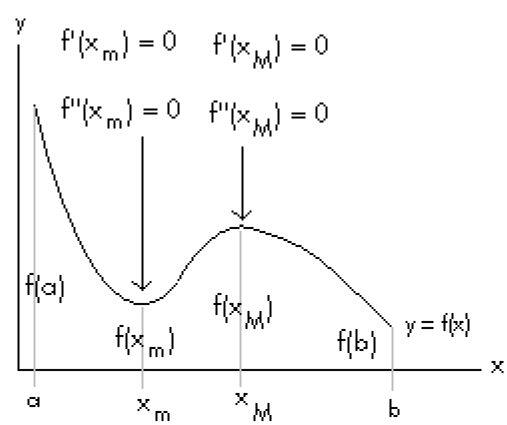
Drücken Sie ON, um zum normalen Display des Taschenrechners zurückzukehren. Drücken Sie ◄, um dieses letzte Ergebnis aus dem Stack zu entfernen.

Ebene 1 enthält nun zwei Listen, die der obersten und untersten Zeile der zuvor dargestellten Grafikmatrix entsprechen. Diese Listen können zum Programmieren verwendet werden. Drücken Sie ◄, um dieses letzte Ergebnis aus dem Stack zu entfernen.

Im Folgenden wird die oben dargestellte Abweichungstabelle erläutert: Die Funktion F(X) nimmt für X im Intervall (-∞, -1) zu und erreicht das Maximum 36 bei X = -1. Dann nimmt F(X) bis X = 11/3 ab und erreicht das Minimum -400/27. Anschließend nimmt F(X) bis +∞ zu. Außerdem ist bei X = ±∞ auch F(X) = ±∞.

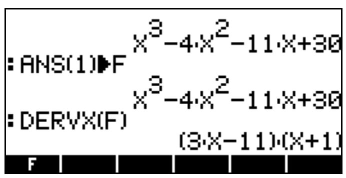
Verwenden von Ableitungen zum Berechnen von Extrempunkten

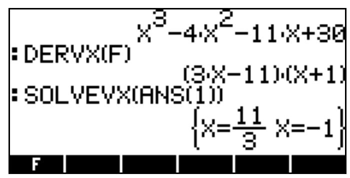
„Extrempunkte“ bzw. Extremwerte sind die allgemeine Bezeichnung für die Maximal- und Minimalwerte einer Funktion in einem bestimmten Intervall. Da die Ableitung einer Funktion an einem bestimmten Punkt die Steigung der Tangente zur Kurve an diesem Punkt darstellt, stellen die Werte von x für f'(x) = 0 die Punkte dar, an denen der Graph der Funktion das Maximum oder Minimum erreicht. Darüber hinaus gibt der Wert der zweiten Ableitung der Funktion f“(x) an diesen Punkten an, ob der Punkt ein *relatives oder lokales Maximum* [f“x)<0] bzw. *Minimum* [f“(x)>0] ist. Dies wird in der folgenden Abbildung veranschaulicht.



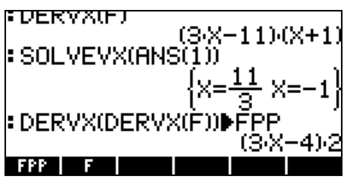
In dieser Abbildung beschränken wir uns darauf, die Extrempunkte der Funktion y = f(x) im x-Intervall [a,b] zu bestimmen. In diesem Intervall befinden sich zwei Punkte, $x = x_m$ und $x = x_M$, wobei f'(x)=0 ist. Der Punkt $x = x_m$ stellt ein lokales Minimum dar, wobei f"(x)>0 ist, während der Punkt $x = x_M$ ein lokales Maximum darstellt, wobei f"(x)<0 ist. Aus dem Graphen von y = f(x) folgt, dass sich das absolute Maximum im Intervall [a,b] bei x = a und das absolute Minimum bei x = b befindet.

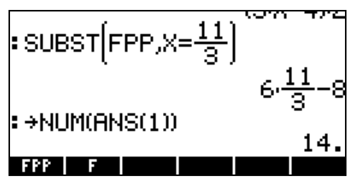
Um beispielsweise zu bestimmen, wo sich die kritischen Punkte der Funktion `X^3-4*X^2-11*X+30 befinden, können wir im ALG-Modus die folgenden Eingaben verwenden:



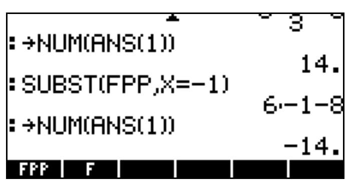


Wir ermitteln zwei kritische Punkte, bei x = 11/3 und bei x = -1. Geben Sie zum Berechnen der zweiten Ableitung an jedem Punkt Folgendes ein:

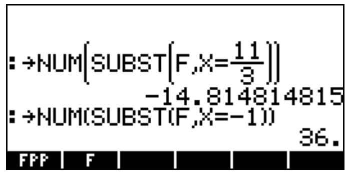




Auf dem letzten Bildschirm wird angezeigt, dass f"(11/3) = 14, sodass x = 11/3 ein relatives Minimum ist. Für x = -1 gilt Folgendes:



Dieses Ergebnis bedeutet, dass f“(-1) = -14, sodass x = -1 ein relatives Maximum ist. Berechnen Sie die Funktion an diesen Punkten, um zu überprüfen, ob tatsächlich gilt: (-1) > f(11/3).



Ableitungen höherer Ordnung

Ableitungen höherer Ordnung können durch mehrfaches Anwenden einer Ableitungsfunktion berechnet werden. Beispiel:

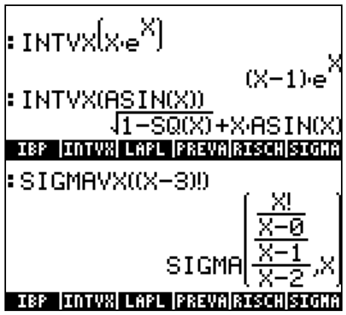
Stammfunktionen und Integrale

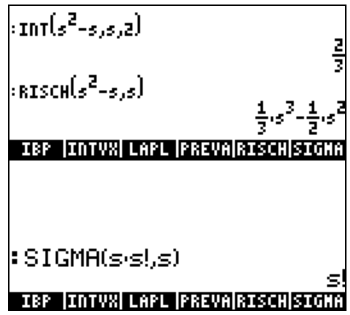
Die Stammfunktion einer Funktion f(x) ist die Funktion F(x), wobei f(x) = dF/dx. Da z. B. d(x^3) /dx = $3x^2$, lautet für f(x) = $3x^2$ die Stammfunktion F(x) = x^3 + C, wobei C eine Konstante ist. Eine Stammfunktion kann als unbestimmtes Integral dargestellt werden, d. h. $\int f(x)dx = F(x) + C$, wenn und nur wenn f(x) = dF/dx und C eine Konstante ist.

Funktionen INT, INTVX, RISCH, SIGMA und SIGMAVX

Der Taschenrechner enthält zum Berechnen der Stammfunktionen von Funktionen die Funktionen INT, INTVX, RISCH, SIGMA und SIGMAVX. Die Funktionen INT, RISCH und SIGMA können mit Funktionen beliebiger Variablen verwendet werden, während die Funktionen INTVX und SIGMAVX die Funktionen der CAS-Variablen VX (normalerweise ‘x’) verwenden. Die Funktionen INT und RISCH erfordern daher nicht nur den Ausdruck für die zu integrierende Funktion, sondern auch den Namen der unabhängigen Variablen. Die Funktion INT erfordert außerdem einen Wert von x, wenn die

Stammfunktion berechnet wird. Die Funktionen INTVX und SIGMAVX benötigen nur den Ausdruck der in Bezug auf VX zu integrierenden Funktion. Im Folgenden werden einige Beispiele im ALG-Modus dargestellt.



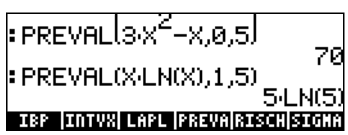


Beachten Sie, dass die Funktionen SIGMAVX und SIGMA für Integranden konzipiert sind, die eine ganzzahlige Funktion, z. B. die oben dargestellte Fakultätsfunktion (!), umfassen. Sie erzeugen eine so genannte diskrete Ableitung, d. h. eine ausschließlich für ganze Zahlen definierte Ableitung.

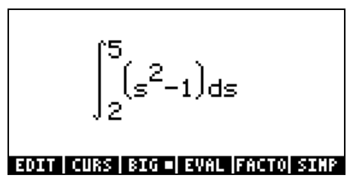
Bestimmte Integrale

Bei einem bestimmten Integral einer Funktion wird die resultierende Stammfunktion am oberen und unteren Grenzwert eines Intervalls (a,b) ausgewertet, und die berechneten Werte werden subtrahiert. Die Formel lautet $\int_a^b f(x)dx = F(b) - F(a),$ wobei f(x) = dF/dx.

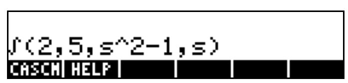
Die PREVAL(f(x),a,b)-Funktion des CAS kann solche Berechnungen vereinfachen. Sie gibt f(b)-f(a) zurück, wobei x die CAS-Variable VX ist.



Der Taschenrechner verfügt zum Berechnen bestimmter Integrale auch über das Integralsymbol als Tastenkombination [→] ∫ (der [TAN]-Taste zugeordnet). Integrale können am einfachsten im EquationWriter erstellt werden (ein Beispiel hierfür finden Sie in Kapitel 2). In EquationWriter erhalten Sie mit dem Symbol ⇨ ∫ das Integralzeichen sowie Platzhalter für die Integrationsgrenzen (a,b), für die Funktion f(x) und für die Variable (x) der Integration. In den folgenden Bildschirmabbildungen wird das Erstellen eines bestimmten Integrals veranschaulicht. Der Einfügecursor befindet sich zunächst an der unteren Grenze der Integration. Geben Sie einen Wert ein, und drücken Sie die Nach-Oben-Taste (▶), um zur oberen Grenze der Integration zu wechseln. Geben Sie an dieser Position einen Wert ein, und drücken Sie erneut ▶, um zur Position des Integranden zu wechseln. Geben Sie den Ausdruck für den Integranden ein, und drücken Sie die Taste erneut, um zum Platzhalter des Differenzials zu wechseln. Geben Sie an dieser Position die Variable der Integration ein, und das Integral kann berechnet werden.



Nun können Sie ENTER drücken, um das Integral an den Stack zurückzugeben, wie in der folgenden Eingabe (ALG-Modus) dargestellt:



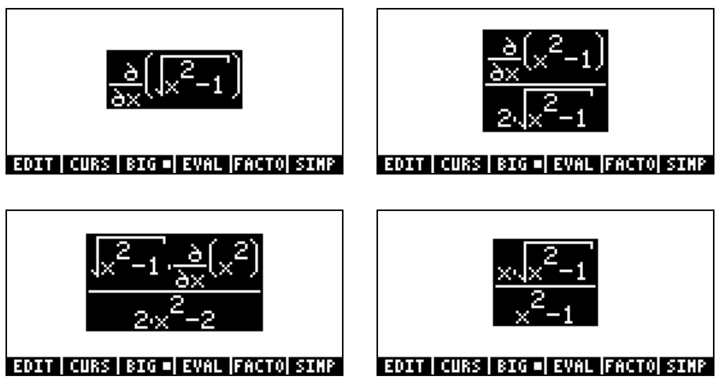
Dies ist das allgemeine Format für das bestimmte Integral, wenn es direkt in den Stack eingegeben wird, d. h. ∫(untere Grenze, obere Grenze, Integrand, Variable der Integration).

Wenn Sie an dieser Stelle ENTER drücken, wird das Integral im Stack berechnet:

Das Integral kann auch in EquationWriter berechnet werden, indem Sie den gesamten Ausdruck auswählen und die Menütaste EVAL verwenden.

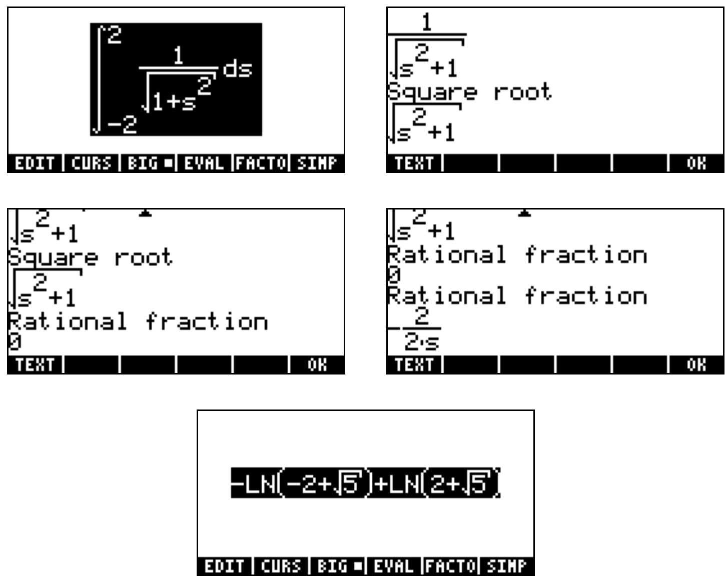
Schrittweise Berechnung von Ableitungen und Integralen

Wenn in den CAS MODES-Fenstern die Option Step/Step ausgewählt ist (siehe Kapitel 1), wird die Berechnung von Ableitungen und Integralen in einzelnen Schritten angezeigt. In der folgenden Abbildung wird z. B. die Berechnung einer Ableitung in EquationWriter dargestellt:



Beachten Sie die Anwendung der Kettenregel im ersten Schritt, wobei die Ableitung der Funktion unter dem Integral explizit im Zähler bleibt. Im zweiten Schritt wird der resultierende Bruch in eine rationale Zahl umgewandelt (die Quadratwurzel aus dem Nenner entfernt) und vereinfacht. Im dritten Schritt wird die endgültige Version angezeigt. Jeder Schritt wird durch Drücken der Menütaste EVAL angezeigt, bis der Punkt erreicht ist, an dem der Ausdruck durch die weitere Anwendung der Funktion EVAL nicht mehr geändert wird.

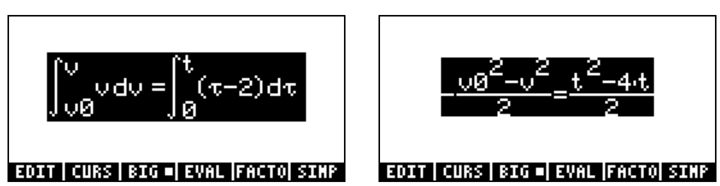
Im folgenden Beispiel wird die schrittweise Berechnung eines bestimmten Integrals in EquationWriter dargestellt:



Beachten Sie, dass durch den schrittweisen Vorgang Informationen über die von CAS zum Lösen dieses Integrals ausgeführten Zwischenschritte bereitgestellt werden. CAS bestimmt zunächst ein Quadratwurzelintegral, dann einen rationalen Bruch sowie einen zweiten rationalen Ausdruck und liefert dann das Endergebnis. Beachten Sie, dass diese Schritte für den Taschenrechner von großer Bedeutung sind, obwohl für den Benutzer nicht genügend Informationen über die einzelnen Schritte geboten werden.

Integrieren einer Gleichung

Das Integrieren eine Gleichung ist unkompliziert. Der Taschenrechner integriert einfach beide Seiten der Gleichung gleichzeitig, z. B.

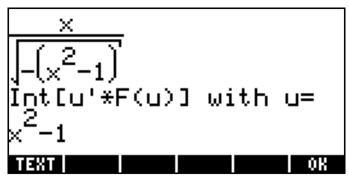


Methoden der Integration

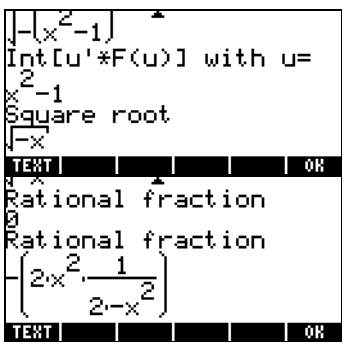
Wie in den folgenden Beispielen gezeigt, können mit dem Taschenrechner mehrere Integrationsmethoden angewendet werden.

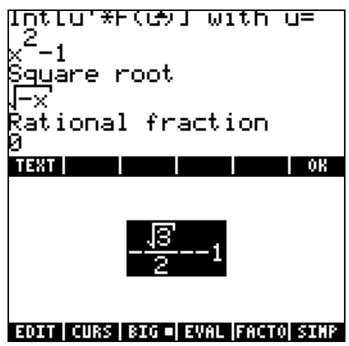
Substitution oder Ändern von Variablen

Angenommen, wir möchten das Integral $\int_0^2 \frac{x}{\sqrt{1-x^2}}dx$ berechnen. Bei einer schrittweisen Berechnung in EquationWriter lautet die Abfolge der Variablensubstitutionen wie folgt:



Im zweiten Schritt wird die zu verwendende ordnungsgemäße Substitution dargestellt, $u = x^2-1$.





In den letzten vier Schritten wird die Entwicklung der Lösung gezeigt: eine Quadratwurzel, dann ein Bruch, ein zweiter Bruch und das Endergebnis. Das Ergebnis kann unter Verwendung der Funktion SIMP vereinfacht werden und wird wie folgt angezeigt:

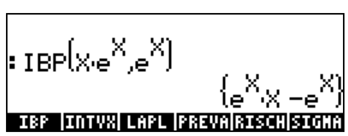
Partielle Integration und Differenziale

Ein Differenzial einer Funktion y = f(x) ist als dy = f'(x) dx definiert, wobei f'(x) die Ableitung von f(x) ist. Differenziale werden für die Darstellung kleiner Inkremente in den Variablen verwendet. Das Differenzial eines Produkts zweier Funktionen y = u(x)v(x) wird durch dy = u(x)dv(x) + du(x)v(x) oder einfach durch d(uv) = udv - vdu berechnet. Somit wird das Integral von udv = d(uv) - vdu als $\int udv = \int d(uv) - \int vdu$ geschrieben. Da gemäß der Definition eines Differenzials ∫dy = y ist, schreiben wir den vorherigen Ausdruck als

$$\int udv = uv - \int vdu\ .$$

Diese als partielle Integration bezeichnete Formel kann zum Bestimmen eines Integrals verwendet werden, wenn dv auf einfache Weise integriert werden kann. Beispielsweise kann das Integral $\int xe^x dx$ durch partielle Integration bestimmt werden, wenn wir u = x, dv = e^xdx verwenden, da v = e^x. Wenn du = dx ist, lautet das Integral $\int xe^x dx = \int udv = uv - \int vdu = xe^x - \int e^x dx = xe^x - e^x$.

Der Taschenrechner enthält im Menü CALC/DERIV&INTG die Funktion IBP, die als Argumente die ursprüngliche zu integrierende Funktion, also u(X)*v'(X), und die Funktion v(X) annimmt, und gibt u(X)*v(X) sowie -v(X)*u'(X) zurück. Mit anderen Worten, die Funktion IBP gibt die beiden Ausdrücke auf der rechten Seite der Gleichung der partiellen Integration zurück. Für das oben verwendete Beispiel können wir im ALG-Modus Folgendes schreiben:



Somit können wir die Funktion IBP verwenden, um die Komponenten einer partiellen Integration bereitzustellen. Der nächste Schritt muss separat ausgeführt werden.

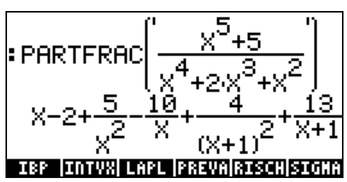
Es ist wichtig zu erwähnen, dass das Integral direkt berechnet werden kann, indem z. B. folgende Eingabe verwendet wird:

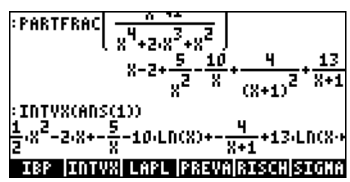
Integration durch Partialbruchzerlegung

Die in Kapitel 5 vorgestellte Funktion PARTFRAC ermöglicht die Zerlegung eines Bruches in Partialbrüche. Diese Methode empfiehlt sich, um einen komplizierten Bruch in eine Summe einfacher Brüche zu zerlegen, die dann nacheinander integriert werden können. Um beispielsweise

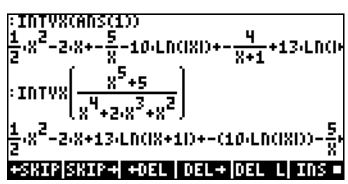
$$\int \frac{X^5+5}{X^4+2X^3+X}\,dX$$

zu integrieren, können wir den Bruch wie folgt in seine Partialbrüche zerlegen:





Die direkte Integration führt zum gleichen Ergebnis, wobei einige Ausdrücke umgestellt werden (in CAS festgelegter rigoroser Modus, siehe Kapitel 2):

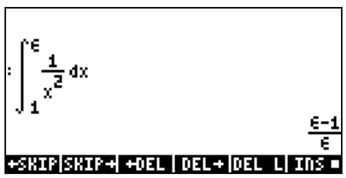


Unzulässige Integrale

Hierbei handelt es sich um Integrale mit Unendlich als Integrationsgrenze. Üblicherweise wird bei einem unzulässigen Integral zunächst das Integral als Grenzwert gegen Unendlich berechnet, z. B.

$$\int_1^{\infty} \frac{dx}{x^2} = \lim_{\varepsilon \to \infty} \int_1^{\varepsilon} \frac{dx}{x^2} .$$

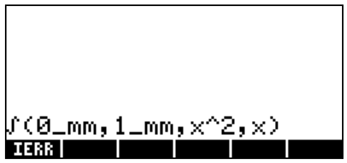
Mit dem Taschenrechner setzen wir die Berechnung wie folgt fort:

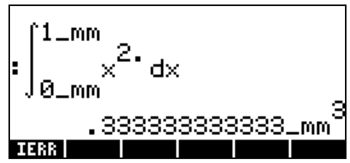


Stattdessen können Sie auch das Integral mit Unendlich als Grenzwert sofort berechnen, z. B.:

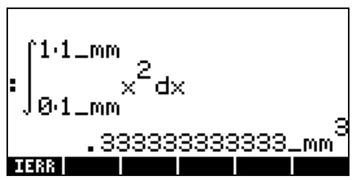
Integralrechnungen mit Einheiten

Eine Integrale kann auch mit, innerhalb der Grenzwerte der Integralen eingeschlossenen Einheiten, berechnet werden, wie man das im nachfolgenden Beispiel im ALG-Modus, mit CAS auf Approx-Modus eingestellt, sehen kann. Auf der linken Seite ist die im Editor eingetippte Integrale vor Drücken der Taste ENTER zu sehen. Auf der rechten Seite das Ergebnis, nach Drücken der Taste ENTER.



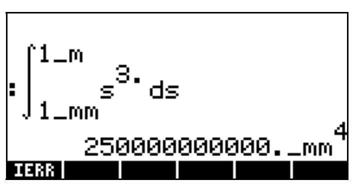


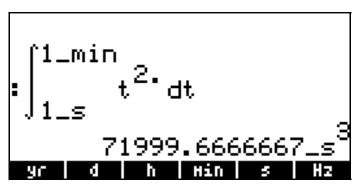
Geben Sie die Integrale mit dem CAS auf Exact-Modus eingestellt ein, erhalten Sie die Meldung in den Approx-Modus umzustellen, die Grenzen der Integrale aber, werden in einem anderen, als hier gezeigten, Format ausgegeben:



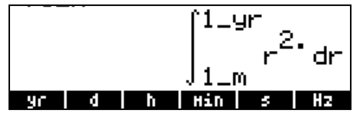
Die hier gezeigten Grenzen sind 1×1_mm und 0×1_mm, welches 1_mm und 0_mm, wie vorher gezeigt entspricht. Beachten Sie nur die unterschiedlichen Ausgabeformate.

Einige Notizen bei der Verwendung von Einheiten innerhalb der Grenzwerte der Integrale:
1 – Die Einheiten der unteren Grenze der Integrale werden die im Endresultat gezeigten sein, wie das im nachstehenden Beispiel veranschaulicht wird:

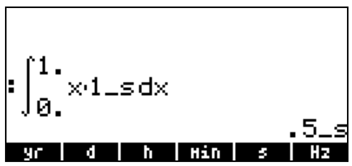


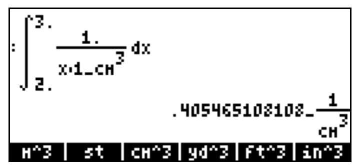


2 – Die oberen Grenzwerte müssen mit den unteren konsistent sein. Andernfalls ist das Ergebnis eine nicht ausgewertete Integrale. So zum Beispiel:

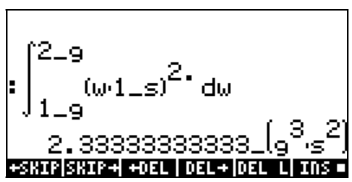


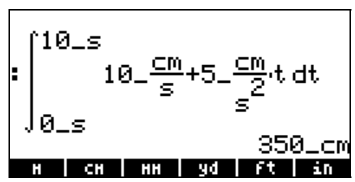
3 – Der Integrand könnte zwei Einheiten beinhalten. So zum Beispiel:





4 – Beinhalten beide, die Grenzwerte wie auch der Integrand, Einheiten, wird das Ergebnis eine Kombination, entsprechend den Regeln der Integrale sein, dieser Einheiten darstellen. So zum Beispiel:





Unendliche Reihen

Eine unendliche Reihe hat die Form $\sum_{n=0,1}^{\infty} h(n)(x-a)^n$. Unendliche Reihen beginnen normalerweise mit dem Index n = 0 oder n = 1. Jedes Glied der Reihe besitzt einen Koeffizienten h(n), der vom Index n abhängt.

Taylor- und MacLaurin-Reihen

Eine Funktion f(x) kann mit einer Taylor-Reihe zu einer unendlichen Reihe um einen Punkt $x=x_0$ entwickelt werden, also

$$f(x) = \sum_{n=0}^{\infty} \frac{f^{(n)}(x_o)}{n!} \cdot (x-x_o)^n \ ,$$

wobei $f^{(n)}(x)$ die n-te Ableitung von f(x) darstellt, mit x, $f^{(0)}(x) = f(x)$.

Wenn x_0 gleich Null ist, wird die Reihe als MacLaurin-Reihe bezeichnet, d. h.

$$f(x) = \sum_{n=0}^{\infty} \frac{f^{(n)}(0)}{n!} \cdot x^n$$

Taylor-Polynom und Rest

In der Realität können nicht alle Glieder einer unendlichen Reihe berechnet werden. Stattdessen berechnen wir mit einem Polynom der Ordnung k, $P_k(x)$ einen Näherungswert für die Reihe und schätzen die Ordnung eines Residuums $R_k(x)$, sodass

$$f(x) = \sum_{n=0}^{k} \frac{f^{(n)}(x_o)}{n!} \cdot (x - x_o)^n + \sum_{n=k+1}^{\infty} \frac{f^{(n)}(x_o)}{n!} \cdot (x - x_o)^n \ ,$$

d. h.

$$f(x) = P_k(x) + R_k(x).$$

Das Polynom $P_k(x)$ wird als Taylor-Polynom bezeichnet. Die Ordnung des Residuums wird als eine kleine Menge $h = x\text{-}x_0$ geschätzt, d. h., das Polynom wird bei einem Wert von x berechnet, der sehr nah an x_0 liegt. Das Residuum wird durch

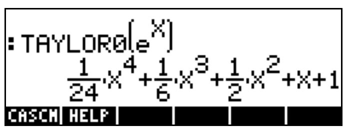
$$R_k(x) = \frac{f^{(k+1)}(\xi)}{k!} \cdot h^{k+1},$$

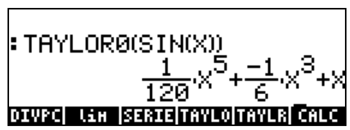
angegeben, wobei ξ eine Zahl nahe $x = x_0$ ist. Da ξ im Gegensatz zu einem Schätzwert für das Residuum normalerweise nicht bekannt ist, geben wir einen Schätzwert für die Ordnung des Residuums in Bezug auf h an, d. h., wir sagen, dass $R_k(x)$ einen Fehler der Ordnung h^{n+1} aufweist, oder $R \approx O(h^{k+1})$. Wenn h eine kleine Zahl ist, z. B. h<<1, ist h^{k+1} in der Regel sehr klein, d. h. $h^{k+1} << h^k << \ldots << h << 1$. Je größer daher für x nahe x_0 die Zahl der Elemente im Taylor-Polynom ist, desto niedriger ist die Ordnung des Residuums.

Funktionen TAYLR, TAYLR0 und SERIES

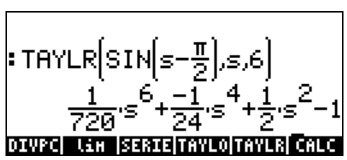
Die Funktionen TAYLR, TAYLR0 und SERIES werden zum Erzeugen von Taylor-Polynomen sowie für Taylor-Reihen mit Residuen verwendet. Diese Funktionen sind im Menü CALC/LIMITS&SERIES verfügbar, das bereits in diesem Kapitel beschrieben wurde.

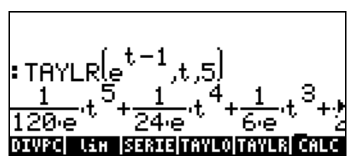
Die Funktion TAYLOR0 ergibt eine MacLaurin-Entwicklung um X = 0 eines Ausdrucks in der unabhängigen Standardvariablen VX (in der Regel 'X'). Bei der Entwicklung wird eine relative Potenz 4. Ordnung verwendet, d. h., die Differenz zwischen der höchsten und niedrigsten Potenz beträgt 4. Beispiel:





Die Funktion TAYLR ergibt die Taylor-Entwicklung der Funktion einer beliebigen Variablen x um einen Punkt x = a für die vom Benutzer angegebene Ordnung k. Somit hat die Funktion das Format TAYLR(f(x-a),x,k). Beispiel:





Die Funktion SERIES ergibt ein Taylor-Polynom, wobei als Argumente die zu entwickelnde Funktion f(x), nur ein Variablenname (bei MacLaurin-Reihen) oder ein Ausdruck der Form 'Variable = Wert', der den Entwicklungspunkt der Taylor-Reihe angibt, und die Ordnung der zu erzeugenden Reihe verwendet werden. Die Funktion SERIES gibt zwei Ausgabeobjekte zurück: Eine Liste mit vier Elementen und ein Ausdruck für h = x - a, wenn das zweite Argument im Funktionsaufruf 'x=a' lautet, also ein Ausdruck für das Inkrement h ist. Die als erstes Ausgabeobjekt zurückgegebene Liste enthält die folgenden Elemente:

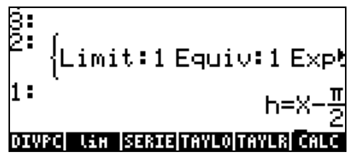
1 -Einen bidirektionalen Grenzwert der Funktion am Entwicklungspunkt, d. h. $\lim_{x \to a} f(x)$

2 - Einen Äquivalenzwert der Funktion nahe x = a

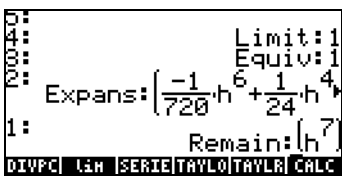
3 - Einen Ausdruck für das Taylor-Polynom

4 - Die Ordnung des Residuums bzw. Restes

Wegen der relativ umfangreichen Ausgabe ist diese Funktion im RPN-Modus leichter zu handhaben. Beispiel:



Entfernen Sie den Inhalt der Ebene 1 des Stacks, indem Sie ⬅ drücken, und geben Sie dann EVAL ein, um die Liste in ihre Bestandteile zu zerlegen. Die Ergebnisse lauten wie folgt:



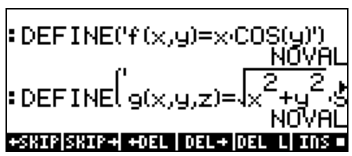
In der Abbildung rechts oben wird der Befehlszeileneditor verwendet, um die Reihenentwicklung im Detail anzuzeigen.

Kapitel 14
Anwendungen multivariater Infinitesimalrechnung

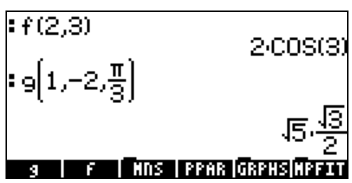
Die Bezeichnung „Multivariate Infinitesimalrechnung" bezieht sich auf Funktionen mit mindestens zwei Variablen. In diesem Kapitel werden die Grundkonzepte der multivariaten Infinitesimalrechnung einschließlich partieller Ableitungen und mehrfacher Integrale erläutert:

Multivariate Funktionen

Eine Funktion mit mindestens zwei Variablen kann im Taschenrechner mit der Funktion DEFINE (↰ DEF) definiert werden. Um das Konzept partieller Ableitungen zu veranschaulichen, definieren wir zwei multivariate Funktionen $f(x,y) = x \cos(y)$ und $g(x,y,z) = (x^2+y^2)^{1/2}\sin(z)$ wie folgt:



Wir können diese Funktionen wie jede andere Taschenrechnerfunktion berechnen, z. B.



Graphen zweidimensionaler Funktionen können mit Fast3D-, Wireframe-, Ps-Contour-, Y-Slice-, Gridmap- und Pr-Surface-Zeichnungen erstellt werden, wie in Kapitel 12 beschrieben.

Partielle Ableitungen

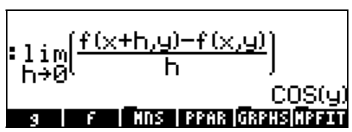
Betrachten Sie die Funktion mit zwei Variablen z = f(x,y). Die partielle Ableitung der Funktion für x ist definiert durch den Grenzwert

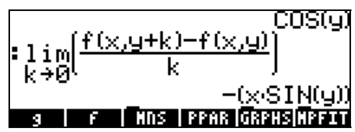
$$\frac{\partial f}{\partial x}=\lim_{h\to 0}\frac{f(x+h,y)-f(x,y)}{h} \quad .$$

Entsprechend ist

$$\frac{\partial f}{\partial y}=\lim_{k\to 0}\frac{f(x,y+k)-f(x,y)}{k}.$$

Wir verwenden die zuvor definierten multivariaten Funktionen, um mit diesen Definitionen partielle Ableitungen zu berechnen. Dies sind die Ableitungen von f(x,y) für x bzw. y:





Beachten Sie, dass die Definition der partiellen Ableitung für x beispielsweise erfordert, dass y unverändert bleibt, während als Grenzwert h→0 verwendet wird. Dies bietet eine Möglichkeit zur schnellen Berechnung partieller Ableitungen multivariater Funktionen: Verwenden Sie die Regeln gewöhnlicher Ableitungen für die relevante Variable, während alle anderen Variablen als Konstanten betrachtet werden. So ist z. B.

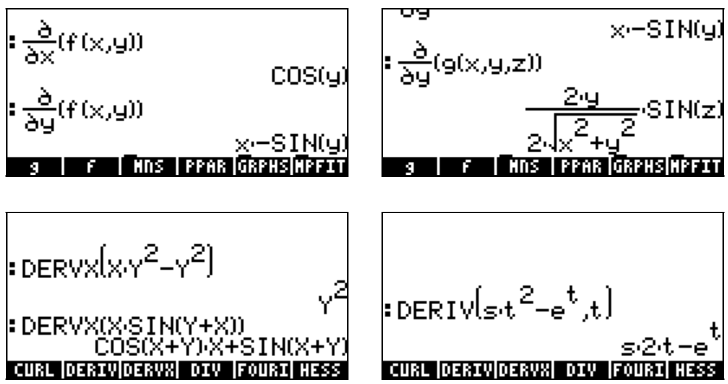
$$\frac{\partial}{\partial x}\left(x\cos(y)\right)=\cos(y),\ \frac{\partial}{\partial y}\left(x\cos(y)\right)=-x\sin(y),$$

wobei es sich um dieselben Ergebnisse wie bei den zuvor berechneten Grenzwerten handelt. Betrachten Sie ein weiteres Beispiel:

$$\frac{\partial}{\partial x}\left(yx^2+y^2\right)=2yx+0=2xy$$

Bei dieser Berechnung behandeln wir y als Konstante und berechnen Ableitungen des Ausdrucks für x.

Entsprechend können Sie die Ableitungsfunktionen des Taschenrechners, z. B. DERVX, DERIV und ∂ (in Kapitel 13 ausführlich beschrieben), zum Berechnen von partiellen Ableitungen verwenden. Entsinnen Sie sich, dass die Funktion DERVX die CAS-Standardvariable VX (in der Regel „X“) verwendet. Sie können daher mit DERVX nur Ableitungen für X berechnen. Im Folgenden werden einige Beispiele für partielle Ableitungen erster Ordnung dargestellt:



Ableitungen höherer Ordnung

Die folgenden Ableitungen höherer Ordnung können als

$$\frac{\partial^2 f}{\partial x^2} = \frac{\partial}{\partial x}\left(\frac{\partial f}{\partial x}\right), \frac{\partial^2 f}{\partial y^2} = \frac{\partial}{\partial y}\left(\frac{\partial f}{\partial y}\right),$$

$$\frac{\partial^2 f}{\partial y \partial x} = \frac{\partial}{\partial y}\left(\frac{\partial f}{\partial x}\right), \frac{\partial^2 f}{\partial x \partial y} = \frac{\partial}{\partial x}\left(\frac{\partial f}{\partial y}\right)$$

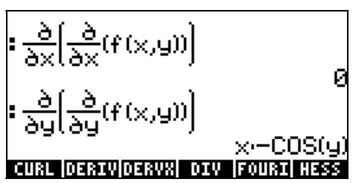
definiert werden. Die letzten beiden Ausdrücke stellen gemischte Ableitungen dar. Die Zeichen im Nenner für partielle Ableitungen geben die Ordnung der

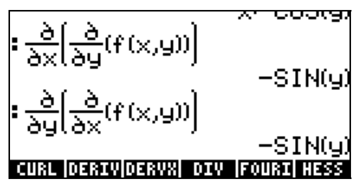
Ableitung an. Auf der linken Seite wird die Ableitung zunächst für x und dann für y berechnet, und auf der rechten Seite ist die Reihenfolge umgekehrt. Es ist wichtig darauf hinzuweisen, dass bei einer stetigen und differenzierbaren Funktion Folgendes gilt:

$$\frac{\partial^2 f}{\partial y \partial x} = \frac{\partial^2 f}{\partial x \partial y}.$$

Ableitungen dritter, vierter und höherer Ordnung werden auf ähnliche Weise definiert.

Um mit dem Taschenrechner Ableitungen höherer Ordnung zu berechnen, wenden Sie einfach die Ableitungsfunktion so häufig wie erforderlich an. Unten werden einige Beispiele dargestellt:



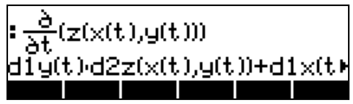


Die Kettenregel für partielle Ableitungen

Betrachten Sie die Funktion z = f(x,y), wobei x = x(t), y = y(t). Die Funktion z stellt eigentlich eine zusammengesetzte Funktion von *t* dar, wenn wir sie als z = f[x(t),y(t)] schreiben. In diesem Fall wird die Kettenregel für die Ableitung dz/dt wie folgt geschrieben:

$$\frac{\partial z}{\partial v} = \frac{\partial z}{\partial x} \cdot \frac{\partial x}{\partial v} + \frac{\partial z}{\partial y} \cdot \frac{\partial y}{\partial v}$$

Um den Ausdruck anzuzeigen, den der Taschenrechner für diese Version der Kettenregel erzeugt, geben Sie Folgendes ein:



Das Ergebnis wird durch d1y(t)·d2z(x(t),y(t))+d1x(t)·d1z(x(y),y(t)) ausgegeben. Der Ausdruck d1y(t) bedeutet „die Ableitung von y(t) für die erste unabhängige Variable, d. h. t" oder d1y(t) = dy/dt. Entsprechend ist d1x(t) = dx/dt. Andererseits bedeutet d1z(x(t),y(t)) „die erste Ableitung von z(x,y) für die erste unabhängige Variable, d. h. x" oder d1z(x(t),y(t)) = ∂z/∂x. Entsprechend ist d2z(x(t),y(t)) = ∂z/∂y. Daher muss der obige Ausdruck folgendermaßen interpretiert werden:

$$dz/dt = (dy/dt)\cdot(\partial z/\partial y) + (dx/dt)\cdot(\partial z/\partial x).$$

Totales Differenzial einer Funktion z = z(x,y)

Wenn wir die letzte Gleichung mit dt multiplizieren, erhalten wir das totale Differenzial der Funktion z = z(x,y), d. h. $dz = (\partial z/\partial x)\cdot dx + (\partial z/\partial y)\cdot dy$.

Eine andere Variante der Kettenregel wird angewendet, wenn z = f(x,y), x = x(u,v), y = y(u,v), sodass z = f[x(u,v), y(u,v)]. Die folgenden Formeln stellen die Kettenregel für diesen Fall dar:

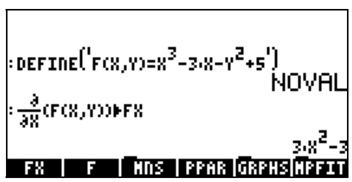
$$\frac{\partial z}{\partial u} = \frac{\partial z}{\partial x}\cdot\frac{\partial x}{\partial u} + \frac{\partial z}{\partial y}\cdot\frac{\partial y}{\partial u}, \qquad \frac{\partial z}{\partial v} = \frac{\partial z}{\partial x}\cdot\frac{\partial x}{\partial v} + \frac{\partial z}{\partial y}\cdot\frac{\partial y}{\partial v}$$

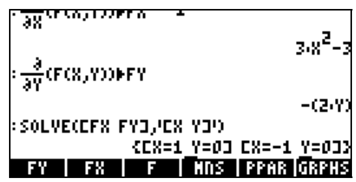
Bestimmen von Extremwerten von Funktionen mit zwei Variablen

Damit die Funktion z = f(x,y) bei (x_o,y_o) einen Extrempunkt (Extremwert) aufweist, müssen ihre Ableitungen ∂f/∂x und ∂f/∂y an diesem Punkt verschwinden. Dies sind die notwendigen Bedingungen. Die hinreichenden Bedingungen dafür, dass die Funktion am Punkt (x_o,y_o) einen Extremwert aufweist, sind ∂f/∂x = 0, ∂f/∂y = 0 und $\Delta = (\partial^2 f/\partial x^2)\cdot(\partial^2 f/\partial y^2)-[\partial^2 f/\partial x\partial y]^2 > 0$. Der Punkt (x_o,y_o) ist ein relatives Maximum, wenn $\partial^2 f/\partial x^2 < 0$, oder ein relatives Minimum, wenn $\partial^2 f/\partial x^2 > 0$. Der Wert Δ wird als Diskriminante bezeichnet.

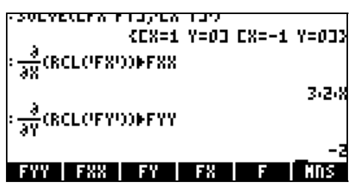
Wenn $\Delta = (\partial^2 f/\partial x^2)\cdot(\partial^2 f/\partial y^2)-[\partial^2 f/\partial x\partial y]^2 < 0$, liegt eine als *Sattelpunkt* bezeichnete Bedingung vor, wobei die Funktion bei x ein Maximum erreicht, wenn y als Konstante beibehalten wird, während gleichzeitig ein Minimum erreicht wird, wenn x als Konstante beibehalten wird, oder umgekehrt.

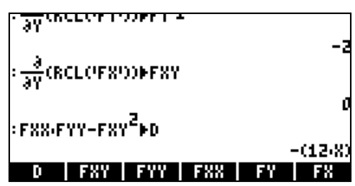
Beispiel 1 – Bestimmen Sie die Extrempunkte (sofern vorhanden) der Funktion $f(X,Y) = X^3\text{-}3X\text{-}Y^2+5$. Wir definieren zunächst die Funktion f(X/Y) und ihre Ableitungen $fX(X,Y) = \partial f/\partial X$, $fY(X,Y) = \partial f/\partial Y$. Anschließend lösen wir gleichzeitig die Gleichungen fX(X,Y) = 0 und fY(X,Y) = 0:



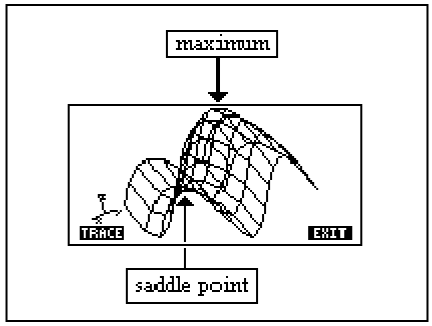


Wir finden kritische Punkte bei (X,Y) = (1,0) und (X,Y) = (-1,0). Zum Berechnen der Diskriminante berechnen wir die zweiten Ableitungen fXX(X,Y) $= \partial^2 f/\partial X^2$, $fXY(X,Y) = \partial^2 f/\partial X/\partial Y$ und $fYY(X,Y) = \partial^2 f/\partial Y^2$.





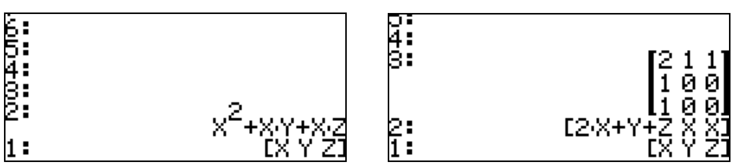
Das letzte Ergebnis gibt an, dass die Diskriminante Δ = -12X lautet. Für (X,Y) = (1,0) ist daher $\Delta<0$ (Sattelpunkt) und für (X,Y) = (-1,0) ist $\Delta>0$ und $\partial^2 f/\partial X^2<0$ (relatives Maximum). Die unten dargestellte, vom Taschenrechner erzeugte und mit dem Computer bearbeitete Abbildung veranschaulicht diese beiden Punkte:



Verwenden der Funktion HESS zum Berechnen von Extremwerten

Die Funktion HESS kann wie folgt zum Berechnen der Extremwerte einer Funktion zweier Variablen verwendet werden. Als Eingabe für die Funktion HESS werden generell eine Funktion mit n unabhängigen Variablen $\phi(x_1, x_2, \ldots, x_n)$ und ein Vektor der Funktionen ['x_1' 'x_2'… 'x_n'] verwendet. Die Funktion HESS gibt die Hesse-Matrix der Funktion ϕ zurück, definiert als Matrix $\mathbf{H} = [h_{ij}] = [\partial^2\phi/\partial x_i \partial x_j]$, den Gradienten der Funktion für n Variablen **grad** f = [$\partial\phi/\partial x_1$, $\partial\phi/\partial x_2$, … $\partial\phi/\partial x_n$] und die Liste der Variablen ['x_1' 'x_2'… 'x_n'].

Im RPN-Modus können Anwendungen der Funktion HESS einfacher grafisch dargestellt werden. Im folgenden Beispiel wenden wir die Funktion HESS auf die Funktion $\phi(X,Y,Z) = X^2 + XY + XZ$ an. Die Bildschirmabbildungen zeigen den RPN-Stack vor und nach der Anwendung der Funktion HESS.



Wenn der Gradient auf Ebene 2 auf eine Funktion mit zwei Variablen angewendet wird und gleich Null ist, stellt er die Gleichungen für kritische Punkte dar, d. h. $\partial\phi/\partial x_i = 0$, während die Matrix auf Ebene 3 Ableitungen zweiter Ordnung darstellt. Somit können die Ergebnisse der Funktion HESS zum Berechnen der Extremwerte von Funktionen mit zwei Variablen verwendet werden. Gehen Sie beispielsweise für die Funktion $f(X,Y) = X^3-3X-Y^2+5$ im RPN-Modus wie folgt vor:

'X^3-3*X-Y^2+5' [ENTER] ['X','Y'] [ENTER]	Funktion und Variablen eingeben
HESS	Funktion HESS anwenden
SOLVE	Kritische Punkte suchen
[EVAL]	Vektor zerlegen
's1' [STO▶] 's2' [STO▶]	Kritische Punkte speichern

Die Variablen s1 und s2 enthalten an dieser Stelle die Vektoren ['X=-1', 'Y=0'] bzw. ['X=1', 'Y=0']. Die Hesse-Matrix befindet sich an dieser Stelle auf Ebene 1.

'H' [STO▶]	Hesse-Matrix speichern
[VAR] [H] [S1] SUBST [↱] →NUM	s1 in H ersetzen

Die resultierende Matrix **A** besitzt Elemente $a_{11} = \partial^2\phi/\partial X^2 =$ -6., $a_{22} = \partial^2\phi/\partial X^2 =$ -2. und $a_{12} = a_{21} = \partial^2\phi/\partial X\partial Y = 0$. Die Diskriminante für diesen kritischen Punkt s1(-1,0) ist $\Delta = (\partial^2 f/\partial x^2)\cdot(\partial^2 f/\partial y^2)-[\partial^2 f/\partial x\partial y]^2 = (-6.)(-2.) = 12.0 > 0$. Da $\partial^2\phi/\partial X^2 < 0$ ist, stellt Punkt s1 ein relatives Maximum dar.

Anschließend ersetzen wir den zweiten Punkt s2 in H:

[VAR] [H] [S2] SUBST [↱] →NUM	s2 in H ersetzen

Die resultierende Matrix besitzt a_{11} Elemente $a_{11} = \partial^2\phi/\partial X^2 =$ 6., $a_{22} = \partial^2\phi/\partial X^2 =$ -2. und $a_{12} = a_{21} = \partial^2\phi/\partial X\partial Y = 0$. Die Diskriminante für diesen kritischen Punkt s2(1,0) ist $\Delta = (\partial^2 f/\partial x^2)\cdot(\partial^2 f/\partial y^2)-[\partial^2 f/\partial x\partial y]^2 = (6.)(-2.)$ $-12.0 < 0$ und gibt einen Sattelpunkt an.

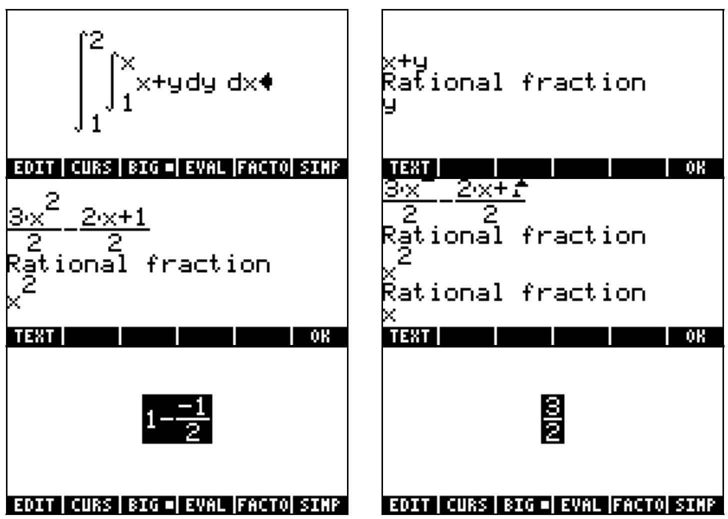
Mehrfache Integrale

Eine physische Interpretation eines normalen Integrals $\int_a^b f(x)dx$ ist die Fläche unter der Kurve y = f(x) und den x-Koordinaten x = a und x = b. Die

Erweiterung eines normalen Integrals auf drei Dimensionen ist ein doppeltes Integral einer Funktion f(x,y) über einem Bereich R auf der x-y-Fläche, das den Rauminhalt des Körpers unter der Fläche f(x,y) über dem Bereich R darstellt. Der Bereich R kann als $R = \{a<x<b, f(x)<y<g(x)\}$ oder als $R = \{c<y<d, r(y)<x<s(y)\}$ beschrieben werden. Somit kann das doppelte Integral wie folgt geschrieben werden:

$$\iint_R \phi(x,y)dA = \int_a^b \int_{f(x)}^{g(x)} \phi(x,y)dydx = \int_c^d \int_{r(y)}^{s(y)} \phi(x,y)dydx$$

Die Berechnung eines doppelten Integrals mit dem Taschenrechner ist unkompliziert. Ein doppeltes Integral kann in EquationWriter erzeugt werden (siehe das Beispiel in Kapitel 2). Ein Beispiel folgt. Dieses doppelte Integral wird direkt in EquationWriter berechnet, indem der ganze Ausdruck ausgewählt und die Funktion EVAL verwendet wird. Das Ergebnis ist 3/2. Wenn Sie im CAS MODES-Fenster die Option Step/Step auswählen, ist eine schrittweise Ausgabe möglich.



Jacobimatrix einer Koordinatentransformation

Betrachten Sie die Koordinatentransformation x = x(u,v), y = y(u,v). Die Jacobimatrix dieser Transformation wird definiert als

$$|J| = \det(J) = \det\begin{pmatrix} \frac{\partial x}{\partial u} & \frac{\partial x}{\partial v} \\ \frac{\partial y}{\partial u} & \frac{\partial y}{\partial v} \end{pmatrix}$$

Wenn Sie mit dieser Transformation ein Integral berechnen, lautet der zu verwendende Ausdruck $\iint_R \phi(x,y)dydx = \iint_{R'} \phi[x(u,v),y(u,v)]\,|J|\,dudv$,

wobei R' der durch die Koordinaten (u,v) angegebene Bereich R ist.

Doppeltes Integral in Polarkoordinaten

Zur Transformation von Polarkoordinaten zu kartesischen Koordinaten verwenden wir x(r,θ) = r cos θ und y(r, θ) = r sin θ. Somit lautet die Jacobimatrix der Transformation

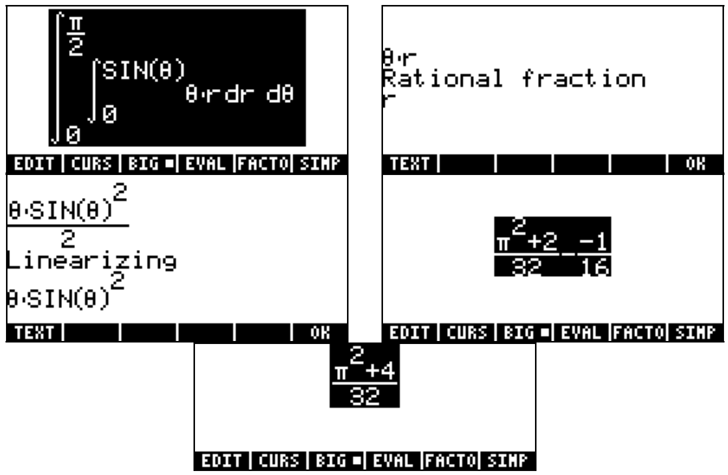
$$|J| = \begin{vmatrix} \frac{\partial x}{\partial r} & \frac{\partial x}{\partial \theta} \\ \frac{\partial y}{\partial r} & \frac{\partial y}{\partial \theta} \end{vmatrix} = \begin{vmatrix} \cos(\theta) & -r\cdot\sin(\theta) \\ \sin(\theta) & r\cdot\cos(\theta) \end{vmatrix} = r$$

Bei diesem Ergebnis werden Integrale in Polarkoordinaten als

$$\iint_{R'} \phi(r,\theta)dA = \int_\alpha^\beta \int_{f(\theta)}^{g(\theta)} \phi(r,\theta)rdrd\theta$$

geschrieben, wobei der Bereich R' in Polarkoordinaten R' = {α < θ < β, f(θ) < r < g(θ)} lautet.

Doppelte Integrale in Polarkoordinaten können in den Taschenrechner eingegeben werden, wenn sichergestellt wird, dass die Jacobimatrix |J| = r im Integranden enthalten ist. Es folgt ein Beispiel für ein doppeltes Integral, dessen Berechnung in Polarkoordinaten schrittweise angezeigt wird:



Kapitel 15
Anwendungen der Vektorrechnung

In diesem Kapitel stellen wir mehrere Funktionen im Menü CALC für die Berechnung von Skalar- und Vektorfeldern vor. Das Menü CALC wurde in Kapitel 13 ausführlich dargestellt. Wir haben insbesondere auf mehrere Funktionen im Menü DERIV&INTEG hingewiesen, die für die Vektorrechnung verwendet werden können, nämlich CURL, DIV, HESS und LAPL. Ändern Sie für die Übungen in diesem Kapitel das Winkelmaß in Radiant.

Definitionen

Eine für einen Raumbereich definierte Funktion, z. B. $\phi(x,y,z)$, wird als Skalarfeld bezeichnet. Beispiele hierfür sind Temperatur, Dichte und Spannung in der Nähe einer Ladung. Wenn die Funktion durch einen Vektor definiert ist, d. h. $\mathbf{F}(x,y,z) = f(x,y,z)\mathbf{i}+g(x,y,z)\mathbf{j}+h(x,y,z)\mathbf{k}$, wird sie als Vektorfeld bezeichnet.

Der folgende Operator, der als Del- oder Nabla-Operator bezeichnet wird, ist ein Vektoroperator, der auf eine Skalar- oder Vektorfunktion angewendet werden kann:

$$\nabla[\ ] = i \cdot \frac{\partial}{\partial x}[\ ] + j \cdot \frac{\partial}{\partial y}[\ ] + k \cdot \frac{\partial}{\partial z}[\ ]$$

Wenn dieser Operator auf eine Skalarfunktion angewendet wird, können wir den Gradienten der Funktion erhalten, und wenn er auf eine Vektorfunktion angewendet wird, können wir die Divergenz und die Rotation dieser Funktion erhalten. Die Kombination von Gradient und Divergenz ergibt einen weiteren Operator, der als Laplace-Operator einer Skalarfunktion bezeichnet wird. Diese Operationen werden nun dargestellt.

Gradient und Richtungsableitung

Der Gradient einer Skalarfunktion $\phi(x,y,z)$ ist eine Vektorfunktion, die durch

$$grad\phi = \nabla\phi = i \cdot \frac{\partial\phi}{\partial x} + j \cdot \frac{\partial\phi}{\partial y} + k \cdot \frac{\partial\phi}{\partial z}$$

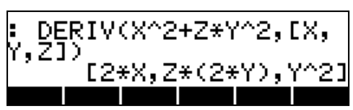
definiert ist. Das skalare Produkt des Gradienten einer Funktion mit einem bestimmten Einheitsvektor stellt die Änderungsrate der Funktion entlang diesem bestimmten Vektor dar.Diese Änderungsrate wird als Richtungsableitung $D_u\phi(x,y,z) = \mathbf{u} \bullet \nabla\phi$ der Funktion bezeichnet.

Die maximale Änderungsrate der Funktion erfolgt an jedem beliebigen Punkt in der Richtung des Gradienten, d. h. entlang einem Einheitsvektor $\mathbf{u} = \nabla\phi/|\nabla\phi|$.

Der Wert der Richtungsableitung ist gleich dem Betrag des Gradienten an einem beliebigen Punkt $D_{max}\phi(x,y,z) = \nabla\phi \bullet \nabla\phi/|\nabla\phi| = |\nabla\phi|$.

Die Gleichung $\phi(x,y,z) = 0$ stellt eine Fläche im Raum dar. Der Gradient der Funktion ist an jedem Punkt der Fläche senkrecht zur Fläche. Daher kann die Gleichung der Tangentialebene zur Kurve an diesem Punkt mit der in Kapitel 9 dargestellten Methode ermittelt werden.

Die im Menü CALC verfügbare Funktion DERIV stellt die einfachste Möglichkeit dar, den Gradienten zu erhalten, z. B.

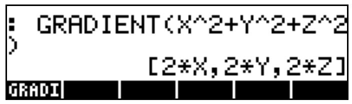


Ein Programm zum Berechnen des Gradienten

Das folgende Programm, das Sie in der Variablen GRADIENT speichern können, verwendet die Funktion DERIV zum Berechnen des Gradienten einer Skalarfunktion von X,Y,Z. Berechnungen für andere Basisvariablen sind nicht möglich. Wenn Sie häufig mit (X,Y,Z) arbeiten, erleichtert diese Funktion jedoch die Berechnungen:

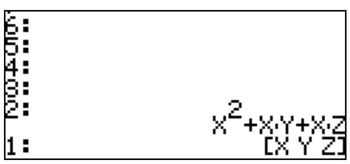
<< X Y Z 3 →ARRY DERIV >>

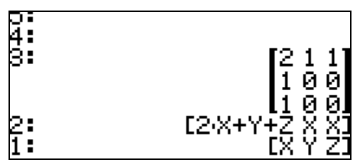
Geben Sie das Programm im RPN-Modus ein. Nachdem Sie den ALG-Modus gestartet haben, können Sie die Funktion GRADIENT wie im folgenden Beispiel aufrufen:



Verwenden der Funktion HESS zum Erhalten des Gradienten

Mit der Funktion HESS können Sie den Gradienten einer Funktion wie im Folgenden dargestellt erhalten. Wie in Kapitel 14 erläutert, wird als Eingabe für die Funktion HESS eine Funktion mit n unabhängigen Variablen $\phi(x_1, x_2, \ldots, x_n)$ und ein Vektor der Funktionen [„x_1“ „x_2“…„x_n“] verwendet. Die Funktion HESS gibt die Hesse-Matrix der Funktion ϕ zurück, definiert als die Matrix $\mathbf{H} = [h_{ij}] = [\partial\phi/\partial x_i \partial x_j]$, den Gradienten der Funktion für n Variablen **grad** $f = [\ \partial\phi/\partial x_1,\ \partial\phi/\partial x_2\ ,\ \ldots\ \partial\phi/\partial x_n]$ und die Liste der Variablen [„x_1“ „x_2“…„x_n“]. Im folgenden Beispiel wenden wir im RPN-Modus die Funktion HESS auf das Skalarfeld $\phi(X,Y,Z) = X^2 + XY + XZ$ an:



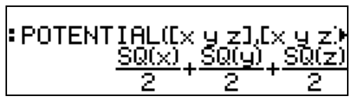


Somit ist der Gradient [2X+Y+Z, X, X]. Alternativ kann man die Funktion DERIV wie folgend verwenden, um das selbe Ergebnis zu erhalten: DERIV(X^2+X*Y+X*Z,[X,Y,Z]).

Potential eines Gradienten

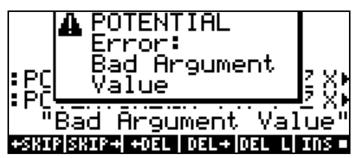
Wenn ein Vektorfeld $\mathbf{F}(x,y,z) = f(x,y,z)\mathbf{i}+g(x,y,z)\mathbf{j}+h(x,y,z)\mathbf{k}$ und eine Funktion $\phi(x,y,z)$ gegeben sind, sodass $f = \partial\phi/\partial x$, $g = \partial\phi/\partial y$ und $h = \partial\phi/\partial z$, dann wird $\phi(x,y,z)$ als Potentialfunktion für das Vektorfeld **F** bezeichnet. Es folgt, dass $\mathbf{F} = \text{grad}\ \phi = \nabla\phi$.

Der Taschenrechner enthält die Funktion POTENTIAL, die über den Befehlskatalog (→ _CAT) verfügbar ist, um die Potentialfunktion eines Vektorfeldes zu berechnen, sofern vorhanden. Wenn beispielsweise **F**(x,y,z) = **xi** + y**j** + z**k** ist, ergibt sich durch Anwenden der Funktion POTENTIAL Folgendes:



Da die Funktion SQ(x) den Wert x^2 darstellt, gibt dieses Ergebnis an, dass die Potentialfunktion für das Vektorfeld **F**(x,y,z) = x**i** + y**j** + z**k** durch die Gleichung $\phi(x,y,z) = (x^2+y^2+z^2)/2$ dargestellt wird.

Beachten Sie, dass die Bedingungen für das Vorhandensein von $\phi(x,y,z)$, nämlich $f = \partial\phi/\partial x$, $g = \partial\phi/\partial y$ und $h = \partial\phi/\partial z$, mit den Bedingungen $\partial f/\partial y = \partial g/\partial x$, $\partial f/\partial z = \partial h/\partial x$ und $\partial g/\partial z = \partial h/\partial y$ äquivalent sind. Anhand dieser Bedingungen lässt sich schnell bestimmen, ob das Vektorfeld über eine entsprechende Potentialfunktion verfügt. Wenn eine der Bedingungen $\partial f/\partial y = \partial g/\partial x$, $\partial f/\partial z = \partial h/\partial x$, $\partial g/\partial z = \partial h/\partial y$ nicht zutrifft, ist die Potentialfunktion $\phi(x,y,z)$ nicht vorhanden. In diesem Fall gibt die Funktion POTENTIAL eine Fehlermeldung zurück. Beispielsweise weist das Vektorfeld **F**(x,y,z) = (x+y)**i** + (x-y+z)**j** + xz**k** keine entsprechende Potentialfunktion auf, da $\partial f/\partial z \neq \partial h/\partial x$. Die Meldung des Taschenrechners in diesem Fall wird unten dargestellt:

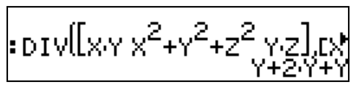


Divergenz

Die Divergenz einer Vektorfunktion **F**(x,y,z) = f(x,y,z)**i**+g(x,y,z)**j**+h(x,y,z)**k** wird als skalares Produkt des Del-Operators mit der Funktion bezeichnet, d. h.

$$divF = \nabla \bullet F = \frac{\partial f}{\partial x} + \frac{\partial g}{\partial y} + \frac{\partial h}{\partial z}$$

Die Divergenz eines Vektorfeldes kann mit der Funktion DIV berechnet werden. Beispielsweise wird für **F**(X,Y,Z) = [XY,X^2+Y^2+Z^2,YZ] die Divergenz wie folgt im ALG-Modus berechnet:

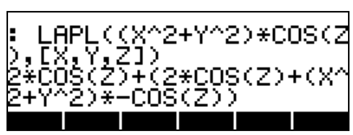


Laplace-Operator

Die Divergenz des Gradienten einer Skalarfunktion ergibt einen Operator, der als Laplace-Operator bezeichnet wird. Der Laplace-Operator einer Skalarfunktion ϕ(x,y,z) wird somit durch

$$\nabla^2\phi = \nabla \bullet \nabla\phi = \frac{\partial^2\phi}{\partial x^2} + \frac{\partial^2\phi}{\partial x^2} + \frac{\partial^2\phi}{\partial x^2}$$

angegeben. Die partielle Differenzialgleichung $\nabla^2\phi = 0$ wird als Laplace-Gleichung bezeichnet.
Der Laplace-Operator einer Skalarfunktion kann mit der Funktion LAPL berechnet werden. Geben Sie beispielsweise zum Berechnen des Laplace-Operators der Funktion ϕ(X,Y,Z) = (X^2+Y^2)cos(Z) Folgendes ein:



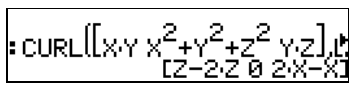
Rotation

Die Rotation eines Vektorfeldes **F**(x,y,z) = f(x,y,z)**i**+g(x,y,z)**j**+h(x,y,z)**k** wird durch das Kreuzprodukt des Del-Operators mit dem Vektorfeld definiert, d. h.

$$
curl\mathbf{F} = \nabla \times \mathbf{F} = \begin{vmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ \dfrac{\partial}{\partial x}[\;] & \dfrac{\partial}{\partial y}[\;] & \dfrac{\partial}{\partial z}[\;] \\ f(x,y,z) & g(x,y,z) & h(x,y,z) \end{vmatrix}
$$

$$
= \mathbf{i}\left(\frac{\partial h}{\partial y} - \frac{\partial g}{\partial z}\right) + \mathbf{j}\left(\frac{\partial f}{\partial z} - \frac{\partial h}{\partial x}\right) + \mathbf{k}\left(\frac{\partial h}{\partial y} - \frac{\partial g}{\partial z}\right)
$$

Die Rotation des Vektorfeldes kann mit der Funktion CURL berechnet werden. Beispielsweise wird für die Funktion **F**(X,Y,Z) = [XY,X^2+Y^2+Z^2,YZ] die Rotation wie folgt berechnet:



Rotationsfreie Felder und Potentialfunktion

In einem vorherigen Abschnitt dieses Kapitels haben wir die Funktion POTENTIAL vorgestellt, um die Potentialfunktion φ(x,y,z) für ein Vektorfeld **F**(x,y,z) = f(x,y,z)**i**+ g(x,y,z)**j**+ h(x,y,z)**k** zu berechnen, sodass **F** = grad φ = ∇φ. Wir haben auch angegeben, dass die Bedingungen für das Vorhandensein von φ wie folgt lauten: ∂f/∂y = ∂g/∂x, ∂f/∂z = ∂h/∂x und ∂g/∂z = ∂h/∂y. Diese Bedingungen sind mit folgendem Vektorausdruck äquivalent:

curl **F** = ∇×**F** = 0.

Ein Vektorfeld **F**(x,y,z) mit der Rotation Null wird als <u>rotationsfreies</u> Feld bezeichnet. Daraus schließen wir, dass für eine Potentialfunktion φ(x,y,z) stets ein rotationsfreies Feld **F**(x,y,z) vorhanden ist.

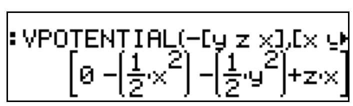
In einem vorherigen Beispiel haben wir versucht, eine Potentialfunktion für das Vektorfeld **F**(x,y,z) = (x+y)**i** + (x-y+z)**j** + xz**k** zu finden und haben eine von der Funktion POTENTIAL zurückgegebene Fehlermeldung erhalten. Um zu überprüfen, ob es sich hierbei um ein Rotationsfeld handelt (d. h. $\nabla\times\mathbf{F} \neq 0$), verwenden wir für dieses Feld die Funktion CURL:

Andererseits ist das Vektorfeld **F**(x,y,z) = **xi** + y**j** + z**k** tatsächlich rotationsfrei, wie unten gezeigt:

Vektorpotential

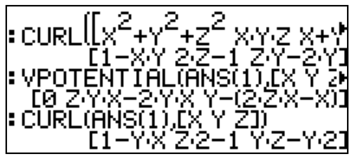
Wenn für ein Vektorfeld **F**(x,y,z) = f(x,y,z)**i**+g(x,y,z)**j**+h(x,y,z)**k** eine Vektorfunktion Φ(x,y,z) = φ(x,y,z)**i**+ψ(x,y,z)**j**+η(x,y,z)**k** vorhanden ist, sodass **F** = curl Φ = ∇× Φ, wird die Funktion Φ(x,y,z) als Vektorpotential von **F**(x,y,z) bezeichnet.

Der Taschenrechner enthält die über den Befehlskatalog (CAT) verfügbare Funktion VPOTENTIAL, um das Vektorpotential Φ(x,y,z) zu berechnen, wenn das Vektorfeld **F**(x,y,z) = f(x,y,z)**i**+g(x,y,z)**j**+h(x,y,z)**k** vorhanden ist. Für das Vektorfeld **F**(x,y,z) = -(y**i**+z**j**+x**k**) ergibt VPOTENTIAL beispielsweise



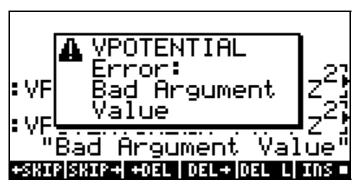
d. h. $\Phi(x,y,z) = -x^2/2\mathbf{j} + (-y^2/2+zx)\mathbf{k}$.

Es sollte darauf hingewiesen werden, dass für ein Vektorfeld **F** mehrere Vektorpotentialfunktionen Φ vorhanden sind. In der folgenden Bildschirmabbildung wird beispielsweise gezeigt, dass die Rotation der Vektorfunktion $\Phi_1 = [X^2+Y^2+Z^2,XYZ,X+Y+Z]$ durch den Vektor $\mathbf{F} = \nabla\times \Phi_2 = [1-XY,2Z-1,ZY-2Y]$ dargestellt wird. Durch Anwendung der Funktion VPOTENTIAL wird die Vektorpotentialfunktion Φ_2 = [0, ZYX-2YX, Y-(2ZX-X)] erstellt, die sich von Φ_1 unterscheidet. Der letzte Befehl in der Bildschirmabbildung zeigt, dass tatsächlich **F** = $\nabla\times\ \Phi_2$. Somit ist eine Vektorpotentialfunktion nicht eindeutig bestimmt.

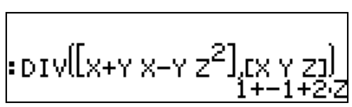


Die Beziehung der Komponenten des Vektorfeldes **F**(x,y,z) = f(x,y,z)**i**+g(x,y,z)**j** +h(x,y,z)**k** und der Vektorpotentialfunktion Φ(x,y,z) = ϕ(x,y,z)**i**+ψ(x,y,z)**j**+η(x,y,z)**k** wird durch f = $\partial\eta/\partial y$ - $\partial\psi/\partial x$, g = $\partial\phi/\partial z$ - $\partial\eta/\partial x$ und h = $\partial\psi/\partial x$ - $\partial\phi/\partial y$ bestimmt.

Als Bedingung für das Vorhandensein der Funktion Φ(x,y,z) ist **F** = ∇•**F** = 0, d. h. $\partial f/\partial x + \partial g/\partial y + \partial f/\partial z = 0$. Wenn daher diese Bedingung nicht erfüllt ist, ist die Vektorpotentialfunktion Φ(x,y,z) nicht vorhanden. Beispielsweise gibt die Funktion VPOTENTIAL für **F** = [X+Y,X-Y,Z^2] eine Fehlermeldung zurück, da die Funktion F die Bedingung ∇•**F** = 0 nicht erfüllt:



Die Bedingung ∇•**F** $\neq$ 0 wird in der folgenden Bildschirmabbildung überprüft:



Kapitel 16
Differentialgleichungen

In diesem Kapitel stellen wir Beispiele zur Lösung gewöhnlicher Differentialgleichungen (ODE) mithilfe der Rechnerfunktionen vor. Eine Differentialgleichung ist eine Gleichung, die Ableitungen der unabhängigen Variable einschließt. In den meisten Fällen suchen wir die abhängige Funktion, welche die Differentialgleichung erfüllt.

Grundfunktionen mit Differentialgleichungen

In diesem Abschnitt stellen wir einige Anwendungen des Taschenrechners zur Eingabe, Überprüfung und Anzeige von ODE-Lösungen vor.

Differentialgleichungen eingeben

Der Schlüssel zur Verwendung von Differentialgleichungen im Taschenrechner ist die Eingabe der Ableitungen der Gleichung. Der einfachste Weg eine Differentialgleichung einzugeben, ist die Eingabe in den Gleichungslöser. Zur Eingabe der folgenden ODE schreiben Sie z. B.:
$(x-1)\cdot(dy(x)/dx)^2 + 2\cdot x\cdot y(x) = e^x \sin x$, verwenden Sie:

[↱] EQW [↰] () [ALPHA] [↰] [X] [−] [1] [▶] [▶] [▶] [×] [↱] ∂ [ALPHA] [↰] [X]
[▶] [ALPHA] [↰] [Y] [↰] () [ALPHA] [↰] [X] [▶] [▶] [Y^x] [2] [▶] [▶] [+] [2] [×]
[ALPHA] [↰] [X] [×] [ALPHA] [↰] [Y] [↰] () [ALPHA] [↰] [X] [▶] [▶] [▶] [▶]
[↱] = [↰] e^x [ALPHA] [↰] [X] [▶] [×] [SIN] [ALPHA] [↰] [X] [ENTER]

Die Ableitung dy/dx wird dargestellt durch `∂x(y(x))` oder durch `d1y(x)`. Für Lösungs- oder Berechnungszwecke muss y(x) im Ausdruck spezifiziert sein, z. B. die abhängige Variable muss in jeder Ableitung in der Gleichung ihre unabhängige Variable(n) enthalten.

Sie können eine Gleichung auch direkt in den Speicher eingeben, indem Sie das Symbol ∂ in den Ableitungen verwenden. Zur Eingabe der folgenden ODE mit Ableitungen zweiter Ordnung: $d^2u(x)/dx^2 + 3u(x)\cdot(du(x)/dx) + u(x)^2 = 1/x$ direkt in den Schirm, verwenden Sie:

['] [↱] ∂ [ALPHA] [↰] [X] [↰] () [↱] ∂ [ALPHA] [↰] [X] [↰] () [ALPHA] [↰] [U]

← () ALPHA ← X ▶ ▶ ▶ + 3 × ALPHA ← U ← ()
ALPHA ← X ▶ × → ∂ ALPHA ← X ← () ALPHA ← U ← ()
ALPHA ← X ▶ ▶ + ALPHA ← U ← () ALPHA ← X ▶ Y^x 2
→ = 1 ÷ ALPHA ← X ENTER

Das Ergebnis lautet: ‘`∂x(∂x(u(x)))+3*u(x)*∂x(u(x))+u^2=1/x` ’. Dieses Format wird auf dem Display angezeigt, wenn die Option _Textbook in der Display-Einstellung (MODE DISP) nicht aktiviert ist. Drücken Sie ▼ zum Anzeigen der Gleichung im Gleichungsschreiber.

Eine alternative Notation für direkt in den Speicher eingegebene Ableitungen ist die Verwendung von ‘d1’ für die Ableitung nach der ersten unabhängigen Variable, ‘d2’ für die Ableitung nach der zweiten unabhängigen Variable, usw. Eine Ableitung zweiter Ordnung wäre z. B. d^2x/dt^2, wobei x = x(t), als ‘d1d1x(t)’ geschrieben würde und $(dx/dt)^2$ als ‘d1x(t)^2’. Somit würde die PDE $\partial^2 y/\partial t^2 - g(x,y)\cdot (\partial^2 y/\partial x^2)^2 = r(x,y)$, mithilfe dieser Notaiton als ‘d2d2y(x,t)-g(x,y)*d1d1y(x,t)^2=r(x,y)’ geschrieben werden.

Die Notation, die ‘d’ und die Ordnung der unabhängigen Variable verwendet, wird vom Taschenrechner bevorzugt, wenn eine Berechnung Ableitungen umfasst. Zum Beispiel führt die Verwendung von DERIV, im ALG-Modus, wie neben DERIV(‘x*f(x,t)+g(t,y) = h(x,y,t)’,t) angezeigt, zum folgenden Ausdruck: `x*d2f(x,t)+d1g(t,y)=d3h(x,y,t)`’. Auf Papier repräsentiert dieser Ausdruck die partielle Differentialgleichung $x\cdot(\partial f/\partial t) + \partial g/\partial t = \partial h/\partial t$.

Weil die Ordnung der Variable t in f(x,t), g(t,y), und h(x,y,t) verschieden ist, haben Ableitungen nach t verschiedene Indizes, d.h. d2f(x,t), d1g(t,y), und d3h(x,y,t). Sie sind jedoch alles Ableitungen nach derselben Variable.

Ausdrücke nach Ableitungen, welche die Indexnotations-Ordnung der Variable verwenden, übersetzen sich nicht in Ableitungsnotationen im Gleichungsschreiber, wie Sie durch Drücken von ▼ überprüfen können, solange das letzte Ergebnis in der Speicherebene 1 enthalten ist. Der Taschenrechner versteht jedoch beide Notationen und geht dementsprechend nach der verwendeten Notation vor.

Lösungen im Taschenrechner überprüfen

Um zu überprüfen, ob eine Funktion eine bestimmte Gleichung unter Verwendung des Taschenrechners erfüllt, verwenden Sie die Funktion SUBST (siehe Kapitel 5), um die Lösung in der Form 'y = f(x)' oder 'y = f(x,t)', usw. in der Differentialgleichung zu ersetzen. Möglicherweise müssen Sie das Ergebnis mithilfe von EVAL vereinfachen, um die Lösung zu bestätigen. Um z. B. zu überprüfen, ob $u = A \sin \omega_o t$ eine Lösung der Gleichung $d^2u/dt^2 + \omega_o^2 \cdot u = 0$ ist, verwenden Sie Folgendes:

Im ALG-Modus:

SUBST('∂t(∂t(u(t)))+ω0^2*u(t) = 0','u(t)=A*SIN (ω0*t)') ENTER
EVAL(ANS(1)) ENTER

Im RPN-Modus:

'∂t(∂t(u(t)))+ω0^2*u(t) = 0' ENTER 'u(t)=A*SIN (ω0*t)' ENTER
SUBST EVAL

Das Ergebnis ist '0=0'.

Für dieses Beispiel könnten Sie auch Folgendes verwenden:
'∂t(∂t(u(t))))+ ω0^2*u(t) = 0' um die Differentialgleichung einzugeben.

Lösungen als Steigungsfeld anzeigen

Steigungsfeld-Anzeigen, die in Kapitel 12 näher erklärt werden, werden verwendet, um die Lösung einer Differentialgleichung der Form dy/dx = f(x,y) anzuzeigen. Eine Steigungsfeld-Anzeige zeigt eine Reihe von Segmenten tangential zur Lösungskurve y = f(x). Die Steigung des Segments wird an jedem Punkt (x,y) durch dy/dx = f(x,y) angezeigt, bei jedem Punkt (x,y) ausgewertet und stellt die Steigung der Tangentengeraden bei Punkt (x,y) dar.

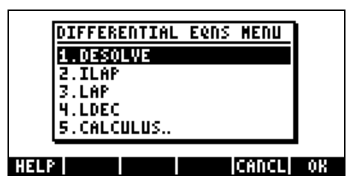
<u>Beispiel 1</u> – Berechnen Sie die Lösung zur Differentialgleichung y' = f(x,y) = sin x cos y mithilfe einer Steigungsfeld-Anzeige. Um dieses Problem zu lösen, befolgen Sie die Anweisungen in Kapitel 12 für *Steigungsfeld*-Zeichnungen.

Wenn Sie die Steigungsfeld-Zeichnung manuell nachzeichnen könnten, könnten Sie mit der Hand die Linien verfolgen, die zu den in der Zeichnung gezeigten Liniensegmenten tangential verlaufen. Diese Linien bilden die Linien von y(x,y) = konstant für die Lösung von y' = f(x,y). Steigungsfelder sind nützliche Hilfsmittel bei der Anzeige außergewöhnlich schwierig zu lösender Gleichungen.

Zusammenfassend sind Steigungsfelder grafische Hilfsmittel, um die Kurven von y = g(x) anzuzeigen, die den Lösungen der Differentialgleichung dy/dx = f(x,y) entsprechen.

Das Menü CALC/DIFF

Das Untermenü DIFFERENTIAL EQNS im Menü CALC (CALC) enthält Funktionen zur Lösung von Differentialgleichungen. Das Menü ist unten mit dem Systemflag 117 aufgelistet. Das Flag setzen, um Kästchen auszuwählen (CHOOSE boxes).



Diese Funktionen werden nachfolgend kurz beschrieben. Später in diesem Kapitel werden Sie eingehender beschrieben.

DESOLVE: Die Differentialgleichung SOLVEr führt zu einer Lösung, falls möglich.
ILAP: Inverse LAPlace-Transformation, $L^{-1}[F(s)] = f(t)$
LAP: LAPlace-Transformation, L[f(t)]=F(s)
LDEC: löst lineare Differentialgleichungen mit konstanten Koeffizienten, einschließlich System- oder Differentialgleichungen mit konstanten Koeffizienten.

Lösung linearer und nicht-linearer Gleichungen

Eine Gleichung, in der die abhängige Variable und all deren relevante Ableitungen ersten Grades sind, wird lineare Differentialgleichung genannt. Anderenfalls wird die Gleichung als nicht-linear bezeichnet. Beispiele für lineare Differentialgleichungen sind: $d^2x/dt^2 + \beta\cdot(dx/dt) + \omega_o\cdot x = A \sin \omega_f t$, und $\partial C/\partial t + u\cdot(\partial C/\partial x) = D\cdot(\partial^2 C/\partial x^2)$.

Eine Gleichung, dessen rechter Teil (die Funktion oder dessen Ableitungen ausgeschlossen) gleich Null ist, wird als homogene Gleichung bezeichnet. Anderenfalls wird sie als nicht-homogen bezeichnet. Die Lösung der homogenen Gleichung ist als allgemeine Lösung bekannt. Eine bestimmte Lösung erfüllt die nicht-homogene Gleichung.

Funktion LDEC

Der Taschenrechner stellt die Funktion LDEC (Befehl Lineare Differentialgleichung) zur Verfügung, um die allgemeine Lösung einer linearen ODE jeder beliebigen Ordnung mit konstanten Koeffizienten zu finden, ob sie homogen ist oder nicht. Diese Funktion erfordert zwei verschiedene Eingaben von Ihrer Seite:

- den rechten Teil der ODE
- die charakteristische Gleichung der ODE

Beide Eingaben müssen im Hinblick auf die vorgegebene unabhängige Variable für die CAS (normalerweise 'X') des Taschenrechners erfolgen. Die Ausgabe der Funktion ist die allgemeine Lösung der ODE. Die Funktion LDEC ist über das Menü CALC/DIFF verfügbar. Die Beispiele sind im RPN-Modus angeführt, sie können jedoch problemlos in den ALG-Modus übertragen werden.

Beispiel 1 – So lösen Sie die homogene ODE: $d^3y/dx^3-4\cdot(d^2y/dx^2)-11\cdot(dy/dx)+30\cdot y = 0$, geben Sie ein: `0` ENTER `'X^3-4*X^2-11*X+30'` ENTER `LDEC`. Die Lösung hierfür lautet:

$$-\frac{6\cdot cC0-(cC1+cC2)}{24}\cdot e^{5\cdot X}+\frac{10\cdot cC0-(7\cdot cC1-cC2)}{40}\cdot e^{-(3\cdot X)}+\frac{15\cdot cC0+2\cdot cC1-cC2}{15}\cdot e^{2\cdot X}$$

wobei cC0, cC1, und cC2 Integrationskonstanten sind. Dieses Ergebnis scheint sehr kompliziert zu sein und kann vereinfacht werden durch Verwendung von

K1 = (10*cC0-(7+cC1-cC2))/40, K2 = -(6*cC0-(cC1+cC2))/24,

und

K3 = (15*cC0+(2*cC1-cC2))/15.

Die Lösung lautet dann:

$$y = K_1 \cdot e^{-3x} + K_2 \cdot e^{5x} + K_3 \cdot e^{2x}.$$

Der Grund für die komplizierte Kombination von Konstanten über LDEC ist, dass LDEC für die Errechnung der Lösung intern Laplace-Transformationen verwendet (später in diesem Kapitel angeführt), welche die Lösung einer ODE in eine algebraische Lösung umwandeln. Die Kombination von Konstanten ist auf die Ausklammerung der Exponentialterme, nach Erhalt der Lösung für die Laplace-Transformation, zurückzuführen.

<u>Beispiel 2</u> – Mithilfe der Funktion LDEC werden nicht-homogene ODE gelöst:

$$d^3y/dx^3-4\cdot(d^2y/dx^2)-11\cdot(dy/dx)+30\cdot y = x^2.$$

Geben Sie Folgendes ein:

`'X^2'` ENTER `'X^3-4*X^2-11*X+30'` ENTER `LDEC`

Die Lösung, die hier teilweise im Gleichungsschreiber angezeigt wird, lautet:

$$-\frac{750\cdot cC0-(125\cdot cC1+125\cdot cC2+2)}{3000}\cdot e^{5\cdot X}+\frac{270\cdot cC0-(189\cdot cC1-(27\cdot cC2-2))}{1080}\cdot e\cdot e^{-(3\cdot X)}+\frac{450\cdot X^2+330\cdot X+241}{13500}\cdot e^{(}$$

Wenn Sie die Kombination der Konstanten, welche die Exponentialterme begleiten, mit einfachen Werten ersetzen, kann der Ausdruck vereinfacht werden und lautet $y = K_1 \cdot e^{-3x} + K_2 \cdot e^{5x} + K_3 \cdot e^{2x} + (450\cdot x^2+330\cdot x+241)/13500$.

Wir erkennen die ersten drei Terme als allgemeine Lösung der homogenen Gleichung (siehe Bsp. 1 oben). Wenn y_h die Lösung für die homogene Gleichung darstellt, d.h. $y_h = K_1 \cdot e^{-3x} + K_2 \cdot e^{5x} + K_3 \cdot e^{2x}$, können Sie beweisen, dass die verbleibenden Terme in der oben angeführten Lösung, d.h. y_p = $(450 \cdot x^2+330 \cdot x+241)/13500$ eine besondere Lösung der ODE darstellen.

Hinweis: Dieses Ergebnis ist allgemein für alle nicht-homogenen linearen ODE, d.h. vorausgesetzt die Lösung der homogenen Gleichung $y_h(x)$, die Lösung der entsprechenden nicht-homogenen Gleichung $y(x)$, kann als

$$y(x) = y_h(x) + y_p(x),$$

geschrieben werden, wobei $y_p(x)$ eine besondere Lösung der ODE ist.

Um zu beweisen, dass $y_p = (450 \cdot x^2+330 \cdot x+241)/13500$, tatsächlich eine besondere Lösung der ODE ist, verwenden Sie Folgendes:

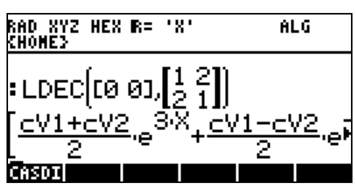
```
'd1d1d1Y(X)-4*d1d1Y(X)-11*d1Y(X)+30*Y(X) = X^2' [ENTER]
      'Y(X)=(450*X^2+330*X+241)/13500' [ENTER]
                  SUBST  EVAL
```

Geben Sie dem Taschenrechner 10 Sekunden zur Errechnung des Ergebnisses: 'X^2 = X^2'.

<u>Beispiel 3</u> – Ein System linearer Differentialgleichungen mit konstanten Koeffizienten lösen.
Nehmen wir das System linearer Differentialgleichungen an als:

$$x_1'(t) + 2x_2'(t) = 0,$$
$$2x_1'(t) + x_2'(t) = 0.$$

In algebraischer Form geschrieben als: $\mathbf{A} \cdot \mathbf{x}'(t) = 0$, wobei $\mathbf{A} = \begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix}$. Das System kann mithilfe der Funktion LDEC mit den Argumenten [0,0] und Matrix A, wie im folgenden Bildschirm im ALG-Modus gelöst werden:



Die Lösung wird als Vektor mit den Funktionen [$x_1(t)$, $x_2(t)$] angezeigt. Durch Drücken von ▼ wird der Matrix-Schreiber ausgelöst und ermöglicht dem Anwender die Komponenten des Vektors anzusehen. Um alle Details jeder einzelnen Komponente zu sehen, drücken Sie die Softmenü-Taste EDIT. Vergewissern Sie sich, dass die Komponenten wie folgt lauten:

Funktion DESOLVE

Der Taschenrechner kann mithilfe der Funktion DESOLVE (Löser für Differentialgleichungen) bestimmte Arten von Differentialgleichungen lösen. Die Funktion benötigt die Eingabe der Differentialgleichung und der unbekannten Funktion und erzeugt gegebenenfalls die Lösung zur Gleichung. Sie können auch einen Vektor mit der Differentialgleichung und den Anfangsbedingungen eingeben, anstatt nur eine Differentialgleichung als Eingabe für DESOLVE zur Verfügung zu stellen. Die Funktion DESOLVE ist über das Menü CALC/DIFF verfügbar. Beispiele für DESOLVE-Anwendungen werden unten im RPN-Modus angeführt.

Beispiel 1 – Lösen Sie die ODE erster Ordnung:

$$dy/dx + x^2 \cdot y(x) = 5.$$

Verwenden Sie im Taschenrechner:

```
'd1y(x)+x^2*y(x)=5' ENTER 'y(x)' ENTER DESOLVE
```

Die Lösung lautet {'y = (INT(5*EXP(xt^3/3),xt,x)+cC0)*1/EXP(x^3/3))' }, d.h.

$$y(x) = \exp(-x^3/3) \cdot \left(\int 5 \cdot \exp(x^3/3) \cdot dx + cC_0 \right)$$

Die Variable ODETYPE

Auf den Kennzeichnungen für die Softmenü-Taste werden Sie eine neue Variable ODETY (ODETYPE) erkennen. Diese Variable wird unter Verwendung der Funktion DESOL erzeugt und enthält einen String, der die Art der verwendeten ODE als Eingabe für DESOLVE zeigt. Drücken Sie ODETY, um den String "`1st order linear`" zu erhalten.

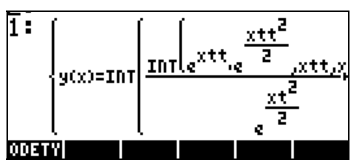
Beispiel 2 – Lösen Sie die ODE zweiter Ordnung:

$$d^2y/dx^2 + x\,(dy/dx) = \exp(x).$$

Verwenden Sie im Taschenrechner:

`'d1d1y(x)+x*d1y(x) = EXP(x)'` ENTER `'y(x)'` ENTER `DESOLVE`

Das Ergebnis ist ein Ausdruck, welcher zwei implizierte Integrationen enthält, nämlich



Bei dieser speziellen Gleichung erkennen wir jedoch, dass der linke Teil der Gleichung d/dx(x dy/dx) lautet, somit würde die ODE nun wie folgt geschrieben:

$$d/dx(x\ dy/dx\ ) = \exp x,$$

und

$$x\ dy/dx = \exp x + C.$$

Dann können wir schreiben:

$$dy/dx = (C + \exp x)/x = C/x + e^x/x.$$

Sie können versuchen, Folgendes in den Taschenrechner zu integrieren:

‘d1y(x) = (C + EXP(x))/x’ `ENTER` ‘y(x)’ `ENTER` DESOLVE

Das Ergebnis lautet: { ‘y(x) = INT((EXP(xt)+C)/xt,xt,x)+C0’ }, d.h..,

$$y(x) = \int \cdot \frac{e^x + C}{x} dx + C_0$$

Wenn wir versuchen die Integration manuell durchzuführen, schaffen wir das nur bis dahin:

$$y(x) = \int \cdot \frac{e^x}{x} dx + C \cdot \ln x + C_0$$

weil das Integral von exp(x)/x in geschlossener Form nicht vorhanden ist.

Beispiel 3 – Eine Lösung mit Anfangsbedingungen lösen. Lösen Sie

$d^2y/dt^2 + 5y = 2 \cos(t/2)$,

mit den Anfangsbedingungen

y(0) = 1,2, y’(0) = -0,5.

Verwenden Sie im Taschenrechner:

[‘d1d1y(t)+5*y(t) = 2*COS(t/2)’ ‘y(0) = 6/5’ ‘d1y(0) = -1/2’] `ENTER`
‘y(t)’ `ENTER`
DESOLVE

Achten Sie darauf, dass die Anfangsbedingungen im Hinblick auf ihre exakten Ausdrücke ‘y(0) = 6/5’, eher als ‘y(0)=1.2’, und ‘d1y(0) = -1/2’, eher als, ‘d1y(0) = -0.5’ verändert wurden. Durch Veränderungen nach diesen exakten Ausdrücken wird die Lösung vereinfacht.

Hinweis: Um Brüche für Dezimalwerte zu erhalten, verwenden Sie die Funktion →Q (siehe Kapitel 5).

Die Lösung hierfür lautet:

Drücken Sie EVAL EVAL, um das Ergebnis wie folgt zu vereinfachen:

‘y(t) = -((19*√5*SIN(√5*t)-(148*COS(√5*t)+80*COS(t/2)))/190)’.

Drücken Sie VAR ODETY ,um den String “`Linear w/ cst coeff`” für die ODE-Art in diesem Fall zu erhalten.

Laplace-Transformationen

Die Laplace-Transformation einer Funktion f(t) erzeugt eine Funktion F(s) in der Bilddomäne und kann verwendet werden, um die Lösung einer linearen Differentialgleichung mit f(t) über algebraische Methoden zu erhalten. In dieser Anwendung sind drei Schritte nötig:

1. Durch Verwendung der Laplace-Transformation wird die lineare ODE umgewandelt und fügt f(t) in eine algebraische Gleichung ein.
2. Die unbekannte F(s) wird nach einer Bilddomäne über algebraische Manipulation gelöst.
3. Eine inverse Laplace-Transformation wird verwendet, um die in Schritt 2 genannte Bildfunktion in die Lösung der Differentialgleichung f(t) umzuwandeln.

Definitionen

Die Laplace-Transformation für die Funktion f(t) ist die Funktion F(s), definiert als

$$L\{f(t)\} = F(s) = \int_0^\infty f(t)\cdot e^{-st}\,dt.$$

Die Bildvariable s kann eine komplexe Zahl sein und ist normalerweise auch eine.

Viele praktische Anwendungen der Laplace-Transformation enthalten eine ursprüngliche Funktion f(t), wobei t für Zeit steht, z. B. Kontrollsysteme in elektrischen oder hydraulischen Schaltkreisen. Normalerweise ist die Systemantwort nach der Zeit t>0 von Interesse, somit enthält die oben genannte Definition für die Laplace-Transformation eine Integration für Werte mit t größer als Null.
Die inverse Laplace-Transformation legt die Funktion F(s) auf die ursprüngliche Funktion f(t) in der Zeitdomäne aus, d.h. $L^{-1}\{F(s)\} = f(t)$.

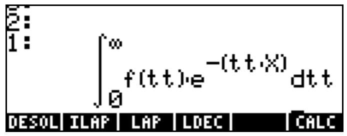
Das Konvolutionsintegral oder das Konvolutionsprodukt zweier Funktionen f(t) und g(t), wobei g in der Zeit verschoben ist, wird definiert als:

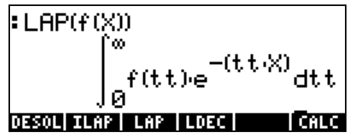
$$(f * g)(t) = \int_0^t f(u) \cdot g(t-u) \cdot du \, .$$

Laplace-Transformationen und Umkehrungen im Taschenrechner

Mithilfe der Funktionen LAP und ILAP können Laplace-Transformationen und inverse Laplace-Transformationen einer Funktion f(VX) gleichermaßen ausgeführt werden, wobei VX die vorgegebene unabhängige CAS Variable darstellt, die Sie auf 'X' setzen sollten.
Somit gibt der Taschenrechner die Transformation oder die inverse Transformation als Funktion von X wieder. Die Funktionen LAP und ILAP sind im Menü CALC/DIFF verfügbar. Die Beispiele sind im RPN-Modus angeführt, sie können jedoch problemlos in den ALG-Modus übertragen werden. Für diese Beispiele setzen Sie den CAS-Modus auf Real und Exact.

Beispiel 1 – Um die Definition der Laplace-Transformation zu erhalten, verwenden Sie Folgendes: 'f(X)' ENTER LAP im RPN-Modus oder LAP(F(X)) im ALG-Modus. Der Taschenrechner gibt das Ergebnis (RPN, links, ALG, rechts) wieder:





Vergleichen Sie die Ausdrücke mit der vorher in der Definition der Laplace-Transformation angegebenen, d.h.,

$$\mathsf{L}\{f(t)\} = F(s) = \int_0^\infty f(t) \cdot e^{-st}\, dt,$$

und Sie werden merken, dass die CAS Standardvariable X im Bildschirm des Gleichungsschreibers die Variable in dieser Definition ersetzt. Somit erhalten Sie durch Verwendung der Funktion LAP eine Funktion durch X, welche die Laplace-Transformation von f(X) ist.

Beispiel 2 – Bestimmen Sie die Laplace-Transformation von f(t) = $e^{2t}\cdot$sin(t). Verwenden Sie:
'EXP(2*X)*SIN(X)' `ENTER` LAP Der Taschenrechner gibt folgendes Ergebnis wieder:1/(SQ(X-2)+1). Drücken Sie `EVAL`, um 1/(X^2-4X+5) zu erhalten.

Wenn Sie dieses Ergebnis auf Papier übertragen würden, würden Sie Folgendes schreiben:

$$F(s) = \mathsf{L}\{e^{2t} \cdot \sin t\} = \frac{1}{s^2 - 4 \cdot s + 5}$$

Beispiel 3 – Bestimmen Sie die inverse Laplace-Transformation von F(t) = sin(s). Verwenden Sie:
'SIN(X)' `ENTER` ILAP. Geben Sie dem Taschenrechner einige Sekunden zur Errechnung des Ergebnisses: 'ILAP(SIN(X))', bedeutet, dass kein Ausdruck geschlossener Form f(t) vorliegt, sodass f(t) = L^{-1}\{sin(s)\}.

Beispiel 4 – Bestimmen Sie die inverse Laplace-Transformation von F(s) = 1/s^{3}. Verwenden Sie:
'1/X^3' `ENTER` ILAP `EVAL`. Der Taschenrechner kommt zu folgendem Ergebnis: 'X^2/2', was als $\mathsf{L}^{-1}\{1/s^3\} = t^2/2$ interpretiert wird.

Beispiel 5 – Bestimmen Sie die Laplace-Transformation der Funktion f(t) = cos (a·t+b). Verwenden Sie: ‘COS(a*X+b)’ ENTER LAP. Der Taschenrechner kommt zu folgendem Ergebnis:

$$\frac{X}{SQ(X)+SQ(a)}\cdot COS(b)-SIN(b)\cdot\frac{a}{SQ(X)+SQ(a)}$$

Drücken Sie EVAL, um $-(a\ \sin(b) - X\ \cos(b))/(X^2+a^2)$ zu erhalten. Die Transformation wird wie folgt interpretiert: $L\ \{\cos(a\cdot t+b)\} = (s\cdot\cos b - a\cdot\sin b)/(s^2+a^2)$.

Laplace-Transformations-Theoreme

Um die Bestimmung der Laplace-Transformation von Funktionen zu erleichtern, können Sie verschiedene Theoreme verwenden. Einige sind unten aufgelistet. Es sind auch einige Beispiele der Theorem-Anwendungen enthalten.

- Ableitungssatz für die erste Ableitung. Nehmen Sie f_o als Anfangsbedingung für f(t), d.h., $f(0) = f_o$, dann

$$L\{df/dt\} = s\cdot F(s) - f_o.$$

Beispiel 1 – Die Schnelligkeit eines sich bewegenden Partikels v(t) wird definiert als v(t) = dr/dt, wobei r = r(t) die Position des Partikels ist. Nehmen Sie $r_o = r(0)$, und R(s) =L{r(t)}, dann kann die Transformation der Schnelligkeit als $V(s) = L\{v(t)\}=L\{dr/dt\}= s\cdot R(s)-r_o$ geschrieben werden.

- Ableitungssatz für die zweite Ableitung. Nehmen Sie $f_o = f(0)$, und $(df/dt)_o = df/dt|_{t=0}$, dann ist $L\{d^2f/dt^2\} = s^2\cdot F(s) - s\cdot f_o - (df/dt)_o$.

Beispiel 2 – Als Folge von Beispiel 1 wird die Beschleunigung a(t) als a(t) = d^2r/dt^2 definiert. Wenn die Anfangsgeschwindigkeit $v_o = v(0) = dr/dt|_{t=0}$, ist, dann kann die Laplace-Transformation der Beschleunigung wie folgt geschrieben werden:

$$A(s) = L\{a(t)\} = L\{d^2r/dt^2\}= s^2\cdot R(s) - s\cdot r_o - v_o.$$

- Ableitungssatz für die n-te Ableitung. Nehmen Sie $f^{(k)}_o = d^kf/dx^k|_{t=0}$, und $f_o = f(0)$, dann

$$L\{d^nf/dt^n\} = s^n \cdot F(s) - s^{n-1} \cdot f_o - \ldots - s \cdot f^{(n-2)}_o - f^{(n-1)}_o.$$

- Linearitätssatz. $L\{af(t)+bg(t)\} = a \cdot L\{f(t)\} + b \cdot L\{g(t)\}$.

- Ableitungssatz für die Bildfunktion. Angenommen $F(s) = L\{f(t)\}$, dann ist $d^nF/ds^n = L\{(-t)^n \cdot f(t)\}$.

Beispiel 3 – Nehmen Sie $f(t) = e^{-at}$, unter Verwendung des Taschenrechners mit 'EXP(-a*X)' ENTER LAP, und Sie erhalten '1/(X+a)', oder F(s) = 1/(s+a). Die dritte Ableitung dieses Ausdrucks kann wie folgt berechnet werden:

'X' ENTER ↱ ∂ 'X' ENTER ↱ ∂ 'X' ENTER ↱ ∂ EVAL

Das Ergebnis lautet

'-6/(X^4+4*a*X^3+6*a^2*X^2+4*a^3*X+a^4)', oder

$$d^3F/ds^3 = -6/(s^4+4 \cdot a \cdot s^3+6 \cdot a^2 \cdot s^2+4 \cdot a^3 \cdot s+a^4).$$

Verwenden Sie nun '(-X)^3*EXP(-a*X)' ENTER LAP EVAL. Das Ergebnis ist genau das gleiche.

- Integrationssatz Nehmen Sie $F(s) = L\{f(t)\}$, dann

$$L\left\{\int_0^t f(u)du\right\} = \frac{1}{s} \cdot F(s).$$

- Konvolutionssatz Nehmen Sie $F(s) = L\{f(t)\}$, und $G(s) = L\{g(t)\}$, dann

$$L\left\{\int_0^t f(u)g(t-u)du\right\} = L\{(f*g)(t)\} =$$

$$L\{f(t)\} \cdot L\{g(t)\} = F(s) \cdot G(s)$$

Beispiel 4 – Verwenden Sie das Konvolutionstheorem und berechnen Sie die Laplace-Transformation von (f*g)(t), if f(t) = sin(t), und g(t) = exp(t). Zur Berechnung von F(s) = L{f(t)}, und G(s) = L{g(t)}, verwenden Sie: 'SIN(X)' ENTER LAP EVAL. Ergebnis, '1/(X^2+1)', d.h., $F(s) = 1/(s^2+1)$.
Und, 'EXP(X)' ENTER LAP. Ergebnis, '1/(X-1)', d.h. G(s) = 1/(s-1). Somit ist $L\{(f*g)(t)\} = F(s)\cdot G(s) = 1/(s^2+1)\cdot 1/(s-1) = 1/((s-1)(s^2+1)) = 1/(s^3-s^2+s-1)$.

- Verschiebungssatz für eine Verschiebung nach rechts. Nehmen Sie F(s) = L{f(t)}, dann

$$L\{f(t-a)\}=e^{-as}\cdot L\{f(t)\} = e^{-as}\cdot F(s).$$

- Verschiebungssatz für eine Verschiebung nach links. Nehmen Sie F(s) = L{f(t)} und a >0, dann

$$L\{f(t+a)\} = e^{as}\cdot\left(F(s)-\int_0^a f(t)\cdot e^{-st}\cdot dt\right).$$

- Ähnlichkeitssatz Nehmen Sie F(s) = L{f(t)}, und a>0, dann ist L{f(a·t)} = (1/a)·F(s/a).
- Dämpfungssatz Nehmen Sie F(s) = L{f(t)}, dann ist $L\{e^{-bt}\cdot f(t)\} = F(s+b)$.
- Divisionssatz Nehmen Sie F(s) = L{f(t)}, dann

$$L\left\{\frac{f(t)}{t}\right\} = \int_s^\infty F(u)du.$$

- Laplace-Transformation einer periodischen Funktion der Periode T:

$$L\{f(t)\} = \frac{1}{1-e^{-sT}}\cdot\int_0^T f(t)\cdot e^{-st}\cdot dt.$$

- Grenzwerttheorem für den Anfangswert: Nehmen Sie F(s) = L{f(t)}, dann

$$f_0 = \lim_{t\to 0} f(t) = \lim_{s\to\infty}[s \cdot F(s)].$$

- Grenzwerttheorem für den endgültigen Wert: Nehmen Sie F(s) = L{f(t)}, dann

$$f_\infty = \lim_{t\to\infty} f(t) = \lim_{s\to 0}[s \cdot F(s)].$$

Dirac'sche Deltafunktion und Heavisides Schrittfunktion

Bei der Analyse von Kontrollsystemen ist es üblich eine Funktionsart zu verwenden, die bestimmte physikalische Vorfälle, wie die plötzliche Aktivierung eines Schalters (Heavisides Schrittfunktion, H(t)) oder ein plötzlicher, unmittelbarer Höhepunkt bei einer Eingabe in ein System (Dirac'sche Deltafunktion δ(t)) darstellen. Sie sind Teil einer Klasse von Funktionen, die als generalisierte oder symbolische Funktionen bekannt sind [als Beispiel siehe Friedman, B., 1956, Principles and Techniques of Applied Mathematics, Dover Publications Inc., New York (1990 Nachdruck)].

Die formale Definition der Dirac'schen Deltafunktion δ(x) ist δ(x) = 0, für x ≠0, und

$$\int_{-\infty}^{\infty} \delta(x)dx = 1.0.$$

auch wenn f(x) eine kontinuierliche Funktion ist, dann ist

$$\int_{-\infty}^{\infty} f(x)\delta(x-x_0)dx = f(x_0).$$

Eine Interpretation für das oben genannte Integral laut Friedman (1990) ist, dass die Funktion δ den Wert der Funktion f(x) bei x = x_0. „heraushebt". Die Dirac'sche Deltafunktion ist typischerweise von einem Aufwärtspfeil beim Punkt x = x0 gekennzeichnet und zeigt an, dass die Funktion ein Nicht-Null Wert nur beim speziellen Wert von x_0 ist.

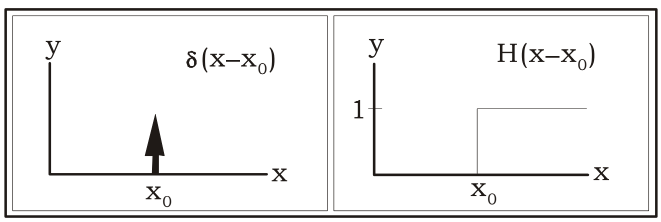
Heavisides Schrittfunktion, H(x), wird definiert als

$$H(x)=\begin{cases}1, & x>0\\ 0, & x<0\end{cases}$$

Auch für die kontinuierliche Funktion f(x) sind die

$$\int_{-\infty}^{\infty} f(x)H(x-x_0)dx = \int_{x_0}^{\infty} f(x)dx.$$

Die Dirac'sche Deltafunktion und Heavisides Schrittfunktion sind durch dH/dx = δ(x) verbunden. Die zwei Funktionen sind in der nachfolgenden Abbildung angeführt.



Sie können beweisen, dass L{H(t)} = 1/s,
woraus hervorgeht, dass L{U_o·H(t)} = U_o/s ist,
wobei U_o eine Konstante ist. Weiter gilt L^{-1}{1/s}=H(t),
und L^{-1}{ U_o /s}= U_o·H(t).
Auch bei Verwendung des Verschiebungssatzes für eine Verschiebung nach rechts L{f(t-a)}=e^{-as}·L{f(t)} = e^{-as}·F(s), können wir Folgendes schreiben: L{H(t-k)}=e^{-ks}·L{H(t)} = e^{-ks}·(1/s) = (1/s)·e^{-ks}.

Ein weiteres wichtiges Ergebnis, als <u>zweiter Verschiebungssatz für eine Verschiebung nach rechts</u> bekannt, ist, dass L^{-1}{e^{-as} ·F(s)}=f(t-a)·H(t-a), mit F(s) = L{f(t)}.

Im Taschenrechner wird die Heaviside Schrittfunktion H(t) einfach als '1' bezeichnet. Um die Transformation im Taschenrechner zu überprüfen, verwenden Sie: 1 ENTER LAP. Das Ergebnis lautet '1/X', d.h., L{1} = 1/s.
Ähnlich führt 'U0' ENTER LAP , zum Ergebnis 'U0/X', d.h., L{U_0} = U_0/s.

Sie erhalten die Dirac'sche Deltafunktion im Taschenrechner durch Verwendung von: 1 ENTER ILAP
Das Ergebnis lautet '`Delta(X)`'.

Das Ergebnis ist rein symbolisch, z. B. können Sie keinen numerischen Wert für '`Delta(5)`' finden.

Dieses Ergebnis kann als Laplace-Transformation der Dirac'schen Deltafunktion bezeichnet werden, weil aus $L^{-1}\{1.0\}= \delta(t)$, folgt, dass $L\{\delta(t)\} = 1.0$

Auch bei Verwendung des Verschiebungssatzes für eine Verschiebung nach rechts $L\{f(t-a)\}=e^{-as}\cdot L\{f(t)\} = e^{-as}\cdot F(s)$, können wir $L\{\delta(t-k)\}=e^{-ks}\cdot L\{\delta(t)\} = e^{-ks}\cdot 1.0 = e^{-ks}$ schreiben.

Anwendungen der Laplace-Transformation bei der Lösung linearer ODE

Zu Beginn des Abschnitts über die Laplace-Transformation haben wir darauf hingewiesen, dass Sie diese Transformationen verwenden können, um eine lineare ODE in der Zeitdomäne in eine algebraische Gleichung in der Bilddomäne umzuwandeln. Die resultierende Gleichung wird dann nach einer Funktion F(s) mittels algebraischen Methoden gelöst und die Lösung der ODE wird durch Verwendung der inversen Laplace-Transformation auf F(s) gefunden.

Die Theoreme auf Ableitungen einer Funktion, d.h.

$$L\{df/dt\} = s\cdot F(s) - f_o,$$

$$L\{d^2f/dt^2\} = s^2\cdot F(s) - s\cdot f_o - (df/dt)_o,$$

und im Allgemeinen

$$L\{d^nf/dt^n\} = s^n\cdot F(s) - s^{n-1}\cdot f_o - \ldots - s\cdot f^{(n-2)}_o - f^{(n-1)}_o,$$

sind besonders nützlich für die Transformation einer ODE in eine algebraische Gleichung.

Beispiel 1 – So lösen Sie die ODE erster Ordnung,

$$dh/dt + k\cdot h(t) = a\cdot e^{-t},$$

unter Verwendung der Laplace-Transformation können wir Folgendes schreiben:

$$L\{dh/dt + k\cdot h(t)\} = L\{a\cdot e^{-t}\},$$

$$L\{dh/dt\} + k\cdot L\{h(t)\} = a\cdot L\{e^{-t}\}.$$

Hinweis: 'EXP(-X)' [ENTER] LAP , ergibt '1/(X+1)', d.h., $L\{e^{-t}\}=1/(s+1)$.

Bei $H(s) = L\{h(t)\}$, und $L\{dh/dt\} = s\cdot H(s) - h_o$, wobei $h_o = h(0)$ ist, lautet die umgewandelte Gleichung $s\cdot H(s)-h_o+k\cdot H(s) = a/(s+1)$.

Verwenden Sie den Taschenrechner zum Lösen nach H(s) durch Eingabe von:

'X*H-h0+k*H=a/(X+1)' [ENTER] 'H' ISOL

Das Ergebnis ist 'H=((X+1)*h0+a)/(X^2+(k+1)*X+k)'.

Um die Lösung der ODE h(t) zu finden, müssen wir die inverse Laplace-Transformation wie folgt verwenden:

OBJ→ [◄] [◄]	Isoliert den rechten Teil des letzten Ausdrucks
ILAP [EVAL]	Ergibt die inverse Laplace-Transformation

Das Ergebnis lautet $\frac{a\cdot e^{k\cdot X}+((k-1)\cdot ho-a)\cdot e^{X}}{(k-1)\cdot e^{X}\cdot e^{k\cdot X}}$. X mit t in diesem Ausdruck ersetzen und Ergebnisse in $h(t) = a/(k-1)\cdot e^{-t} +((k-1)\cdot h_o-a)/(k-1)\cdot e^{-kt}$ vereinfachen.

Überprüfen Sie die Lösung der ODE, wenn Sie die Funktion LDEC verwenden würden.

'a*EXP(-X)' `ENTER` 'X+k' `ENTER` LDEC `EVAL`

Das Ergebnis lautet: $\frac{a\cdot e^{k\cdot X}+((k-1)\cdot cC0-a)\cdot e^{X}}{(k-1)\cdot e^{X}\cdot e^{k\cdot X}}$ d.h.

$$h(t) = a/(k\text{-}1)\cdot e^{-t} + ((k\text{-}1)\cdot cC_o\text{-}a)/(k\text{-}1)\cdot e^{-kt}.$$

Somit stellt cC0 im Ergebnis von LDEC die Anfangsbedingung h(0) dar.

Hinweis: Wenn Sie die Funktion LDEC zur Lösung einer linearer ODE der Ordnung n in f(X) verwenden, wird das Ergebnis als n-Konstanten cC0, cC1, cC2, …, cC(n-1) angezeigt und stellt die Anfangsbedingungen f(0), f'(0), f"(0), …, $f^{(n-1)}$ (0) dar.

<u>Beispiel 2</u> – Verwenden Sie die Laplace-Transformation zur Lösung der linearen Gleichung zweiter Ordnung

$$d^2y/dt^2+2y = \sin 3t.$$

Unter Verwendung der Laplace-Transformation können wir Folgendes schreiben:

$$L\{d^2y/dt^2+2y\} = L\{\sin 3t\},$$

$$L\{d^2y/dt^2\} + 2\cdot L\{y(t)\} = L\{\sin 3t\}.$$

Hinweis: 'SIN(3*X)' `ENTER` LAP `EVAL` ergibt '3/(X^2+9)', d.h.,
$L\{\sin 3t\}=3/(s^2+9)$.

Bei $Y(s) = L\{y(t)\}$, und $L\{d^2y/dt^2\} = s^2\cdot Y(s) - s\cdot y_o - y_1$, wobei y_o = h(0) ist und y_1 = h'(0), lautet die umgewandelte Gleichung:

$$s^2\cdot Y(s) - s\cdot y_o - y_1 + 2\cdot Y(s) = 3/(s^2+9).$$

Verwenden Sie den Taschenrechner zum Lösen nach Y(s) durch Schreiben von:

'X^2*Y-X*y0-y1+2*Y=3/(X^2+9)' `ENTER` 'Y' ISOL

Das Ergebnis lautet

‘Y=((X^2+9)*y1+(y0*X^3+9*y0*X+3))/(X^4+11*X^2+18)’.

Um die Lösung der ODE y(t) zu finden, müssen wir die inverse Laplace-Transformation wie folgt verwenden:

OBJ→ ◄ ◄	Isoliert den rechten Teil des letzten Ausdrucks
ILAP EVAL	Ergibt die inverse Laplace-Transformation

Das Ergebnis lautet

$$\frac{(7\cdot\sqrt{2}\cdot y1+3\cdot\sqrt{2})\cdot SIN(\sqrt{2}\cdot X)+14\cdot y0\cdot COS(\sqrt{2}\cdot X)-2\cdot SIN(3\cdot X)}{14}$$

d.h.

$$y(t) = -(1/7) \sin 3x + y_o \cos \sqrt{2}x + (\sqrt{2}\ (7y_1+3)/14) \sin \sqrt{2}x.$$

Überprüfen Sie die Lösung der ODE, wenn Sie die Funktion LDEC verwenden würden.

‘SIN(3*X)’ ENTER ‘X^2+2’ ENTER LDEC EVAL

Das Ergebnis lautet:

$$\frac{(7\cdot\sqrt{2}\cdot cC1+3\cdot\sqrt{2})\cdot SIN(\sqrt{2}\cdot X)+14\cdot cC0\cdot COS(\sqrt{2}\cdot X)-2\cdot SIN(3\cdot X)}{14}$$

d.h. das gleiche wie zuvor mit cC0 = y0 und cC1 = y1.

Hinweis: Wenn wir die zwei hier angeführten Beispiele verwenden, können wir vorher Aufgezeigtes bestätigen, d.h., dass die Funktion ILAP Laplace-Transformationen und inverse Transformationen zum Lösen linearer ODE verwendet, wenn der rechte Teil der Gleichung und die charakteristische Gleichung der entsprechenden homogenen ODE bekannt ist.

Beispiel 3 - Betrachten Sie die folgende Gleichung:

$$d^2y/dt^2+y = \delta(t\text{-}3),$$

wobei $\delta(t)$ eine Dirac'sche Deltafunktion ist.

Unter Verwendung der Laplace-Transformation können wir Folgendes schreiben:

$$L\{d^2y/dt^2+y\} = L\{\delta(t\text{-}3)\},$$

$$L\{d^2y/dt^2\} + L\{y(t)\} = L\{\delta(t\text{-}3)\}.$$

Bei 'Delta(X-3)' ENTER LAP errechnet der Taschenrechner EXP(-3*X), d.h., $L\{\delta(t\text{-}3)\} = e^{-3s}$. Bei $Y(s) = L\{y(t)\}$, und $L\{d^2y/dt^2\} = s^2\cdot Y(s) - s\cdot y_o - y_1$, wobei y_o = h(0) ist und y_1 = h'(0), lautet die umgewandelte Gleichung $s^2\cdot Y(s) - s\cdot y_o - y_1 + Y(s) = e^{-3s}$. Verwenden Sie den Taschenrechner zum Lösen nach Y(s) durch Eingeben von:

'X^2*Y-X*y0-y1+Y=EXP(-3*X)' ENTER 'Y' ISOL

Das Ergebnis lautet 'Y=(X*y0+(y1+EXP(-(3*X))))/(X^2+1)'.

Um die Lösung der ODE y(t) zu finden, müssen wir die inverse Laplace-Transformation wie folgt verwenden:

OBJ→ ◄ ◄	Isoliert den rechten Teil des letzten Ausdrucks
ILAP EVAL	Ergibt die inverse Laplace-Transformation

Das Ergebnis lautet 'y1*SIN(X)+y0*COS(X)+SIN(X-3)*Heaviside(X-3)'.

Hinweise:
[1]. Ein weiterer Weg, um die inverse Laplace-Transformation des Ausdrucks '(X*y0+(y1+EXP(-(3*X))))/(X^2+1)' zu erhalten, ist die Partialbruchzerlegung des Ausdrucks, d.h.

'y0*X/(X^2+1) + y1/(X^2+1) + EXP(-3*X)/(X^2+1)', und die Verwendung des Linearitätssatzes der inversen Laplace-Transformation

$$L^{-1}\{a\cdot F(s)+b\cdot G(s)\} = a\cdot L^{-1}\{F(s)\} + b\cdot L^{-1}\{G(s)\},$$

zum Schreiben von:

$$L^{-1}\{y_o\cdot s/(s^2+1)+y_1/(s^2+1)) + e^{-3s}/(s^2+1)) \} =$$

$$y_o\cdot L^{-1}\{s/(s^2+1)\}+ y_1\cdot L^{-1}\{1/(s^2+1)\}+ L^{-1}\{e^{-3s}/(s^2+1))\},$$

Somit können wir den Taschenrechner verwenden, um folgendes Ergebnis zu erhalten:

'X/(X^2+1)' `ENTER` ILAP Ergebnis, 'COS(X)', d.h., $L^{-1}\{s/(s^2+1)\}= \cos t$.
'1/(X^2+1)' `ENTER` ILAP Ergebnis, 'SIN(X)', d.h.., $L^{-1}\{1/(s^2+1)\}= \sin t$.
'EXP(-3*X)/(X^2+1)' `ENTER` ILAP Ergebnis, SIN(X-3)*Heaviside(X-3)'.

[2].Das letzte Ergebnis, d.h. die inverse Laplace-Transformation des Ausdrucks '(EXP(-3*X)/(X^2+1))', kann auch unter Verwendung des zweiten Verschiebungssatzes für eine Verschiebung nach rechts berechnet werden,

$$L^{-1}\{e^{-as}\cdot F(s)\}=f(t-a)\cdot H(t-a),$$

wenn wir eine inverse Laplace-Transformation für $1/(s^2+1)$ finden können. Versuchen Sie mit dem Taschenrechner '1/(X^2+1)' `ENTER` ILAP. Das Ergebnis lautet 'SIN(X)'. Somit ist $L^{-1}\{e^{-3s}/(s^2+1)\} = \sin(t-3)\cdot H(t-3)$,

Überprüfen Sie die Lösung der ODE, wenn Sie die Funktion LDEC verwenden würden.

'Delta(X-3)' `ENTER` 'X^2+1' `ENTER` LDEC `EVAL`

Das Ergebnis lautet:

'SIN(X-3)*Heaviside(X-3) + cC1*SIN(X) + cC0*COS(X)+'.

Bitte beachten Sie, dass die Variable X in diesem Ausdruck tatsächlich die Variable t in der ursprünglichen ODE ist. Somit kann die Lösung auf Papier wie folgt geschrieben werden:

$$y(t) = Co \cdot \cos t + C_1 \cdot \sin t + \sin(t-3) \cdot H(t-3)$$

Wenn wir dieses Ergebnis mit dem vorherigen Ergebnis für y(t) vergleichen, schließen wir daraus, dass $cC_o = y_o$, $cC_1 = y_1$ ist.

Definition und Anwendung der Schrittfunktion von Heaviside im Taschenrechner

Das vorhergehende Beispiel zeigte einige Erfahrungen in der Verwendung der Dirac'schen Deltafunktion als Eingabe in einem System auf (d.h. im rechten Teil der ODE bei der Beschreibung des Systems). In diesem Beispiel verwenden wir die Schrittfunktion von Heaviside H(t). Im Taschenrechner können wir diese Funktion wie folgt definieren:

'H(X) = IFTE(X>0, 1, 0)' ENTER ← DEF

Diese Definition führt zur Variable H auf der Softmenü-Taste im Taschenrechner.

Beispiel 1 – Um die Anzeige für H(t-2) zu erhalten, verwenden Sie z. B. eine Anzeige des Typs FUNCTION (siehe Kapitel 12):

- Drücken Sie ← 2D/3D , gleichzeitig im RPN-Modus, um ins Fenster PLOT SETUP zu gelangen.
  - Ändern Sie ggf. `TYPE` auf `FUNCTION`.
  - Ändern Sie EQ zu 'H(X-2)'.
  - Vergewissern Sie sich, dass `Indep` auf 'X' gesetzt ist.
  - Drücken Sie NXT OK, um zum normalen Taschenrechnerdisplay zurückzukehren.
- Drücken Sie ← WIN , gleichzeitig, um ins Fenster PLOT zu gelangen.
  - Ändern Sie den H-VIEW Bereich auf 0 bis 20, und den V-VIEW Bereich auf -2 bis 2.

➢ Drücken Sie ERASE DRAW zum Anzeigen der Funktion.

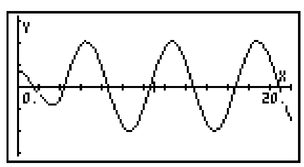
Die Funktion H(X) kann im Taschenrechner nicht mit LDEC, LAP, oder ILAP verwendet werden. Sie müssen die Hauptergebnisse verwenden, die Sie vorher mit der Schrittfunktion von Heaviside errechnet haben, verwenden, d.h. $L\{H(t)\} = 1/s$, $L^{-1}\{1/s\}=H(t)$, $L\{H(t-k)\}=e^{-ks}\cdot L\{H(t)\} = e^{-ks}\cdot(1/s) = \cdot(1/s)\cdot e^{-ks}$ und $L^{-1}\{e^{-as}\cdot F(s)\}=f(t-a)\cdot H(t-a)$.

<u>Beispiel 2</u> – Die Funktion $H(t-t_o)$ hat den Effekt, dass Sie auf die Funktion f(t) bei $t = t_o$ umschaltet, wenn sie mit einer Funktion f(t), d.h., $H(t-t_o)f(t)$, multipliziert wird. Die Lösung, zu der wir in Beispiel 3 gekommen sind, war z.B. $y(t) = y_o \cos t + y_1 \sin t + \sin(t-3)\cdot H(t-3)$. Angenommen wir verwenden die Anfangsbedingungen $y_o = 0{,}5$, und $y_1 = -0{,}25$. Lassen Sie uns sehen, wie diese Funktion aussieht:

- Drücken Sie ← 2D/3D, gleichzeitig falls im RPN-Modus, um ins Fenster PLOT SETUP zu gelangen.

➢ Ändern Sie ggf. `TYPE` auf `FUNCTION`.
➢ Ändern Sie EQ auf '0,5*COS(X)-0,25*SIN(X)+SIN(X-3)*H(X-3)'.
➢ Vergewissern Sie sich, dass `Indep` auf 'X' gesetzt ist.
➢ Drücken Sie ERASE DRAW zum Anzeigen der Funktion.
➢ Drücken Sie EDIT NXT LABEL zum Anzeigen der Funktion.

Die resultierende Grafik wird wie folgt aussehen.



Beachten Sie, dass das Signal mit einer relativ kleinen Amplitude beginnt, bei t=3 jedoch plötzlich auf ein Schwingungssignal mit größeren Schwingungen umschaltet. Der Unterschied im Signalverlauf vor und nach t = 3 ist das Einsetzen der besonderen Lösung $y_p(t) = \sin(t-3)\cdot H(t-3)$. Der Signalverlauf vor t = 3 stellt den Beitrag der homogenen Lösung $y_h(t) = y_o \cos t + y_1 \sin t$ dar.

Die Lösung einer Gleichung mit einem Treibersignal nach der Schrittfunktion von Heaviside wird unten angeführt.

Beispiel 3 – Bestimmen Sie die Lösung der Gleichung $d^2y/dt^2+y = H(t-3)$, wobei H(t) eine Schrittfunktion von Heaviside ist. Unter Verwendung der Laplace-Transformation können wir Folgendes schreiben: $L\{d^2y/dt^2+y\} = L\{H(t-3)\}$, $L\{d^2y/dt^2\} + L\{y(t)\} = L\{H(t-3)\}$. Der letzte Term im Ausdruck lautet: $L\{H(t-3)\} = (1/s)\cdot e^{-3s}$. Bei $Y(s) = L\{y(t)\}$, und $L\{d^2y/dt^2\} = s^2\cdot Y(s) - s\cdot y_o - y_1$, wobei $y_o = h(0)$ ist und $y_1 = h'(0)$, lautet die umgewandelte Gleichung: $s^2\cdot Y(s) - s\cdot y_o - y_1 + Y(s) = (1/s)\cdot e^{-3s}$. Wechseln Sie ggf. den CAS-Modus auf Exact. Verwenden Sie den Taschenrechner zum Lösen nach Y(s) durch Eingabe von:

‘X^2*Y-X*y0-y1+Y=(1/X)*EXP(-3*X)’ ENTER ‘Y’ ISOL.

Das Ergebnis lautet ‘Y=(X^2*y0+X*y1+EXP(-3*X))/(X^3+X)’.

Um die Lösung der ODE y(t) zu finden, müssen wir die inverse Laplace-Transformation wie folgt verwenden:

OBJ→ ◄ ◄	Isoliert den rechten Teil des letzten Ausdrucks
ILAP	Ergibt die inverse Laplace-Transformation

Das Ergebnis lautet ‘y1*SIN(X-1)+y0*COS(X-1)-(COS(X-3)-1)*Heaviside(X-3)’.

Somit schreiben wir als Lösung: $y(t) = y_o \cos t + y_1 \sin t + H(t-3)\cdot(1+\sin(t-3))$.

Überprüfen Sie die Lösung der ODE, wenn Sie die Funktion LDEC verwenden würden.

‘H(X-3)’ ENTER [ENTER] ‘X^2+1’ ENTER LDEC

Das Ergebnis lautet:

$$\mathrm{SIN(X)}\cdot\int_0^{+\infty}\frac{\mathrm{IFTE(ttt-3>0,1,0)}}{e^{X\cdot ttt}}\,dttt + \mathrm{cC1}\cdot\mathrm{SIN(X)}+\mathrm{cC0}\cdot\mathrm{COS(X)}$$

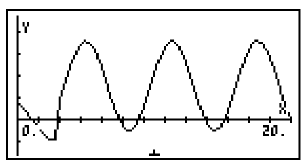
Bitte beachten Sie, dass die Variable X in diesem Ausdruck tatsächlich die Variable t in der ursprünglichen ODE darstellt, und dass die Variable *ttt* in diesem Ausdruck eine Hilfsvariable ist. Somit kann die Lösung auf Papier wie folgt geschrieben werden:

$$y(t) = Co \cdot \cos t + C_1 \cdot \sin t + \sin t \cdot \int_0^{\infty} H(u-3) \cdot e^{-ut} \cdot du.$$

<u>Beispiel 4</u> – Zeigen Sie die Lösung zu Beispiel 3 unter Verwendung der gleichen Werte für y_o und y_1 an, die in der Zeichnung in Beispiel 1 oben verwendet wurden. Wir zeichnen nun die Funktion:

y(t) = 0,5 cos t –0,25 sin t + (1+sin(t-3))·H(t-3).

Im Bereich 0 < t < 20 und bei Änderung des vertikalen Bereichs auf (-1,3), sollte die Grafik wie folgt aussehen.



Wiederum gibt es eine neue Komponente in der Bewegung, die bei t=3, d.h. die besondere Lösung y_p(t) = [1+sin(t-3)]·H(t-3), einsetzt und die Lösung für t>3 verändert.

Die Schrittfunktion von Heaviside kann mit einer konstanten Funktion und einer linearen Funktion kombiniert werden, um endliche Rechteckimpulse, Dreieckimpulse und Sägezahnimpulse wie folgt zu verändern:

- Rechteckimpuls der Größe U_o im Intervall a < t < b:

  f(t) = Uo[H(t-a)-H(t-b)].

- Dreieckimpuls mit einem Maximalwert von Uo, der ab a < t < b ansteigt und ab b < t < c fällt:

$$f(t) = U_o\cdot ((t-a)/(b-a)\cdot[H(t-a)-H(t-b)]+(1-(t-b)/(b-c))[H(t-b)-H(t-c)]).$$

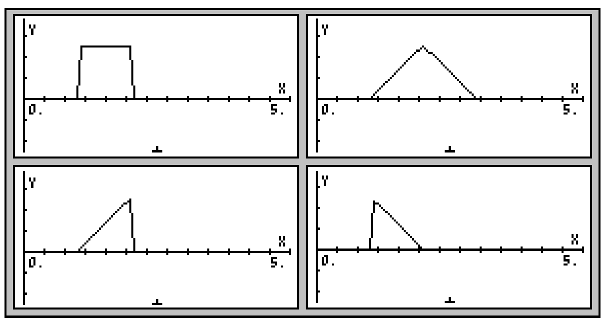
- Sägezahnimpuls steigt bis zu einem Maximalwert von Uo für a < t < b und sinkt bei b = t plötzlich auf Null ab:

$$f(t) = U_o\cdot (t-a)/(b-a)\cdot[H(t-a)-H(t-b)].$$

- Sägezahnimpuls steigt plötzlich bis zu einem Maximalwert von Uo bei t = a, und sinkt dann linear bis Null für a < t < b:

$$f(t) = U_o\cdot[1-(t-a)/(b-1)]\cdot[H(t-a)-H(t-b)].$$

Beispiele für die Veränderung der Zeichnungen durch diese Funktionen für Uo = 1, a = 2, b = 3, c = 4, horizontaler Bereich = (0,5), und vertikaler Bereich = (-1; 1,5), sehen Sie in den Abbildungen unten:



Fourier-Reihen

Fourier-Reihen sind Reihen, welche die Sinus- und Kosinusfunktionen einbeziehen, die typischerweise zur Entwicklung periodischer Funktionen verwendet werden. Eine Funktion f(x) wird als *periodisch* mit Periode T bezeichnet, wenn f(x+T) = f(t). Beispiel: Weil sin(x+2π) = sin x, und cos(x+2π) = cos x, sind die Funktionen *sin* und *cos* 2π-periodische Funktionen. Wenn zwei Funktionen f(x) und g(x) periodisch mit der Periode T sind, dann ist auch ihre lineare Kombination h(x) = a·f(x) + b·g(x), periodisch mit Periode T. Eine T-periodische Funktion f(t) kann in eine Reihe von Sinus- und Kosinus-

Funktionen, bekannt als Fourier-Reihe, entwickelt werden, die gegeben ist durch:

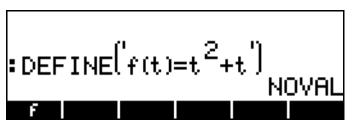
$$f(t) = a_0 + \sum_{n=1}^{\infty}\left(a_n \cdot \cos\frac{2n\pi}{T}t + b_n \cdot \sin\frac{2n\pi}{T}t\right)$$

wobei die Koeffizienten a_n und b_n gegeben sind durch

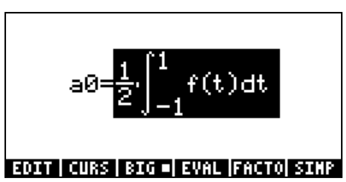
$$a_0 = \frac{1}{T}\int_{-T/2}^{T/2} f(t)\cdot dt, \quad a_n = \frac{2}{T}\int_{-T/2}^{T/2} f(t)\cdot\cos\frac{2n\pi}{T}t\cdot dt,$$

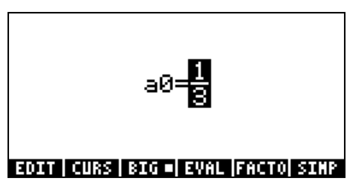
$$b_n = \int_{-T/2}^{T/2} f(t)\cdot\sin\frac{2n\pi}{T}t\cdot dt.$$

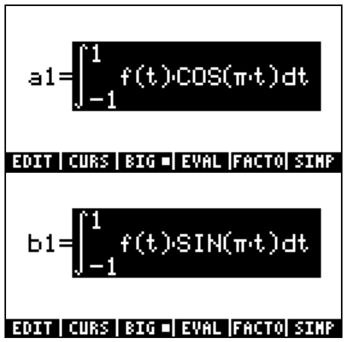
Die folgenden Übungen werden im ALG-Modus durchgeführt, wobei der CAS-Modus auf Exact gesetzt ist. (Wenn Sie eine Grafik produzieren, wird der CAS-Modus auf Approx. zurückgestzt. Vergessen Sie nicht, ihn auf Exact zurückzusetzen, nachdem die Grafik erzeugt wurde). Angenommen die Funktion $f(t) = t^2+t$ ist periodisch mit Periode $T = 2$. Um die Koeffizienten a_0, a_1, und b_1 für die entsprechende Fourier-Reihe zu bestimmen, müssen wir wie folgt vorgehen: Bestimmen Sie zuerst Funktion $f(t) = t^2+t$:

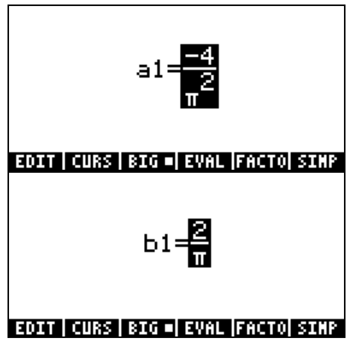


Nun berechnen wir die Koeffizienten mithilfe des Gleichungsschreibers:





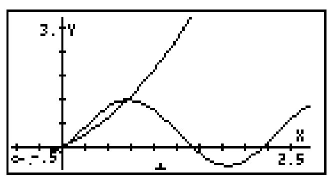




Somit sind die ersten drei Terme der Funktion:

$$f(t) \approx 1/3 - (4/\pi^2)\cdot\cos(\pi\cdot t) + (2/\pi)\cdot\sin(\pi\cdot t).$$

Ein grafischer Vergleich der ursprünglichen Funktion mit der Fourier-Ausweitung unter Verwendung der drei Terme zeigt, dass die Annäherung für t < 1, oder um diesen Bereich herum, akzeptabel ist. Wir hatten jedoch angenommen, dass T/2 = 1. Deshalb ist die Annäherung nur zwischen –1 < t < 1 gültig.



Funktion FOURIER

Ein weiterer Weg, um eine Fourier-Reihe zu bestimmen, ist die Verwendung von komlexen Zahlen wie folgt:

$$f(t) = \sum_{n=-\infty}^{+\infty} c_n \cdot \exp\left(\frac{2in\pi t}{T}\right),$$

wobei

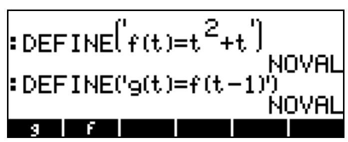
$$c_n = \frac{1}{T}\int_0^T f(t)\cdot \exp\left(-\frac{2\cdot i\cdot n\cdot \pi}{T}\cdot t\right)\cdot dt, \quad n = -\infty,\ldots,-2,-1,0,1,2,\ldots\infty.$$

Die Funktion FOURIER ergibt den Koeffizienten c_n der komplexen Form der Fourier-Reihe, wenn die Funktion f(t) und der Wert von n gegeben sind. Die Funktion FOURIER erfordert, dass Sie den Wert der Periode (T) einer T-periodischen Funktion in die CAS Variable PERIOD speichern, bevor Sie die Funktion aufrufen. Die Funktion FOURIER ist im Untermenü DERIV im CALC-Menü vefügbar (↰ CALC).

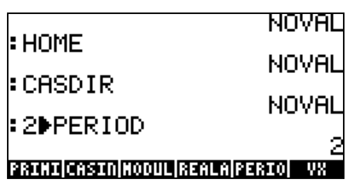
Fourier-Reihe für eine quadratische Funktion

Bestimmen Sie die Koeffizienten c_0, c_1, und c_2 für die Funktion $f(t) = t^2+t$, mit Periode T = 2. (Hinweis: Weil das von der Funktion FOURIER verwendete Integral im Intervall [0,T] berechnet wird, und das vorher definierte im Intervall [-T/2,T/2] berechnet wurde, müssen wir die Funktion in der T-Achse verschieben, indem wir T/2 von t subtrahieren, d.h. wir verwenden $g(t) = f(t-1) = (t-1)^2+(t-1)$.)

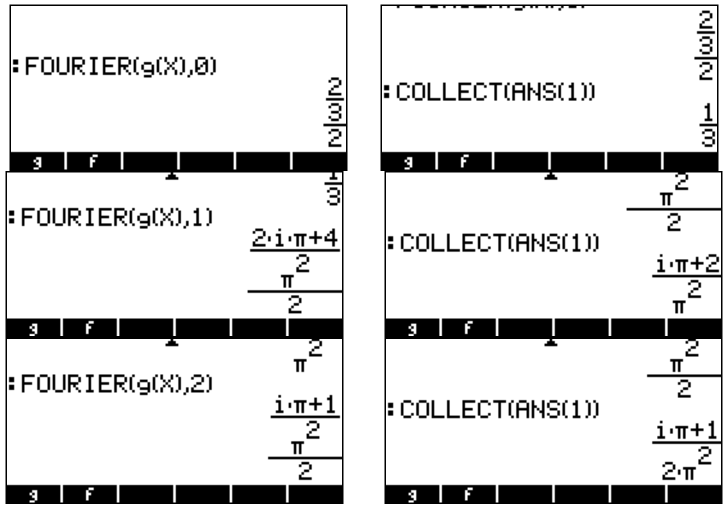
Mithilfe des Taschenrechners im ALG-Modus definieren wir erst die Funktionen f(t) und g(t).



Als nächstes verschieben wir das CASDIR-Unterverzeichnis nach HOME, um den Wert der Variable PERIOD zu verändern, z. B. ↰ (gedrückt halten) UPDIR ENTER VAR CASDI ENTER 2 STO▶ PERIO ENTER



Kehren Sie zum Unterverzeichnis zurück, in dem Sie die Funktionen f und g definiert haben, und berechnen Sie die Koeffizienten (Accept ggf. auf Complex-Modus umschalten):

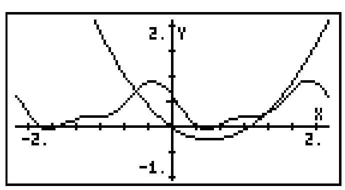


Somit ist $c_0 = 1/3$, $c_1 = (\pi \cdot i+2)/\pi^2$, $c_2 = (\pi \cdot i+1)/(2\pi^2)$.

Die Fourier-Reihe mit drei Elementen wird wie folgt geschrieben:

$$g(t) \approx \mathrm{Re}[(1/3) + (\pi \cdot i+2)/\pi^2 \cdot \exp(i \cdot \pi \cdot t)+ (\pi \cdot i+1)/(2\pi^2) \cdot \exp(2 \cdot i \cdot \pi \cdot t)].$$

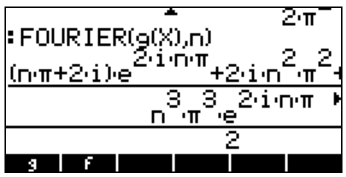
Eine Zeichnung der verschobenen Funktion g(t) und die Fourier-Reihen-Annäherung folgt:

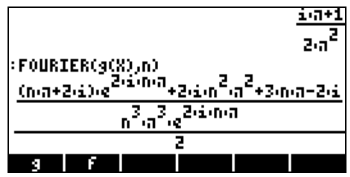


Die Annäherung ist einigermaßen annehmbar für 0<t<2, auch wenn sie nicht so gut wie im vorherigen Beispiel ist.

Ein allgemeiner Ausdruck für c_n

Die Funktion FOURIER kann einen allgemeinen Ausdruck für den Koeffizienten c_n der komplexen Fourier-Reihen-Entwicklung ergeben. Z. B. bei Verwendung der gleichen Funkton g(t) wie vorher, ist der allgemeine Term c_n gegeben durch (die Abbildungen zeigen normale und kleine Schriftarten an):



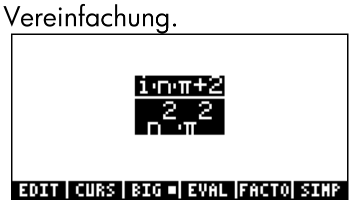


Nach Vereinfachung des vorherigen Ergebnisses lautet der allgemeine Ausdruck:

$$c_n = \frac{(n\pi + 2i)\cdot e^{2in\pi} + 2i^2 n^2 \pi^2 + 3n\pi - 2i}{2n^3\pi^3 \cdot e^{2in\pi}}$$

Wir können diesen Ausdruck noch weiter vereinfachen, wenn wir die Formel für komplexe Zahlen von Euler verwenden, d.h. $e^{2in\pi} = \cos(2n\pi) + i\cdot\sin(2n\pi) = 1 + i\cdot 0 = 1$, da $\cos(2n\pi) = 1$, und $\sin(2n\pi) = 0$, für n ganzzahlig.

Mit dem Taschenrechner können Sie den Ausdruck im Gleichungsschreiber vereinfachen (↱ EQW), indem Sie $e^{2in\pi}$ = 1ersetzen. Die Abbildung zeigt den Ausdruck nach der Vereinfachung.

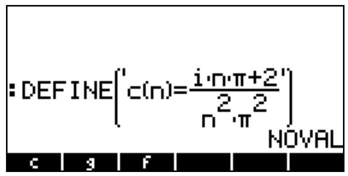


Das Ergebnis lautet $c_n = (i\cdot n\cdot\pi+2)/(n^2\cdot\pi^2)$.

Die komplexe Fourier-Reihe zusammenfügen

Nachdem wir den allgemeinen Ausdruck für c_n, festgelegt haben, können wir eine endliche komplexe Fourier-Reihe zusammenfügen, indem wir die Summenfunktion (Σ) im Taschenrechner wie folgt verwenden:

- Bestimmen Sie zuerst eine Funktion c(n), die den allgemeinen Term c_n in der komplexen Fourier-Reihe darstellt.



- Definieren Sie nun die endliche komplexe Fourier-Reihe F(X,k), wobei X die unabhängige Variable ist und k die Anzahl der zu verwendenden Terme bestimmt. Idealerweise würden wir diese endliche komplexe Fourier-Reihe wie folgt schreiben:

$$F(X,k) = \sum_{n=-k}^{k} c(n) \cdot \exp(\frac{2 \cdot i \cdot \pi \cdot n}{T} \cdot X)$$

  Weil jedoch die Funktion c(n) für n = 0 nicht definiert ist, sollten wir den Ausdruck wie folgt schreiben:

$$F(X,k,c0) = c0 +$$

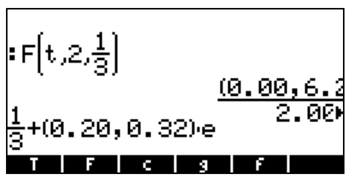
$$\sum_{n=1}^{k} [c(n) \cdot \exp(\frac{2 \cdot i \cdot \pi \cdot n}{T} \cdot X) + c(-n) \cdot \exp(-\frac{2 \cdot i \cdot \pi \cdot n}{T} \cdot X)],$$

Oder in der Eingabezeile des Taschenrechners als:

DEFINE('F(X,k,c0) = c0+Σ(n=1,k,c(n)*EXP(2*i*π*n*X/T)+
c(-n)*EXP(-(2*i*π*n*X/T))'),

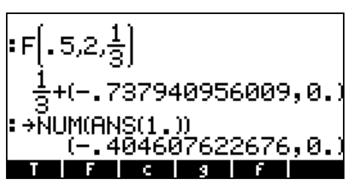
wobei T die Periode T = 2 darstellt. Die folgenden Screenshots zeigen die Definition der Funktion F und die Speicherung von T = 2.

Die Funktion kann verwendet werden, um den Ausdruck für komplexe Fourier-Reihen für einen endlichen Wert von k zu verändern. Z. B. für k = 2, c_0 = 1/3 und unter Verwendung von t als unabhängige Variable können wir F(t,2,1/3) auswerten, um Folgendes zu erhalten:



Das Ergebnis zeigt nur den ersten Term (c0) und einen Teil des ersten Exponentialterms in der Reihe. Das Dezimalformat wechselte auf Fix mit 2 Dezimalstellen, um einige Koeffizienten in der Entwicklung und im Exponenten anzuzeigen. Wie erwartet sind die Koeffizienten komplexe Zahlen.

Die Funktion F, somit definiert, genügt, um Werte der endlichen Fourier-Reihe zu erhalten. Ein Einzelwert der Reihe F(0,5;2,1/3) kann z. B. erhalten werden, wenn man Folgendes verwendet (CAS-Modus auf Exact gesetzt, step-by-step, und Complex):

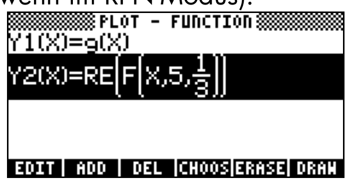


Die Umschaltung auf den `Approx`-Modus ggf. akzeptieren. Das Ergebnis ist der Wert –0,40467…. Der tatsächliche Wert der Funktion g(0,5) ist g(0,5) =

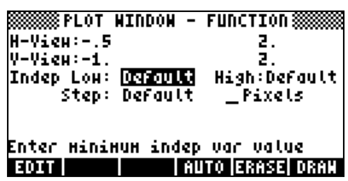
-0,25. Die folgenden Berechnungen zeigen wie sehr sich die Fourier-Reihe an diesen Wert nähert, wenn die Anzahl der Komponenten in der Reihe, gegeben durch k, zunimmt:

F (0,5; 1; 1/3) = (-0,303286439037,0.)
F (0,5; 2; 1/3) = (-0,404607622676,0.)
F (0,5; 3; 1/3) = (-0,192401031886,0.)
F (0,5; 4; 1/3) = (-0,167070735979,0.)
F (0,5; 5; 1/3) = (-0,294394690453,0.)
F (0,5; 6; 1/3) = (-0,305652599743,0.)

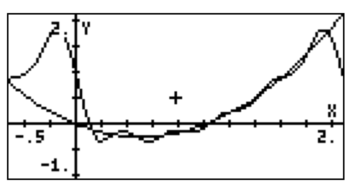
Um die Ergebnisse der Reihe mit denen der ursprünglichen Funktion zu vergleichen, laden Sie diese Funktionen in die Eingabeform PLOT-FUNCTION (Y= , gleichzeitig, wenn im RPN-Modus):



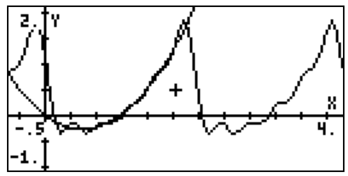
Die Grenzen des Zeichnungsfensters (WIN) wie folgt verändern:



Drücken sie die Softmenü-Tasten ERASE DRAW, um die Zeichnung zu erstellen:



Beachten Sie, dass die Reihe mit 5 Termen die Grafik mit den Funktionen sehr eng im Intervall 0 bis 2 umrandet (z. B. durch die Periode T = 2). Es ist auch eine Periodizität in der Grafik der Reihen zu erkennen. Diese Periodizität kann leicht veranschaulicht werden, indem der horizontale Bereich der Zeichnung auf (-0,5;4) ausgedehnt wird:



Fourier-Reihe für eine Dreieckschwingung

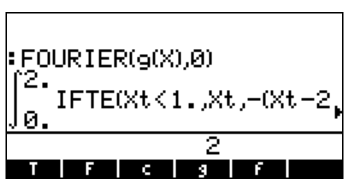
Beachten Sie die Funktion

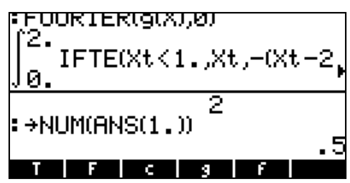
$$g(x) = \begin{cases} x, & if\ 0 < x < 1 \\ 2 - x, & if\ 1 < x < 2 \end{cases}$$

die wir als periodisch mit T = 2 annehmen. Diese Funktion kann im Taschenrechner im ALG-Modus durch folgenden Ausdruck definiert werden:

DEFINE('g(X) = IFTE(X<1,X,2-X)')

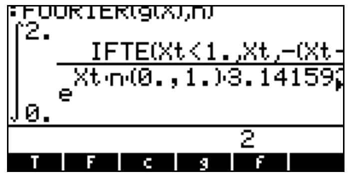
Wenn Sie dieses Beispiel nach Beenden von Beispiel 1 begonnen haben, haben Sie bereits einen Wert für 2 in der CAS Variable PERIOD gespeichert. Wenn Sie sich nicht sicher sind, überprüfen Sie den Wert dieser Variable und speichern Sie ggf. eine 2 darin. Der Koeffizient c_0 für die Fourier-Reihe wird wie folgt berechnet:





Der Taschenrechner wird Sie auffordern, in den Approx-Modus zu wechseln, um die im Integranden enthaltene Funktion IFTE() zu integrieren. Bei

Bestätigung des Wechsels zu Approx erscheint c_0 = 0,5. Wenn wir nun einen allgemeinen Ausdruck für den Koeffizienten c_n erhalten wollen, verwenden wir:

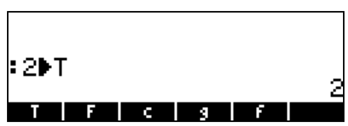


Der Taschenrechner ermittelt ein Integral, das numerisch nicht ausgewertet werden kann, weil es vom Parameter n abhängt. Der Koeffizient kann dennoch berechnet werden, indem seine Definition in den Taschenrechner eingegeben wird, d.h.

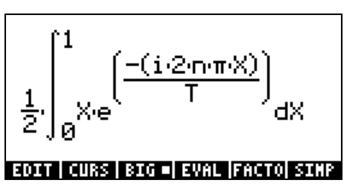
$$\frac{1}{2}\cdot\int_0^1 X\cdot EXP\left(-\frac{i\cdot 2\cdot n\cdot\pi\cdot X}{T}\right)\cdot dX +$$

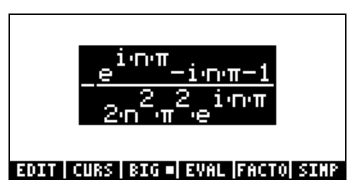
$$\frac{1}{2}\cdot\int_1^2 (2-X)\cdot EXP\left(-\frac{i\cdot 2\cdot n\cdot\pi\cdot X}{T}\right)\cdot dX$$

wobei T = 2 die Periode ist. Der Wert von T kann gespeichert werden mittels:

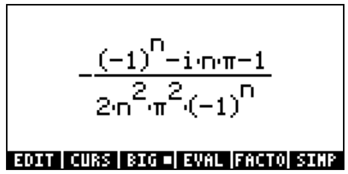


Durch Eingabe des ersten Integrals oben in den Gleichungsschreiber und Auswahl des gesamten Ausdrucks und der Verwendung von EVAL, erhalten wir Folgendes:

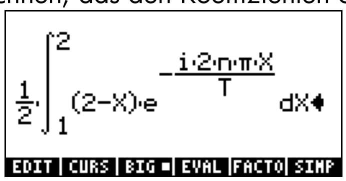


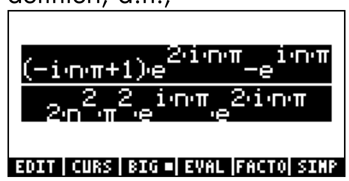


Rufen Sie $e^{in\pi} = \cos(n\pi) + i\cdot\sin(n\pi) = (-1)^n$ auf. Durch den Austauschvorgang im oben erhaltenen Ergebnis erhalten wir:

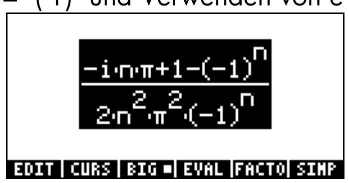


Drücken Sie ENTER ENTER, um dieses Ergebnis auf den Bildschirm zu kopieren. Aktivieren Sie dann erneut den Gleichungsschreiber, um das zweite Integral zu berechnen, das den Koeffizienten c_n, definiert, d.h.,

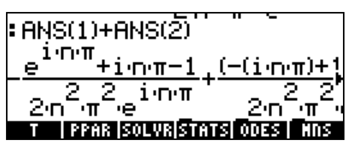




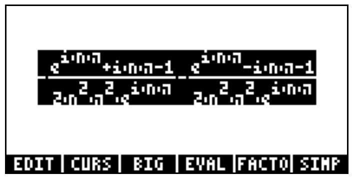
Durch Ersetzen von $e^{in\pi} = (-1)^n$ und Verwenden von $e^{2in\pi} = 1$ erhalten wir:

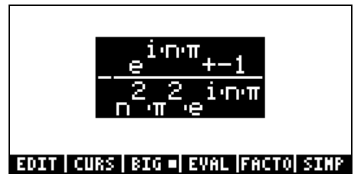


Drücken Sie ENTER ENTER, um dieses zweite Ergebnis auf den Bildschirm zu kopieren. Fügen Sie nun ANS(1) und ANS(2) hinzu, um den vollständigen Ausdruck für c_n zu erhalten:



Durch Drücken von ▼ wird dieses Ergebnis in den Gleichungsschreiber gelegt, wo wir es zum Lesen vereinfachen (SIMP) können:



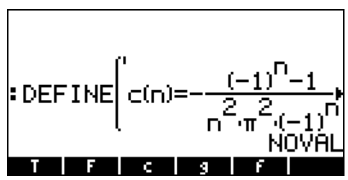


Durch Ersetzen von $e^{in\pi}$ = $(-1)^n$ erhalten wir:

Dieses Ergebnis wird verwendet, um die Funktion c(n) wie folgt zu definieren:

DEFINE('c(n) = - (((-1)^n-1)/(n^2*π^2*(-1)^n)')

d.h.

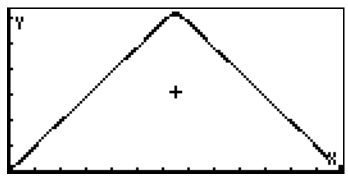


Als nächstes definieren wir die Funktion F(X,k,c0), um die Fourier-Reihe zu berechnen (wenn Sie Beispiel 1 abgeschlossen haben, ist diese Funktion bereits gespeichert):

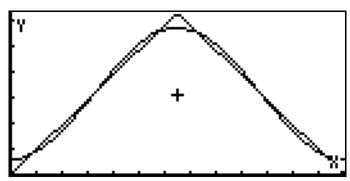
DEFINE('F(X,k,c0) = c0+Σ(n=1,k,c(n)*EXP(2*i*π*n*X/T)+
c(-n)*EXP(-(2*i*π*n*X/T))'),

Um die ursprüngliche Funktion und die Fourier-Reihe zu vergleichen, können wir eine Simultanzeichnung beider Funktionen erzeugen. Die Details sind ähnlich wie in Beispiel 1, außer dass wir hier einen horizontalen Bereich von 0 bis 2 und einen vertikalen Bereich von 0 bis 1 verwenden und die Gleichungen so anpassen, dass wir sie wie hier angeführt zeichnen können:

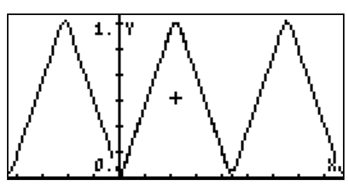
Die resultierende Grafik ist unten für k = 5 angeführt (die Anzahl der Elemente in der Reihe ist 2k+1, d.h. 11 in diesem Fall):



In der Zeichnung ist es schwierig, die ursprüngliche Funktion von der Fourier-Reihen-Annäherung zu unterscheiden. Die Verwendung von k= 2 oder 5 Termen in der Reihe führt nicht zu einer so guten Annäherung:



Die Fourier-Reihe kann verwendet werden, um eine periodische Dreieckschwingung (oder Sägezahnwelle) zu verändern, indem der horizontale Achsenbereich z. B. von -2 auf 4 verändert wird. Die unten angeführte Grafik verwendet k = 5:



Fourier-Reihe für eine Rechteckschwingung

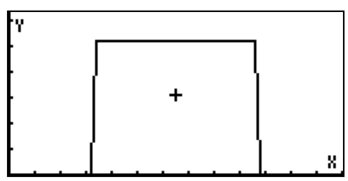
Eine Rechteckschwingung kann verändert werden unter Verwendung der Funktion:

$$g(x) = \begin{cases} 0, & if\ 0 < x < 1 \\ 1, & if\ 1 < x < 3 \\ 0, & if\ 3 < x < 4 \end{cases}$$

In diesem Fall ist die Periode T 4. Achten Sie darauf, dass Sie den Wert der Variable T auf 4 setzen (verwenden Sie: 4 STO▶ T ENTER). Funktion g(X) kann im Taschenrechner definiert werden durch

DEFINE('g(X) = IFTE((X>1) AND (X<3),1,0)')

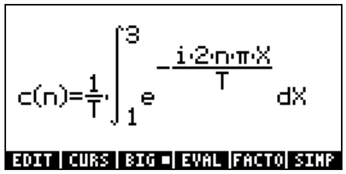
Die Funktion wird wie folgt gezeichnet (horizontaler Bereich: 0 bis 4, vertikaler Bereich:0 bis 1,2):

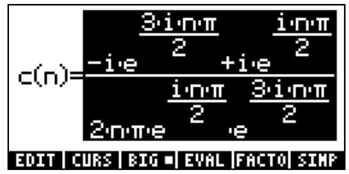


Bei Verwendung eines ähnlichen Vorganges wie dem der Dreiecksform in Beispiel 2 oben, werden Sie sehen, dass

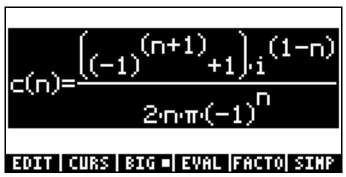
$$c_0 = \frac{1}{T} \cdot \left(\int_1^3 1 \cdot dX \right) = 0.5\,,$$

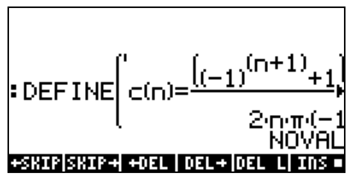
und



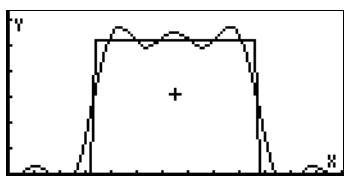


Wir können diesen Ausdruck vereinfachen, indem wir $e^{in\pi/2} = i^n$ und $e^{3in\pi/2} = (-i)^n$ verwenden, um Folgendes zu erhalten:

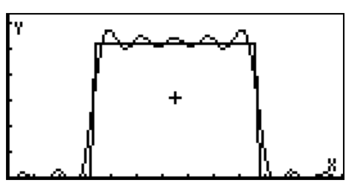




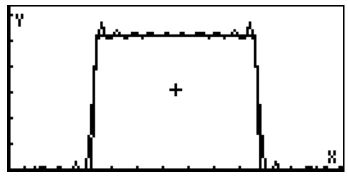
Die Vereinfachung des rechten Teils von c(n) oben ist einfacher auf Papier zu vollziehen, (d.h. manuell). Geben Sie dann den Ausdruck für c(n) erneut wie in der linken oberen Abbildung ein, um die Funktion c(n) zu definieren. Die Fourier-Reihe wird mit F(X,k,c0) berechnet, wie in den Beispielen 1 und 2 oben mit c0 = 0,5. Für k = 5, d.h. mit 11 Komponenten ist die Annäherung unten angeführt:



Eine bessere Annäherung kann durch Verwendung von k = 10 erreicht werden, d.h.



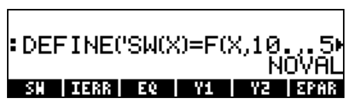
Bei k = 20 ist die Annäherung sogar noch besser, aber die Erstellung der Grafik dauert länger.



Fourier-Reihen-Anwendungen bei Differentialgleichungen

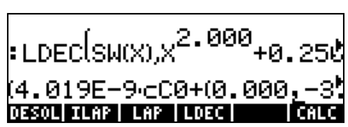
Angenommen wir wollen die periodische Rechteckschwingung aus dem vorherigen Beispiel als Anregung eines ungedämpften Feder-Masse-Systems berechnen, dessen homogene Gleichung wie folgt lautet: $d^2y/dX^2 + 0.25y = 0$.

Wir können die Anregungskraft ändern, indem wir die Annäherung mit k = 10 aus der Fourier-Reihe erhalten, unter Verwendung von SW(X) = F(X;10;0,5):

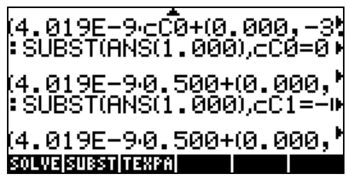


Wir können dieses Ergebnis als erste Eingabe für die Funktion LDEC verwenden, wenn es verwendet wird, um eine Lösung des Systems $d^2y/dX^2 + 0{,}25y = SW(X)$ zu erhalten, wobei SW(X) für die Rechteckschwingung von X steht. Gegenstand der zweiten Eingabe ist die charakteristische Gleichung, die der oben angeführten homogenen ODE entspricht, d.h. 'X^2+0,25'.

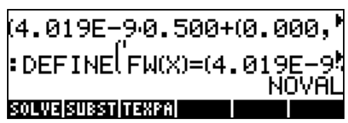
Mithilfe dieser zwei Eingaben kommt die Funktion LDEC zu folgendem Ergebnis (Dezimalformat wechselte zu Fix mit 3 Dezimalstellen).



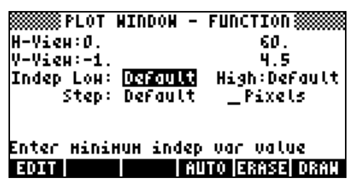
Drücken Sie zum Anzeigen des vollständigen Ausdrucks im Gleichungsschreiber. Die Untersuchung der Gleichung im Gleichungsschreiber bringt hervor, dass zwei Integrationskonstanten existieren, cC0 und cC1. Die Werte wurden mithilfe der Anfangsbedingungen berechnet. Angenommen wir verwenden die Werte cC0 = 0,5 und cC1 = -0,5, dann können wir diese Werte in der obigen Lösung mithilfe der Funktion SUBST (siehe Kapitel 5) ersetzen. Verwenden Sie in diesem Fall SUBST(ANS(1),cC0=0,5) ENTER, gefolgt von SUBST(ANS(1),cC1=-0,5) ENTER. Zurück in dem normalen Taschenrechnerdisplay können wir Folgendes verwenden:



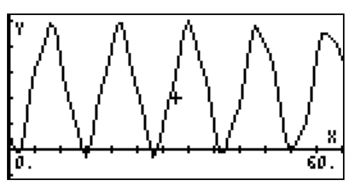
Das letztere Ergebnis kann als Funktion FW(X) wie folgt definiert werden (das letzte Ergebnis ausschneiden und in den Befehl einfügen):



Wir können nun den realen Teil dieser Funktion zeichnen. Wechseln Sie vom Dezimal-Modus auf Standard und verwenden Sie:



Die Lösung ist unten angeführt:



Fourier-Transformationen

Bevor wir auf das Konzept von Fourier-Transformationen eingehen, sprechen wir über die allgemeine Definition einer Integral-Transformation. Generell gesehen, ist eine Integral-Transformation eine Transformation, die eine Funktion f(t) mit einer neuen Funktion F(s) verbindet und zwar durch die Integration der Form $F(s) = \int_a^b \kappa(s,t) \cdot f(t) \cdot dt$. Die Funktion κ(s,t) wird als Kern der Transformation bezeichnet.

Mithilfe einer Integral-Transformation können wir eine Funktion in ein bestimmtes Spektrum von Komponenten zerlegen. Um das Konzept eines Spektrums zu verstehen, beachten Sie die Fourier-Reihe

$$f(t) = a_0 + \sum_{n=1}^{\infty} \left(a_n \cdot \cos \omega_n x + b_n \cdot \sin \omega_n x \right),$$

die eine periodische Funktion mit einer Periode T darstellt. Diese Fourier-Reihe kann wie folgt geschrieben werden:

$$f(x) = a_0 + \sum_{n=1}^{\infty} A_n \cdot \cos(\varpi_n x + \phi_n),$$

wobei

$$A_n = \sqrt{a_n^2 + b_n^2}, \quad \phi_n = \tan^{-1}\left(\frac{b_n}{a_n}\right),$$

für n =1,2, …

Die Amplitude A_n wird als das Spektrum der Funktion bezeichnet und ermittelt die Größenordnung der Komponente von f(x) mit der Frequenz $f_n = n/T$. Die Grund- oder Fundamentalfrequenz in der Fourier-Reihe ist $f_0 = 1/T$, somit sind alle anderen Frequenzen Vielfache dieser Grundfrequenz, d.h. $f_n = n \cdot f_0$. Wir können auch eine Winkelfrequenz , $\omega_n = 2n\pi/T = 2\pi \cdot f_n = 2\pi \cdot n \cdot f_0 = n \cdot \omega_0$, bestimmen, wobei ω_0 die grundlegende oder fundamentale Winkelfrequenz der Fourier-Reihe ist.

Bei Verwendung der Winkelfrequenz-Notation wird die Fourier-Reihen-Entwicklung wie folgt geschrieben:

$$f(x) = a_0 + \sum_{n=1}^{\infty} A_n \cdot \cos(\omega_n x + \phi_n).$$

$$= a_0 + \sum_{n=1}^{\infty} \left(a_n \cdot \cos \omega_n x + b_n \cdot \sin \omega_n x \right)$$

Eine Zeichnung der Werte A_n vs. ω_n ist die typische Darstellung eines diskreten Spektrums für eine Funktion. Das diskrete Spektrum wird zeigen, dass die Funktion Komponenten bei Winkelfrequenzen ω_n aufweist, die Integervielfache der fundamentalen Winkelfrequenz ω_0 sind.

Angenommen wir müssen eine nicht periodische Funktion in eine Sinus- und Kosinus-Komponente entwickeln. Eine nicht periodische Funktion hat normalerweise eine unendlich große Periode. Somit wird für einen sehr großen Wert von T die fundamentale Winkelfrequenz $\omega_0 = 2\pi/T$ zu einer sehr kleinen Menge, z. B. $\Delta\omega$. Die Winkelfrequenz, die $\omega_n = n \cdot \omega_0 = n \cdot \Delta\omega$, $(n = 1, 2, \ldots, \infty)$ entspricht, erhält nun auch Werte, die näher und näher aneinander heranrücken, was auf die Notwendigkeit eines kontinuierlichen Spektrums von Werten hinweist.

Die nicht periodische Funktion kann deshalb wie folgt geschrieben werden:

$$f(x) = \int_0^{\infty} [C(\omega) \cdot \cos(\omega \cdot x) + S(\omega) \cdot \sin(\omega \cdot x)] d\omega,$$

wobei

$$C(\omega) = \frac{1}{2\pi} \cdot \int_{-\infty}^{\infty} f(x) \cdot \cos(\omega \cdot x) \cdot dx,$$

und

$$S(\omega) = \frac{1}{2\pi} \cdot \int_{-\infty}^{\infty} f(x) \cdot \sin(\omega \cdot x) \cdot dx$$

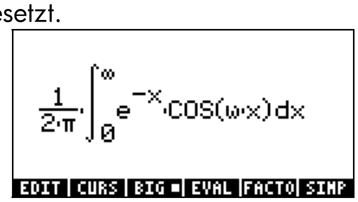
Das kontinuierliche Spektrum ist gegeben durch

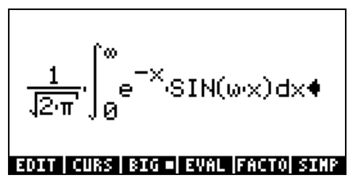
$$A(\omega) = \sqrt{[C(\omega)]^2 + [S(\omega)]^2}$$

Die Funktionen C(ω), S(ω) und A(ω) sind kontinuierliche Funktionen einer Variable ω, welche zur transformierten Variable für die Fourier-Transformation wird, die unten angeführt ist:

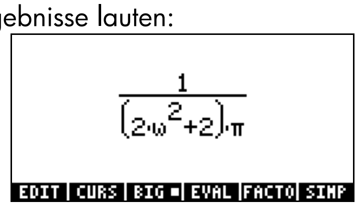
Beispiel 1 – Bestimmen Sie die Koeffizienten C(ω), S(ω) und das kontinuierliche Spektrum A(ω), für die Funktion f(x) = exp(x) für x > 0, und f(x) = 0, x < 0.

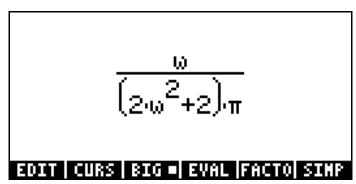
Geben Sie die folgenden Integrale in den Taschenrechner ein und werten Sie diese aus, um C(ω) und S(ω) zu berechnen. Die CAS-Modi sind auf Exact und Real gesetzt.



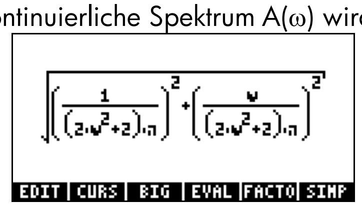


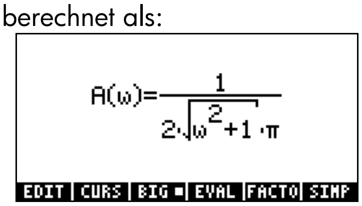
Die Ergebnisse lauten:



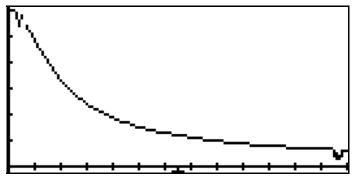


Das kontinuierliche Spektrum A(ω) wird berechnet als:





Definieren Sie den Ausdruck als Funktion durch Verwendung der Funktion DEFINE (⬅ DEF). Dann plotten Sie das ununterbrochene Spektrum, in der Strecke 0 < w < 10, wie:



Definition von Fourier-Transformationen

Es können verschiedene Arten von Fourier-Transformationen definiert werden. Nachfolgend finden Sie die Definitionen der Sinus-, Kosinus- und der vollständigen Fourier-Transformationen und deren Inversionen, die in diesem Kapitel verwendet werden:

Fourier-Sinustransformation

$$\mathrm{F}_s\{f(t)\} = F(\omega) = \frac{2}{\pi} \cdot \int_0^\infty f(t) \cdot \sin(\omega \cdot t) \cdot dt$$

Inverse-Sinustransformation

$$\mathrm{F}_s^{-1}\{F(\omega)\} = f(t) = \int_0^\infty F(\omega) \cdot \sin(\omega \cdot t) \cdot dt$$

Fourier-Kosinustransformation

$$\mathrm{F}_c\{f(t)\} = F(\omega) = \frac{2}{\pi} \cdot \int_0^\infty f(t) \cdot \cos(\omega \cdot t) \cdot dt$$

Inverse-Kosinustransformation

$$\mathrm{F}_c^{-1}\{F(\omega)\} = f(t) = \int_0^\infty F(\omega) \cdot \cos(\omega \cdot t) \cdot dt$$

Fourier-Transformation (echte)

$$\mathrm{F}\{f(t)\} = F(\omega) = \frac{1}{2\pi} \cdot \int_{-\infty}^\infty f(t) \cdot e^{-i\omega t} \cdot dt$$

Inverse Fourier-Transformation (echte)

$$\mathrm{F}^{-1}\{F(\omega)\} = f(t) = \int_{-\infty}^\infty F(\omega) \cdot e^{-i\omega t} \cdot dt$$

Beispiel 1 – Bestimmen Sie die Fourier-Transformation der Funktion f(t) = exp(-t), für t >0, und f(t) = 0, für t<0.

Das kontinuierliche Spektrum F(ω)wird mit dem folgenden Integral berechnet:

$$\frac{1}{2\pi}\int_0^{\infty} e^{-(1+i\omega)t}\,dt = \lim_{\varepsilon\to\infty}\frac{1}{2\pi}\int_0^{\varepsilon} e^{-(1+i\omega)t}\,dt$$

$$= \lim_{\varepsilon\to\infty}\frac{1}{2\pi}\left[\frac{1-\exp(-(1+i\omega)\varepsilon)}{1+i\omega}\right] = \frac{1}{2\pi}\cdot\frac{1}{1+i\omega}.$$

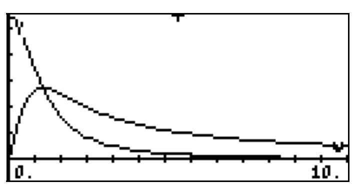
Dieses Ergebnis kann durch Multiplizieren von Zähler und Nenner mit der Konjugate des Nenners vereinfacht werden, d. h. 1-iω. Das Ergebnis lautet nun:

$$F(\omega) = \frac{1}{2\pi}\cdot\frac{1}{1+i\omega} = \frac{1}{2\pi}\cdot\left(\frac{1}{1+i\omega}\right)\cdot\left(\frac{1-i\omega}{1-i\omega}\right)$$

$$= \frac{1}{2\pi}\left(\frac{1}{1+\omega^2} - i\cdot\frac{\omega}{1+\omega^2}\right)$$

was eine komplexe Funktion ist.

Der absolute Wert der reellen und imaginären Teile der Funktion kann wie folgt gezeichnet werden:



Hinweise:

Die Größenordnung oder der Absolutwert der Fourier-Transformation $|F(\omega)|$ ist das Frequenzspektrum der ursprünglichen Funktion f(t). Für das oben angeführte Beispiel gilt $|F(\omega)| = 1/[2\pi(1+\omega^2)]^{1/2}$. Die Zeichnung für $|F(\omega)|$ vs. ω wurde vorher gezeigt.

Einige Funktionen, wie konstante Werte, sin x, exp(x), x^2, etc., haben keine Fourier-Transformation. Funktionen, die gleich schnell auf Null gehen wie x gegen Unendlich geht, haben Fourier-Transformationen.

Eigenschaften der Fourier-Transformation

Linearität: Wenn a und b Konstanten sind, und f und g Funktionen, dann ist $F\{a\cdot f + b\cdot g\} = a\, F\{f\} + b\, F\{g\}$.

Transformation partieller Ableitungen   Nehmen Sie u = u(x,t). Wenn die Fourier-Transformation die Variable x umwandelt, dann

$$F\{\partial u/\partial x\} = i\omega\, F\{u\},\quad F\{\partial^2 u/\partial x^2\} = -\omega^2\, F\{u\},$$
$$F\{\partial u/\partial t\} = \partial F\{u\}/\partial t,\quad F\{\partial^2 u/\partial t^2\} = \partial^2 F\{u\}/\partial t^2$$

Konvolution: Für Fourier-Transformations-Anwendungen ist die Konvolutionfunktion definiert als:

$$(f * g)(x) = \frac{1}{\sqrt{2\pi}} \cdot \int f(x-\xi) \cdot g(\xi) \cdot d\xi.$$

Die folgende Eigenschaft verlangt nach Konvolution:

$$F\{f*g\} = F\{f\}\cdot F\{g\}.$$

Fast Fourier-Transformation (FFT)

Die Fast Fourier-Transformation ist ein Berechnungsalgorithmus, durch welchen eine diskrete Fourier-Transformation (DFT) sehr effizient berechnet werden kann. Dieser Algorithmus findet in der Analyse verschiedener zeitabhängiger Signale Anwendung, von der Messung von Turbulenzen bis zu Kommunikationssignalen.

Die diskrete Fourier-Transformation einer Reihe von Datenwerten $\{x_j\}$, j = 0, 1, 2, …, n-1, ist eine neue endliche Folge $\{X_k\}$, definiert als

$$X_k = \frac{1}{n}\sum_{j=0}^{n-1} x_j \cdot \exp(-i \cdot 2\pi kj/n), \qquad k = 0,1,2,...,n-1$$

Die direkte Berechnung der Folge X_k bezieht n^2 Produkte ein, die einen enormen Aufwand an Computerzeit (oder Taschenrechnerzeit), vor allem für große Werte von n, benötigen würden. Die Fast Fourier-Transformation reduziert die benötigte Anzahl an Vorgängen in der Ordnung von n $\log_2 n$. Z. B. für n = 100 benötigt die FFT ungefähr 664 Vorgänge, während die direkte Berechnung 10,000 Vorgänge benötigen würde. Somit wird die Anzahl der Vorgänge mithilfe der FFT um einen Faktor von 10000/664 ≈ 15 reduziert.

Die FFT arbeitet an der Sequenz $\{x_j\}$, indem sie sie in eine Reihe kleinerer Sequenzen teilt. Die DFTs der kürzeren Sequenzen werden berechnet und später auf höchst effiziente Weise zusammengeführt. Details zum Algorithmus finden Sie z. B. in Kapitel 12 in Newland, D.E., 1993, „An Introduction to Random Vibrations, Spectral & Wavelet Analysis – Third Edition“ , Longman Scientific and Technical, New York.

Die einzige Voraussetzung für die Anwendung der FFT ist, dass die Zahl n eine Potenz von 2 ist, d.h. wählen Sie Ihre Daten so, dass 2, 4, 8, 16, 32, 62 usw. Punkte enthalten sind.

Beispiele für FFT-Anwendungen

FFT-Anwendungen enthalten normalerweise Daten, die von einem zeitabhängigen Signal diskretisiert wurden. Der Taschenrechner kann über einen Computer oder Daten-Logger mit diesen Daten gespeist werden, damit sie verarbeitet werden können. Sie können auch Ihre eigenen Daten verändern, indem Sie eine Funktion programmieren und einige zufällige Zahlen hinzufügen.

Beispiel 1 – Definieren Sie die Funktion f(x) = 2 sin (3x) + 5 cos(5x) + 0,5*RAND, wobei RAND der gleichförmige Zufalls-Zahlengenerator ist, der vom Taschenrechner erstellt wird. Erzeugen Sie 128 Datenpunkte mithilfe der Werte von x im Intervall (0;12,8). Speichern Sie diese Werte in einer Matrix und führen Sie eine FFT an der Matrix durch.

Zuerst definieren wir die Funktion f(x) als RPN-Programm:

<< →x '2*SIN(3*x) + 5*COS(5*x)' EVAL RAND 5 * + →NUM >>

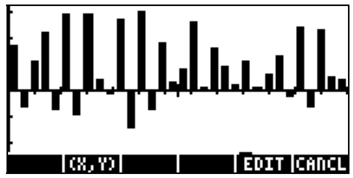
und speichern dieses Programm in Variable f. Geben Sie dann das folgende Programm ein, um 2^m Datenwerte zwischen a und b zu errechnen. Das Programm wird die Werte von m, a und b verwenden:

<< → m a b << '2^m' EVAL → n << '(b-a)/(n+1)' EVAL → Dx << 1 n FOR j 'a+(j-1)*Dx' EVAL f NEXT n →ARRY >> >> >> >>

Speichern Sie dieses Programm unter dem Namen GDATA (Generate DATA). Führen Sie dann das Programm für die folgenden Werte aus: m = 5, a = 0, b = 100.. Verwenden Sie im RPN-Modus:

5 SPC 0 SPC 1 0 0 GDATA

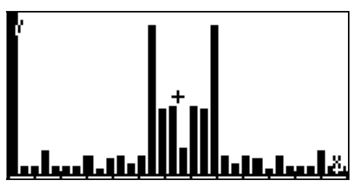
Die unten angeführte Abbildung ist ein Box-Plot der erstellten Daten. Um die Grafik zu erstellen, kopieren Sie zuerst die angelegte Matrix und transformieren Sie diese dann in einen Spaltenvektor unter Verwendung von: OBJ→ 1 + →ARRY (Funktionen OBJ→ und →ARRY sind im Befehlskatalog enthalten → CAT). Speichern Sie die Matrix in ΣDAT mithilfe der Funktion STOΣ (auch über → CAT verfügbar). Wählen Sie Bar im TYPE für die Grafik und setzen Sie das Ansichtsfenster auf H-VIEW: 0 32, V-VIEW: -10 10, und BarWidth auf 1. Drücken Sie CANCL ON , um zum normalen Taschenrechnerdisplay zurückzukehren.



Um die FFT an der Matrix auf der Speicherebene 1 durchzuführen, verwenden Sie die Funktion FFT im Menü MTH/FFT auf die Matrix ΣDAT: ΣDAT FFT. Die FFT errechnet eine Matrix mit komplexen Zahlen, welche die Matrix für die Koeffizienten X_k der DFT sind. Die Größenordnung der Koeffizienten X_k stellt ein Frequenzspektrum der ursprünglichen Daten dar. Um die Größenordnung der Koeffizienten zu erhalten, könnten Sie die Matrix mithilfe der Funktion AXL in eine Liste umwandeln und dann die Funktion ABS an der Liste anwenden. Dies erfolgt durch: OBJ→ EVAL ◄ →LIST ← ABS

Schlussendlich können Sie wie folgt die Liste erneut zurück in einen Spaltenvektor konvertieren, welcher in ΣDAT gespeichert wird: OBJ→ 1 ENTER 2 →LIST →ARRY STOΣ

Um das Spektrum zu zeichnen, folgen Sie den Anweisungen für die Erstellung einer Balkengrafik, wie bereits beschrieben. Der vertikale Bereich muss auf –1 bis 80 geändert werden. Das Frequenzspektrum sieht wie folgt aus gt:



Das Spektrum zeigt zwei große Komponenten für zwei Frequenzen (dies sind die sinusförmigen Komponenten sin (3x) und cos(5x)) und eine Reihe kleinerer Komponenten für andere Frequenzen.

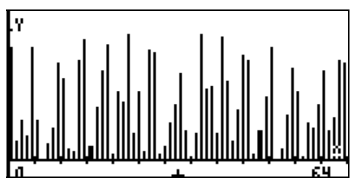
<u>Beispiel 2</u> – Um das Signal zu produzieren, welches durch das Spektrum gegeben ist, ändern wir das Programm GDATA so, dass es einen absoluten Wert beinhaltet und somit wie folgt lautet:

<< → m a b << '2^m' EVAL → n << '(b-a)/(n+1)' EVAL → Dx << 1 n FOR j 'a+(j-1)*Dx' EVAL f ABS NEXT n →ARRY >> >> >> >>

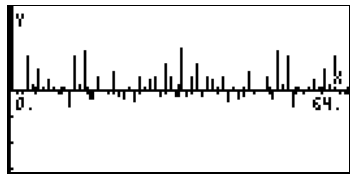
Speichern Sie diese Programmversion unter dem Namen GSPEC (Generate SPECtrum). Führen Sie das Programm dann für die Werte m = 6, a = 0, b = 100 aus, indem Sie folgendes verwenden:

6 SPC 0 SPC 1 0 0 GSPEC

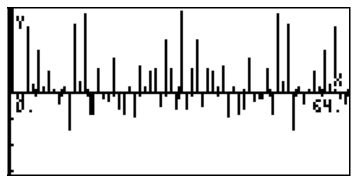
Drücken Sie nach Abschluss ENTER, um eine zusätzliche Kopie der Spektrum-Matrix zu behalten. Konvertieren Sie diesen Reihenvektor in einen Spaltenvektor und speichern Sie diesen in ΣDAT . Befolgen Sie den Vorgang für die Erstellung einer Balkengrafik, dann schaut das für dieses Beispiel erstellte Spektrum wie unten aus. Der horizontale Bereich in diesem Fall ist 0 bis 64 und der vertikale Bereich ist –1 bis 10:



Um das Signal wiederzugeben, für welches das Spektum angezeigt wird, verwenden Sie die Funktion IFFT. Da wir eine Kopie des Spektrums gespeichert haben (einen Reihenvektor), brauchen Sie nur die Funktion IFFT im Menü MTH/FFT oder über den Befehlskatalog ↱ CAT finden. Eine weitere Möglichkeit ist die Eingabe des Funktionsnamens, d.h. geben Sie ein: ALPHA ALPHA I F F T ENTER . Das Signal wird als Matrix (Reihenvektor) mit komplexen Zahlen angezeigt. Wir interessieren uns nur für den realen Teil der Elemente. Um den realen Teil aus den komplexen Zahlen zu ziehen, verwenden Sie die Funktion RE aus dem Menü CMPLX (siehe Kapitel 4), d.h. geben Sie ALPHA ALPHA R E ENTER ein. Daraus resultiert ein weiterer Reihenvektor. Konvertieren Sie ihn zu einem Spaltenvektor, speichern Sie ihn in ΣDAT und erstellen Sie eine Balkengrafik, um das Signal zu zeigen. Das Signal für dieses Beispiel ist unten dargestellt mit einem horizontalen Bereich von 0 bis 64 und einem vertikalen Bereich von –1 bis 1:



Außer einer großen Spitze bei t = 0, ist das Signal vorwiegend Geräusch. Ein kleinerer vertikaler Bereich (-0,5 to 0,5) zeigt das Signal folgendermaßen:



Lösung spezifischer Differentialgleichungen zweiter Ordnung

In diesem Abschnitt zeigen und lösen wir verschiedene Arten gewöhnlicher Differentialgleichungen, deren Lösungen nach einigen klassischen Funktionen definiert werden, z. B. Bessel-Funktion, Hermite'sche Polynome usw. Die Beispiele sind im RPN-Modus angeführt.

Die Cauchy'sche oder Euler-Gleichung

Eine Gleichung der Form $x^2\cdot(d^2y/dx^2) + a\cdot x\cdot (dy/dx) + b\cdot y = 0$, wobei a und b reale Konstanten sind, ist als Cauchy'sche oder Euler-Gleichung bekannt. Eine Lösung zur Cauchy'schen Gleichung kann gefunden werden unter der Annahme, dass $y(x) = x^n$.
Geben Sie die Gleichung wie folgt ein:
'x^2*d1d1y(x)+a*x*d1y(x)+b*y(x)=0' ENTER
Geben Sie dann die vorgeschlagene Lösung ein und ersetzen Sie diese durch
'y(x) = x^n' ENTER SUBST

Das Ergebnis lautet: 'x^2*(n*(x^(n-1-1)*(n-1)))+a*x*(n*x^(n-1))+b*x^n =0, vereinfacht 'n*(n-1)*x^n+a*n*x^n+b*x^n = 0'. Durch eine Division durch x^n erhalten wir eine algebraische Hilfsgleichung: 'n*(n-1)+a*n+b = 0', oder.

$$n^2 + (a-1)\cdot n + b = 0 \,.$$

- Wenn die Gleichung zwei verschiedene Wurzeln enthält, z. B. n_1 und n_2, dann ist die allgemeine Lösung zu dieser Gleichung $y(x) = K_1 \cdot x^{n_1} + K_2 \cdot x^{n_2}$.
- Wenn $b = (1-a)^2/4$, dann hat die Gleichung eine Doppelwurzel $n_1 = n_2 = n = (1-a)/2$, und die Lösung lautet $y(x) = (K_1 + K_2 \cdot \ln x)x^n$.

Legendre'sche Gleichung

Eine Gleichung der Form $(1-x^2)\cdot(d^2y/dx^2)-2\cdot x\cdot (dy/dx)+n\cdot (n+1) \cdot y = 0$, wobei n eine reale Zahl ist, ist als Legendre'sche Gleichung bekannt. Jede Lösung für diese Gleichung wird Legendre'sche Funktion genannt. Wenn n eine nicht negative Ganzzahl ist, nennt man die Lösungen Legendre'sche Polynome. Das Legendre'sche Polynom von Ordnung n ist gegeben durch

$$P_n(x) = \sum_{m=0}^{M} (-1)^m \cdot \frac{(2n-2m)!}{2^n \cdot m! \cdot (n-m)! \cdot (n-2m)!} \cdot x^{n-2m}$$

$$= \frac{(2n)!}{2^n \cdot (n!)^2} \cdot x^n - \frac{(2n-2)!}{2^n \cdot 1! \cdot (n-1)!(n-2)!} \cdot x^{n-2} + \ldots - ..$$

wobei M = n/2 oder (n-1)/2, je nachdem was eine Ganzzahl ist.

Legendre'sche Polynome sind im Taschenrechner vorprogrammiert und können mithilfe der Funktion LEGENDRE aufgerufen werden, falls die Ordnung des Polynoms n gegeben ist. Die Funktion LEGENDRE kann über den Befehlskatalog (↱ _CAT_) oder über das Menü ARITHMETIC/POLYNOMIAL aufgerufen werden (siehe Kapitel 5). Im RPN-Modus erhält man die ersten sechs Legendre'schen Polynome wie folgt:

0 LEGENDRE, Ergebnis: 1,	d.h. $P_0(x) = 1.0$.
1 LEGENDRE, Ergebnis: 'X',	d.h. $P_1(x) = x$.
2 LEGENDRE, Ergebnis: '(3*X^2-1)/2',	d.h. $P_2(x) = (3x^2-1)/2$.
3 LEGENDRE, Ergebnis: '(5*X^3-3*X)/2',	d.h. $P_3(x) =(5x^3-3x)/2$.

4 LEGENDRE, Ergebnis: '(35*X^4-30*X^2+3)/8', d.h.
$P_4(x) = (35x^4-30x^2+3)/8$.
5 LEGENDRE, Ergebnis: '(63*X^5-70*X^3+15*X)/8', d.h.
$P_5(x) = (63x^5-70x^3+15x)/8$.

Die ODE $(1-x^2)\cdot(d^2y/dx^2)-2\cdot x\cdot (dy/dx)+[n\cdot (n+1)-m^2/(1-x^2)] \cdot y = 0$, hat folgende Funktion als Lösung: $y(x) = P_n^m(x)= (1-x^2)^{m/2}\cdot(d^mPn/dx^m)$. Diese Funktion wird eine assozierte Legendre-Funktion genannt.

Bessel-Gleichung

Die gewöhnliche Differentialgleichung $x^2\cdot(d^2y/dx^2) + x\cdot (dy/dx)+ (x^2-\nu^2) \cdot y = 0$, wobei der Parameter ν eine nicht negative reale Zahl ist, wird Bessel'sche Differentialgleichung genannt. Lösungen zu Bessel-Gleichungen sind als Bessel-Funktionen erster Gattung der Ordnung ν gegeben:

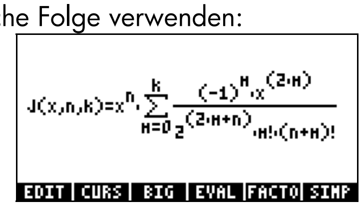
$$J_\nu(x) = x^\nu \cdot \sum_{m=0}^{\infty} \frac{(-1)^m \cdot x^{2m}}{2^{2m+\nu} \cdot m! \cdot \Gamma(\nu+m+1)},$$

wobei ν keine Ganzzahl ist und die Funktion Gamma Γ(α) ist in Kapitel 3 definiert.

Wenn ν = n eine Ganzzahl ist, werden die Bessel-Funktionen erster Gattung für n = Ganzzahl definiert durch

$$J_n(x) = x^n \cdot \sum_{m=0}^{\infty} \frac{(-1)^m \cdot x^{2m}}{2^{2m+n} \cdot m! \cdot (n+m)!}.$$

Egal, ob wir im Taschenrechner ν (keine Ganzzahl) oder n (Ganzzahl) verwenden, können wir die Bessel-Funktion erster Gattung definieren, indem wir die folgende endliche Folge verwenden:



Somit haben wir die Kontrolle über Funktionsordnung n und über die Anzahl der Elemente in der Reihe k. Sobald Sie diese Funktion eingegeben haben, können Sie die Funktion DEFINE verwenden, um die Funktion J(x,n,k) zu definieren. Dies ergibt die Variable [J] in den Softmenü-Tasten. Um z. B. $J_3(0,1)$ mithilfe von 5 Termen in der Reihe auszuwerten, berechnen Sie J(0.1,3,5), z. B. im RPN-Modus: [.] [1] [SPC] [3] [SPC] [5] [J]. Das Ergebnis lautet 2,08203157E-5.

Wenn Sie einen Ausdruck für $J_0(x)$ mit beispielsweise 5 Termen in der Reihe erhalten möchten, verwenden Sie J(x,0,5). Das Ergebnis lautet:

‘1-0,25*x^3+0,015625*x^4-4,3403777E-4*x^6+6,782168E-6*x^8-6,78168*x^10’.

Für nicht ganzzahlige Werte ν wird die Lösung zur Bessel-Gleichung gegeben durch:

$$y(x) = K_1 \cdot J_\nu(x)+K_2 \cdot J_{-\nu}(x).$$

Für ganzzahlige Werte sind die Funktionen Jn(x) und J-n(x) linear abhängig, da

$$J_n(x) = (-1)^n \cdot J_{-n}(x),$$

deshalb können wir sie nicht verwenden, um eine allgemeine Funktion für die Gleichung zu erhalten. Stattdessen verwenden wir die <u>Bessel-Funktionen zweiter Gattung</u> definiert als

$$Y_\nu(x) = [J_\nu(x) \cos \nu\pi - J_{-\nu}(x)]/\sin \nu\pi,$$

für nicht ganze Zahlen ν, und für n Ganzzahl mit n > 0, durch

$$Y_n(x) = \frac{2}{\pi} \cdot J_n(x) \cdot (\ln\frac{x}{2} + \gamma) + \frac{x^n}{\pi} \cdot \sum_{m=0}^{\infty} \frac{(-1)^{m-1} \cdot (h_m + h_{m+n})}{2^{2m+n} \cdot m! \cdot (m+n)!} \cdot x^{2m}$$

$$-\frac{x^{-n}}{\pi} \cdot \sum_{m=0}^{n-1} \frac{(n-m-1)!}{2^{2m-n} \cdot m!} \cdot x^{2m}$$

wobei γ die Euler-Konstante ist, definiert durch

$$\gamma = \lim_{r\to\infty}[1+\frac{1}{2}+\frac{1}{3}+...+\frac{1}{r}-\ln r] \approx 0.57721566490...,$$

und h_m stellt die harmonische Reihe dar.

$$h_m = 1+\frac{1}{2}+\frac{1}{3}+...+\frac{1}{m}$$

Für den Fall n = 0 wird die Bessel-Funktionen zweiter Gattung definiert als

$$Y_0(x) = \frac{2}{\pi}\cdot\left[J_0(x)\cdot(\ln\frac{x}{2}+\gamma)+\sum_{m=0}^{\infty}\frac{(-1)^{m-1}\cdot h_m}{2^{2m}\cdot(m!)^2}\cdot x^{2m}\right].$$

Bei diesen Definitionen ist eine allgemeine Lösung der Bessel-Gleichung für alle Werte von ν gegeben durch $y(x) = K_1\cdot J_\nu(x)+K_2\cdot Y_\nu(x)$.

In einigen Fällen ist es nötig, die Bessel-Gleichung mit einer komplexen Lösung zu versehen, indem die Bessel-Funktion dritter Gattung der Ordnung ν definiert wird als

$H_n^{(1)}(x) = J_\nu(x)+i\cdot Y_\nu(x)$, and $H_n^{(2)}(x) = J_\nu(x)-i\cdot Y_\nu(x)$,

Diese Funktionen sind auch als erste und zweite Hankel-Funktionen der Ordnung ν bekannt.

Bei einigen Anwendungen kann es sein, dass Sie die sogenannte modifizierte Bessel-Funktionen erster Gattung der Ordnung ν verwenden müssen, definiert als $I_\nu(x)= i^{-\nu}\cdot J_\nu(i\cdot x)$, wobei i die imaginäre Zahl der Einheit ist. Diese Funktionen sind Lösungen zur Differentialgleichung $x^2\cdot(d^2y/dx^2) + x\cdot (dy/dx)- (x^2+\nu^2) \cdot y = 0$.

Die modifizierten Bessel-Funktionen zweiter Gattung

$K_\nu(x) = (\pi/2)\cdot[I_{-\nu} (x)-I_\nu (x)]/\sin \nu\pi$, sind auch Lösungen dieser ODE.

Sie können Funktionen anwenden, indem Sie die Bessel-Funktionen im Taschenrechner auf ähnliche Weise verwenden, wie sie zur Definition von Bessel-Funktionen erster Gattung verwendet werden, indem Sie darauf achten, dass die unendlichen Reihen im Taschenrechner in endliche Reihen umgewandelt werden müssen.

Chebyshev oder Tschebycheff-Polynome

Die Funktionen $T_n(x) = \cos(n\cdot\cos^{-1} x)$, und $U_n(x) = \sin[(n+1) \cos^{-1} x]/(1-x^2)^{1/2}$, n = 0, 1 werden Chebyshev oder Tschebycheff-Polymone erster bzw. zweiter Gattung genannt. Die Polynome Tn(x) sind Lösungen zur Differentialgleichung $(1-x^2)\cdot(d^2y/dx^2) - x\cdot (dy/dx) + n^2\cdot y = 0$.

Im Rechner erzeugt die Funktion TCHEBYCHEFF das Chebyshev oder Tchebycheff Polynom der ersten Ordnung n, einen Wert von gegeben n > 0. Wenn die Ganzzahl n (n < 0) negativ ist, erzeugt die Funktion TCHEBYCHEFF ein Tchebycheff Polynom der zweiten Ordnung n dessen Definition ist

$$U_n(x) = \sin(n\cdot\arccos(x))/\sin(\arccos(x)).$$

Sie können über den Befehlskatalog (↱ CAT) auf die Funktion TCHEBYCHEFF zugreifen.

Die ersten vier Chebyshev- oder Tschebycheff-Polynome erster und zweiter Gattung erhält man wie folgt:

0 TCHEBYCHEFF, Ergebnis: 1,	d.h.	$T_0(x) = 1.0$.
-0 TCHEBYCHEFF, Ergebnis: 1,	d.h.	$U_0(x) = 1.0$.
1 TCHEBYCHEFF, Ergebnis: 'X',	d.h.	$T_1(x) = x$.
-1 TCHEBYCHEFF, Ergebnis: 1,	d.h.	$U_1(x) =1.0$.
2 TCHEBYCHEFF, Ergebnis: '2*X^2-1,	d.h.	$T_2(x) =2x^2-1$.
-2 TCHEBYCHEFF, Ergebnis: '2*X',	d.h.	$U_2(x) =2x$.
3 TCHEBYCHEFF, Ergebnis: '4*X^3-3*X',	d.h.	$T_3(x) = 4x^3-3x$.
-3 TCHEBYCHEFF, Ergebnis: '4*X^2-1',	d.h.	$U_3(x) = 4x^2-1$.

Laguerre-Gleichung

Laguerre's Gleichung ist die lineare ODE zweiter Ordnung der Form $x\cdot(d^2y/dx^2) + (1-x)\cdot(dy/dx) + n\cdot y = 0$. Laguerre-Polynome, definiert als

$$L_0(x) = 1, \quad L_n(x) = \frac{e^x}{n!}\cdot\frac{d^n(x^n\cdot e^{-x})}{dx^n},\ n = 1,2,\ldots,$$

sind Lösungen zur Laguerre-Gleichung. Laguerre-Polynome können auch berechnet werden mit

$$L_n(x) = \sum_{m=0}^{n}\frac{(-1)^m}{m!}\cdot\binom{n}{m}\cdot x^m.$$

$$= 1 - n\cdot x + \frac{n(n-1)}{4}\cdot x^2 - \ldots + \ldots. + \frac{(-1)^n}{n!}\cdot x^n$$

Der Term

$$\binom{n}{m} = \frac{n!}{m!(n-m)!} = C(n,m)$$

ist der m-te Koeffizient der Binomialentwicklung $(x+y)^n$. Sie steht auch für die Darstellung der Anzahl der Kombinationen von n-Elementen m auf einmal. Diese Funktion ist im Taschenrechner als Funktion COMB im Menü MTH/PROB verfügbar (siehe auch Kapitel 17).

Sie können die folgende Funktion definieren, um Laguerre-Polynome zu berechnen:

$$L(x,n) = \sum_{m=0}^{n}\frac{(-1)^m}{m!}\cdot COMB(n,m)\cdot x^m$$

Wenn Sie die Definition in den Gleichungsschreiber eingegeben haben, drücken Sie die Funktion DEFINE, um die Funktion L(x,n) in der Variable L zu erzeugen.

Um die ersten vier Laguerre-Polynome zu generieren, verwenden Sie L(x,0), L(x,1), L(x,2), L(x,3). Die Ergebnisse lauten:

$L_0(x) = .$
$L_1(x) = 1-x.$
$L_2(x) = 1-2x+ 0.5x^2$
$L_3(x) = 1-3x+1.5x^2-0.16666\ldots x^3.$

Weber-Gleichung und Hermite-Polynome

Die Weber-Gleichung wird definiert als: $d^2y/dx^2+(n+1/2-x^2/4)y = 0$, für n = 0, 1, 2, … Eine besondere Lösung dieser Gleichung ist durch die Funktion $y(x) = \exp(-x^2/4)H^*(x/\sqrt{2})$ gegeben, wobei die Funktion $H^*(x)$ folgendes Hermite-Polynom ist:

$$H_0{}^* = 1, \quad H_n{}^*(x) = (-1)^n e^{x^2} \frac{d^n}{dx^n}(e^{-x^2}), \quad n = 1,2,..$$

Im Taschenrechner ist die Funktion HERMITE über das Menü ARITHMETIC/POLYNOMIAL verfügbar. Die Funktion HERMITE hat als Argument eine Ganzzahl n und ergibt das Hermite-Polynom des n-ten Grades. Die ersten vier Hermite-Polynome können z. B. erhalten werden durch Verwendung von:

0 HERMITE, Ergebnis: 1,	d.h. $H_0{}^* = 1$.
1 HERMITE, Ergebnis: '2*X',	d.h. $H_1{}^* = 2x$.
2 HERMITE, Ergebnis: '4*X^2-2',	d.h. $H_2{}^* = 4x^2-2$.
3 HERMITE, Ergebnis: '8*X^3-12*X',	d.h. $H_3{}^* = 8x^3-12x$.

Numerische und grafische Lösungen von ODE

Differentialgleichungen, die nicht analytisch gelöst werden können, können numerisch oder grafisch gelöst werden, wie unten aufgezeigt.

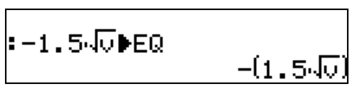
Numerische Lösung einer ODE erster Ordnung

Durch die Verwendung des numerischen Lösers (⇁ NUM.SLV), können Sie auf eine Eingabeform zugreifen, mit deren Hilfe Sie lineare gewöhnliche

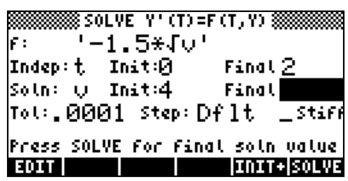
Differentialgleichungen erster Ordnung lösen können. Die Anwendung dieser Funktion wird mit folgendem Beispiel vorgestellt. Die in der Lösung verwendete Methode ist ein Runge-Kutta Algorithmus vierter Ordnung, der im Taschenrechner vorprogrammiert ist.

<u>Beispiel 1</u> – Angenommen wir möchten die Differentialgleichung dv/dt = -1,5 $v^{1/2}$, mit v = 4 at t = 0 lösen. Wir müssen v für t = 2 finden.

Zuerst erstellen wir den Ausdruck, indem wir die Ableitung definieren und sie in der Variable EQ speichern. Die Abbildung links zeigt den Befehl im ALG-Modus und die rechte Abbildung zeigt den RPN-Speicher vor Drücken von STO▶ .



Geben Sie dann die Umgebung des NUMERICAL SOLVERs ein und wählen Sie den Differentialgleichungs-Löser ↱ NUM.SLV ▼ OK Geben Sie die folgenden Parameter ein:



Zum Lösen drücken Sie: SOLVE (warten) EDIT. Das Ergebnis lautet 0,2499 ≈ 0,25. Drücken Sie OK.

Lösung dargestellt als Wertetabelle
Angenommen wir möchten eine Wertetabelle von v für t = 0,00; 0,25; …, 2,00, anfertigen, dann gehen wir wie folgt vor:

Erstellen Sie zuerst die Tabellen, in die Sie die Ergebnisse eintragen. Schreiben Sie das Schritt-für-Schritt-Ergebnis in die Tabelle:

t	v
0.00	0.00
0.25	
…	…
2.00	

Als nächstes ändern Sie in der SOLVE-Umgebung den endgültigen Wert der unabhängigen Variable auf 0,25, verwenden Sie dazu:

.25 OK SOLVE (warten) EDIT
(Löst auf nach v bei t = 0,25, v = 3,285 ….)
OK INIT+ .5 OK SOLVE (warten) EDIT
(Verändert Anfangswert von t auf 0,25 und endgültigen Wert von t auf 0,5, lösen Sie auf nach v(0,5) = 2,640…).
OK INIT+ .75 OK SOLVE (warten) EDIT
(Verändert Anfangswert von t auf 0,5 und endgültigen Wert von t auf 0,75, lösen Sie auf nach v(0,75) = 2,066…).
OK INIT+ 1 OK SOLVE (warten) EDIT
(Verändert Anfangswerte von t auf 0,75 und endgültigen Wert von t auf 1, lösen Sie auf nach v(1) = 1,562…).
Wiederholen Sie für t = 1,25; 1,50; 1,75; 2,00. Drücken Sie OK, nachdem Sie sich das letzte Ergebnis in EDIT angesehen haben. Um zum normalen Taschenrechnerdisplay zurückzukehren, drücken Sie ON oder NXT OK. Die verschiedenen Lösungen werden im Speicher angezeigt, und das letzte Ergebnis in Ebene 1.

Die endgültigen Ergebnisse sehen wie folgt aus (auf die dritte Dezimalstelle gerundet):

t	**v**
0.00	4.000
0.25	3.285
0.50	2.640
0.75	2.066

1.00	1.562
1.25	1.129
1.50	0.766
1.75	0.473
2.00	0.250

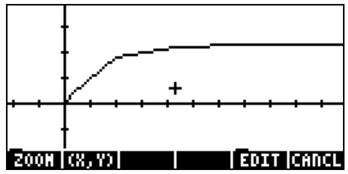
Grafische Lösung einer ODE erster Ordnung

Wenn wir keine Lösung geschlossener Form für das Integral erhalten können, können wir das Integral zeichnen, indem im Feld `TYPE` der PLOT-Umgebung die Funktion `Diff Eq` wie folgt auswählen: Angenommen wir wollen die Position x(t) für eine Schnelligkeitsfunktion v(t) = exp(-t^2), mit x = 0 bei t = 0 zeichnen. Wir wissen, dass es für das Integral keinen Ausdruck geschlossener Form gibt, wir wissen jedoch, dass die Definition von v(t) wie folgt lautet: dx/dt = exp(-t^2).

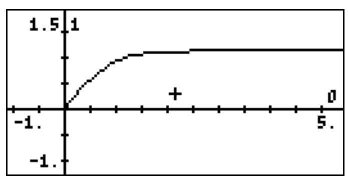
Der Taschenrechner ermöglicht das Zeichnen der Lösung der Differentialgleichungen der Form Y'(T) = F(T,Y). In unserem Fall nehmen wir an Y = x und T = t, darum, F(T,Y) = f(t, x) = exp(-t^2). Zeichnen wir nun die Lösung x(t), für t = 0 bis 5, indem wir die folgende Eingabe-Abfolge verwenden:

- ← 2D/3D (gleichzeitig, wenn im RPN-Modus), um in die PLOT—Umgebung zu gelangen.
- Markieren Sie das Feld vor TYPE mithilfe der Tasten ▲▼. Drücken Sie dann CHOOS, und markieren Sie `Diff Eq` mithilfe der Tasten ▲▼. Drücken Sie OK.
- Ändern Sie Feld F: auf 'EXP(- t^2)'.
- Vergewissern Sie sich, dass die folgenden Parameter gesetzt sind auf: `H-VAR: 0, V-VAR: 1`
- Ändern Sie die unabhängige Variable zu t.
- Bestätigen Sie die Änderungen von PLOT SETUP: NXT OK
- ← WIN (gleichzeitig, wenn im RPN-Modus). Um in die Umgebung PLOT WINDOW zu gelangen
- ändern Sie das horizontale und vertikale Ansichtsfenster auf die folgenden Einstellungen: `H-VIEW: -1     5;  V-VIEW: -1       1.5`
- Verwenden Sie die folgenden Werte auch für die verbleibenden Parameter: Init: 0, Final: 5, Step: Default, Tol: 0,0001, Init-Soln: 0

- Um die Grafik zu zeichnen, verwenden Sie: ERASE DRAW
-



Wenn Sie beobachten, wie die Grafik gezeichnet wird, werden Sie erkennen, dass die Grafik nicht sehr fein ist. Der Grund dafür ist, dass der Zeichner einen Zeitschritt verwendet, der für eine feine Zeichnung zu groß sein könnte. Um die Grafik zu vereinfachen und feiner zu gestalten, verwenden Sie einen Schritt von 0,1. Drücken Sie CANCL und ändern Sie den *Step*-Wert auf 0,1. Verwenden Sie dann ERASE DRAW erneut, um die Grafik zu wiederholen. Der Zeichenvorgang wird länger dauern, aber die Form ist deutlich feiner als zuvor. Versuchen Sie Folgendes: EDIT NXT LABEL MENU, um den Achsennamen und den Bereich zu sehen.



Beachten Sie, dass die Namen für die Achsen als 0 (horizontal für t) und 1 (vertikal für x) angezeigt werden. Die Definitionen für die im Fenster PLOT SETUP gezeigten (← 2D/3D) Achsen lauten H-VAR : 0, und V-VAR: 1. Um die grafische Lösung im Detail zu sehen, verwenden Sie:

NXT NXT PICT	um das Menü wieder herzustellen und zur PICT-Umgebung zurückzukehren.
(X,Y)	Um die Koordinaten jedes beliebigen Punktes in der Grafik zu bestimmen,

verwenden Sie ◀ ▶ Tasten, um den Cursor um den Zeichnungsbereich zu bewegen. Unten im Bildschirm sehen Sie die Koordinaten des Cursors als (X,Y), d.h. der Taschenrechner verwendet X und Y als Standardnamen für die horizontalen und die vertikalen Achsen. Drücken Sie NXT CANCL, um das Menü wiederherzustellen und zur Umgebung PLOT WINDOW zurückzukehren. Drücken Sie schlussendlich ON, um in das normale Display zurückzukehren.

Numerische Lösung einer ODE zweiter Ordnung

Die Integration einer ODE zweiter Ordnung kann durch Definieren der Lösung als Vektor geschehen. Als Beispiel nehmen wir an, dass ein Feder-Masse-System einer Dämpfungskraft ausgesetzt ist, die sich proportional zu seiner Geschwindigkeit verhält, sodass die daraus resultierende Differentialgleichung lautet:

$$\frac{d^2x}{dt^2} = -18.75 \cdot x - 1.962 \cdot \frac{dx}{dt}$$

oder, $x'' = -18.75\,x - 1.962\,x'$,

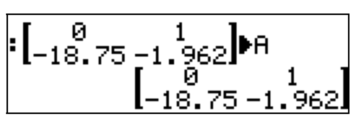
den Anfangsbedingunen v = x' = 6, x = 0, at t = 0 ausgesetzt ist. Wir möchten x, x' bei t = 2 finden.

Schreiben Sie die ODE neu als: **w**' = **Aw**, wobei **w** = [x x']^T, und **A** ist die 2 x 2 Matrix, die unten angeführt wird.

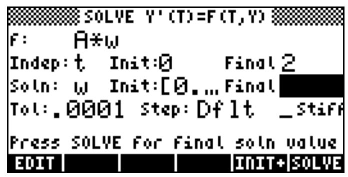
$$\begin{bmatrix} x \\ x' \end{bmatrix}' = \begin{bmatrix} 0 & 1 \\ -18.75 & -1.962 \end{bmatrix} \cdot \begin{bmatrix} x \\ x' \end{bmatrix}$$

Die Anfangsbedingungen werden nun geschrieben als **w** = [0 6]^T, für t = 0. (Hinweis: Das Symbol [ ]^T steht für die Transponierte des Vektors oder der Matrix).

Um dieses Problem zu lösen, erstellen Sie zuerst die Matrix **A** und speichern Sie diese, z. B. im ALG-Modus:



Aktivieren Sie dann den numerischen Differentialgleichungs-Löser mithilfe von: , ↱ NUM.SLV ▼ OK. Um die Differentialgleichung mit Startzeit t = 0 und Endzeit t = 2 zu lösen, sollte die Eingabeform für den Differentialgleichungs-Löser wie folgt aussehen (achten Sie darauf, dass der Init:-Wert für Soln: ein Vektor ist [0, 6]):



Drücken Sie SOLVE (warten) EDIT , um nach w(t=2) aufzulösen. Die Lösung lautet [,16716… -,6271…], d.h., x(2) = 0,16716, und x'(2) = v(2) = -0,6271. Drücken Sie CANCL , um in die SOLVE-Umgebung zurückzukehren.

Lösung als Wertetabelle

Im vorherigen Beispiel waren wir nur daran interessiert, die Werte für Position und Schnelligkeit bei einer bestimmten Zeit t zu finden. Wenn wir eine Wertetabelle von x und x', für t = 0,00; 0,25, …, 2.00, erstellen möchten, gehen wir wie folgt vor: Erstellen Sie zuerst die Tabelle, in die Sie die Ergebnisse eintragen:

t	x	x'
0.00	0.00	6.00
0.25		
…	…	…
2.00		

Als nächstes ändern Sie in der SOLVE-Umgebung den endgültigen Wert der unabhängigen Variable auf 0,25, verwenden Sie dazu:

▲ .25 OK ▶ ▶ SOLVE (warten) EDIT
(Löst auf nach w bei t = 0,25, w = [0,968 1,368].)
OK INIT+ ▲ , 5 OK ▶ ▶ SOLVE (warten) EDIT

(Verändert Anfangswert von t auf 0,25 und endgültigen Wert von t auf 0,5, lösen Sie nochmals auf nach w(0,5) = [0,748 –2,616])

OK INIT+ ▲ .75 OK ▶ ▶ SOLVE (warten) EDIT

(Verändert Anfangswert von t auf 0,5 und endgültigen Wert von t auf 0,75, lösen Sie nochmals nach auf w(0,75) = [0,0147 –2,859])

OK INIT+ ▲ 1 OK ▶ ▶ SOLVE (warten) EDIT

(Verändert Anfangswert von t auf 0,75 und endgültigen Wert von t auf 1, lösen Sie nochmals auf nach w(1) = [-0,469 -0,607])

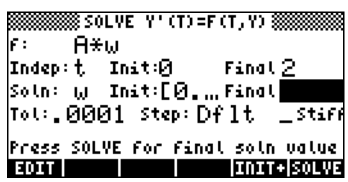
Wiederholen Sie dies für t = 1,25; 1,50; 1,75; 2,00. Drücken Sie OK nach Ansicht des letzten Ergebnisses in EDIT. Um zum normalen Taschenrechnerdisplay zurückzukehren, drücken Sie ON oder NXT OK. Die verschiedenen Lösungen werden im Speicher angezeigt, und das letzte Ergebnis in Ebene 1.

Die Endergebnisse sehen wie folgt aus:

t	x	x'		t	x	x'
0.00	0.000	6.000		1.25	-0.354	1.281
0.25	0.968	1.368		1.50	0.141	1.362
0.50	0.748	-2.616		1.75	0.227	0.268
0.75	-0.015	-2.859		2.00	0.167	-0.627
1.00	-0.469	-0.607				

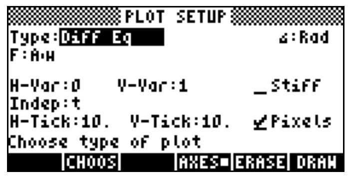
Grafische Lösung einer ODE zweiter Ordnung

Beginnen Sie mit der Aktivierung des numerischen Differentialgleichungs-Lösers ↱ NUM.SLV ▼ OK. Der SOLVE-Bildschirm sollte wie folgt aussehen:



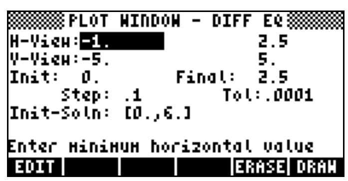
Beachten Sie, dass die Anfangsbedingung für die Lösung (Soln: w Init:[0., …) den Vektor [0, 6] enthält. Drücken Sie NXT OK.

Drücken Sie dann ← 2D/3D (gleichzeitig, wenn im RPN-Modus), um in die PLOT—Umgebung zu gelangen. Markieren Sie das Feld vor `TYPE`, mithilfe der Tasten ▲▼. Drücken Sie dann CHOOS, und markieren Sie `Diff Eq` mithilfe der Tasten ▲▼. Drücken Sie OK. Ändern Sie den Rest des Bildschirms PLOT SETUP, damit er wie folgt aussieht:

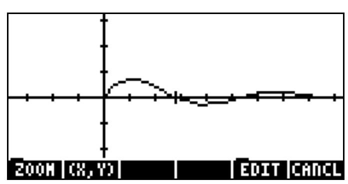


Beachten Sie, dass die Option V-Var: auf 1 gesetzt ist, was darauf hinweist, dass das erste Element in der Vektorlösung, d.h. in x' gegen die unabhängige Variable t gezeichnet werden muss.
Bestätigen Sie die Änderungen von PLOT SETUP durch Drücken von NXT OK.

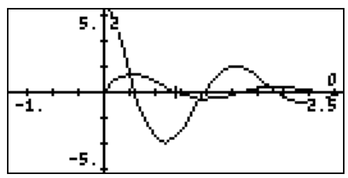
Drücken Sie dann ← WIN (gleichzeitig, wenn im RPN-Modus), um in die PLOT WINDOW—Umgebung zu gelangen. Ändern Sie die Eingabeform, damit sie wie folgt aussieht:



Um die x' gegen t-Grafik zu zeichnen, verwenden Sie: ERASE DRAW. Die Zeichnung von x' gegen t sieht wie folgt aus:



Um die zweite Kurve zu zeichnen, müssen wir das Eingabefenster PLOT SETUP nochmals verwenden. Um auf diese Form der Grafik zuzugreifen, verwenden Sie: CANCL NXT OK ← 2D/3D (gleichzeitig, wenn im RPN-Modus). Ändern Sie die Werte des Feldes V-Var: auf 2 und drücken Sie DRAW (drücken Sie nicht auf ERASE oder Sie verlieren die oben erzeugte Grafik). Verwenden Sie: EDIT NXT LABEL MENU, um Namen und Bereich der Achsen zu sehen. Beachten Sie, dass die Bezeichnung für die x-Achse die Zahl 0 ist (stellt die unabhängige Variable dar), während die Bezeichnung für die y-Achse die Zahl 2 ist (stellt die zweite Variable dar, d.h. die zuletzt gezeichnete Variable). Die kombinierte Grafik sieht wie folgt aus.



Drücken Sie NXT NXT PICT CANCL ON, um zum normalen Taschenrechnerdisplay zurückzukehren.

Numerische Lösung einer steifen ODE erster Ordnung

Wir nehmen die ODE: dy/dt = -100y+100t+101 an, der Anfangsbedingung y(0) = 1 ausgesetzt.

Exakte Lösung

Die Gleichung kann geschrieben werden als dy/dt + 100 y = 100 t + 101, und mithilfe eines integrierenden Faktors IF(t) = exp(100t) wie folgt gelöst werden (RPN-Modus mit CAS auf Exact-Modus gesetzt):

‘(100*t+101)*EXP(100*t)’ ENTER ‘t’ ENTER RISCH

Das Ergebnis lautet ‘(t+1)*EXP(100*t)’.

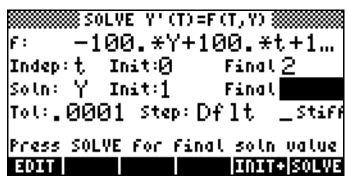
Dann fügen wir eine Integrationskonstante hinzu, mithilfe von: ‘C’ ENTER +

Dann teilen wir durch FI(x), mithilfe von: ‘EXP(100*t)’ ENTER ÷.

Das Ergebnis lautet: ‘((t+1)*EXP(100*t)+C)/EXP(100*t)’, d.h., y(t) = 1+ t +C·e^{100t}. Die Anwendung der Anfangsbedingung y(0) = 1 ergibt 1 = 1 + 0 + C·e^0, or C = 0, wobei die besondere Lösung y(t) = 1+t ist.

Numerische Lösung

Wenn wir eine direkte numerische Lösung der ursprünglichen Gleichung dy/dt = -100y+100t+101 mit dem numerischen Löser des Taschenrechners versuchen, erkennen wir, dass der Taschenrechner zur Erstellung einer Lösung länger braucht, als im Beispiel für die erste Ordnung. Um dies zu überprüfen, setzen Sie den numerischen Differentialgleichungs-Löser (→ NUM.SLV ▼ OK) auf:

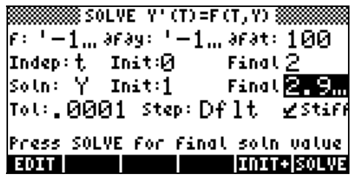


Hier versuchen wir den Wert von y(2) zu erhalten, bei y(0) = 1. Markieren Sie das Feld `Soln: Final` und drücken Sie SOLVE. Sie können feststellen, dass eine Lösung 6 Sekunden braucht, während die Lösung im Beispiel für erste Ordnung fast unmittelbar fertig war. Drücken Sie ON, um die Berechnung abzubrechen.

Dies ist ein Beispiel für eine steife gewöhnliche Differentialgleichung. Die allgemeine Lösung einer steifen ODE enthält Komponenten, die sich durch sehr verschiedene Raten bei gleicher Steigerung der unabhängigen Variable unterscheiden. In diesem speziellen Fall enthält die allgemeine Lösung y(t) = 1+ t +C·e^{100t} die Komponenten ‘t’ und ‘C·e^{100t}’, die bei sehr unterschiedlichen Raten variieren, außer in den Fällen C=0 oder C≈0 (z. B. für C = 1, t =0.1, C·e^{100t} =22026).

Der numerische ODE-Löser des Taschenrechners ermöglicht die Lösung steifer ODE, indem die Funktion `_Stiff` im Bildschirm `SOLVE Y'(T) = F(T,Y)` ausgewählt wird. Ist diese Funktion aktiviert, müssen Sie die Werte für $\partial f/\partial y$ und $\partial f/\partial t$ eingeben. Im vorliegenden Fall $\partial f/\partial y$ =-100 und $\partial f/\partial t$ = 100.

Geben Sie diese Werte in die entsprechenden Felder des Bildschirms `SOLVE Y'(T) = F(T,Y)` ein.



Danach verschieben Sie den Kursor auf das Feld `Soln:Final` und drücken Sie SOLVE. Diesmal erhalten Sie die Lösung nach ca. 1 Sekunde. Drücken Sie EDIT, um folgende Lösung anzuzeigen: 2,9999999999, d.h. 3.0.

Hinweis: Die Option `Stiff` ist auch für grafische Lösungen von Differentialgleichungen verfügbar.

Numerische Lösung von ODE mit dem Menü SOLVE/DIFF

Das Softmenü SOLVE wird aktiviert, indem 74 MENU im RPN-Modus verwendet wird. Dieses Menü wird in Kapitel 6 näher erklärt. Eines der Untermenüs, DIFF, enthält Funktionen für die numerische Lösung gewöhnlicher Differentialgleichungen für die Verwendung beim Programmieren. Diese Funktionen werden als nächstes im RPN-Modus beschrieben, wobei Systemflag 117 auf Menü SOFT gesetzt ist.

Die vom Menü SOLVE/DIFF verfügbaren Funktionen sind folgende:

Funktion RFK

Diese Funktion wird verwendet, um Lösungen eines Anfangswert-Problems für Differentialgleichungen erster Ordnung mithilfe des Runge-Kutta-Fehlbert Lösungsschemas 4^{th} -5^{th}. Ordnung zu berechnen. Angenommen die zu lösende Differentialgleichung ist gegeben durch dy/dx = f(x,y), mit y = 0 at x = 0, und Sie lassen ein Konvergenzkriterium ε für die Lösung zu. Sie können auch eine Verminderung der unabhängigen Variable Δx für die Verwendung in der Funktion spezifizieren. Um diese Funktion auszuführen, müssen Sie den Speicher wie folgt vorbereiten:

3: {'x', 'y', 'f(x,y)'}
2: { ε Δx }
1: x_{final}

Der Wert auf der ersten Speicherebene ist der Wert der unabhängigen Variable, für die Sie eine Lösung finden möchten, d.h. Sie möchten $y_{final} = f_s(x_{final})$, finden, wobei $f_s(x)$ die Lösung zur Differentialgleichung darstellt. Die zweite Speicherebene enthält möglicherweise nur den Wert von ε, und der Schritt Δx wird als kleiner Standardwert genommen. Nachdem Sie die Funktion RKF ausgeführt haben, wird im Speicher Folgendes angezeigt:

2: {'x', 'y', 'f(x,y)'}
1: ε

Der Wert der Lösung y_{final},wird in Variable y verfügbar sein. Diese Funktion ist für Programmiervorgänge geeignet, da die Spezifikationen der Differentialgleichung und die Toleranz im Speicher für neue Lösungen bereit stehen. Beachten Sie, dass die Lösung die Anfangsbedingung x = 0 bei y = 0 verwendet. Wenn Ihre wirklichen Anfangslösungen $x = x_{init}$ bei $y = y_{init}$ sind, dann können Sie diese Werte immer noch der Lösung von RKF hinzufügen, wobei Sie aber auf die folgende Beziehung achten müssen.

RKF-Lösung		Wirkliche Lösung	
x	y	x	y
0	0	x_{init}	y_{init}
x_{final}	y_{final}	$x_{init} + x_{final}$	$y_{init} + y_{final}$

Die folgenden Bildschirme zeigen den RPN-Speicher vor und nach Anwendung der Funktion RKF für die Differentialgleichung dy/dx = x+y, ε = 0,001, Δx = 0,1

Nach Anwendung der Funktion RKF enthält die Variable den Wert 4.3880.

Funktion RRK

Die Funktion ist ähnlich der RKF-Funktion, außer dass RRK (Rosenbrock und Runge-Kutta-Methoden) als Liste auf Speicherebene 3 für die Eingabe nicht nur die Namen der unabhängigen und abhängigen Variablen und die Funktion, die die Differentialgleichung definiert, benötigt, sondern auch die Ausdrücke für die ersten und zweiten Ableitungen des Ausdrucks. Somit sieht der Eingabespeicher für diese Funktion aus wie folgt:

3: {'x', 'y', 'f(x,y)' '∂f/∂x' '∂f/∂y' }
2: { ε Δx }
1: x_{final}

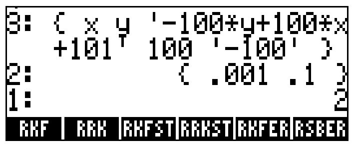
Der Wert auf der ersten Speicherebene der unabhängigen Variable für die Sie eine Lösung finden möchten, d.h. Sie möchten $y_{final} = f_s(x_{final})$, finden, wobei $f_s(x)$ die Lösung zur Differentialgleichung darstellt. Die zweite Speicherebene enthält möglicherweise nur den Wert von ε, und der Schritt Δx wird als kleiner Standardwert genommen. Nachdem Sie die Funktion , ausgeführt haben, wird im Speicher Folgendes angezeigt:

2: {'x', 'y', 'f(x,y)' '∂f/∂x' '∂f/vy' }
1: { ε Δx }

Der Wert der Lösung y_{final},wird in Variable verfügbar sein.

Diese Funktion kann verwendet werden, um sogenannte „steife" Differentialgleichungen zu lösen.

Die folgenden Screenshots zeigen den RPN-Speicher vor und nach Anwendung der Funktion RRK:



Der in der Variable y gespeicherte Wert ist 3,00000000004.

Funktion RKFSTEP

Die Eingabeliste dieser Funktion ist ähnlich wie die der Funktion RKF, auch die Toleranz für die Lösung und ein möglicher Schritt Δx sind ähnlich. Die Funktion ergibt die gleiche Eingabeliste, gefolgt von der Toleranz und einer Schätzung des nächsten Schrittes in der unabhängigen Variable. Die Funktion ergibt die Eingabeliste, Toleranz und den nächsten Schritt in der unabhängigen Variable, welche der Toleranz genügt. Somit sieht der Eingabespeicher wie folgt aus:

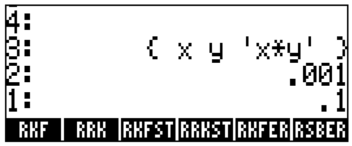
3: {'x', 'y', 'f(x,y)'}
2: ε
1: Δx

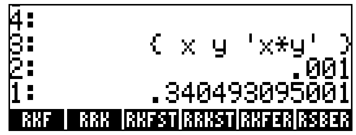
Nachdem Sie die Funktion ausgeführt haben, wird im Speicher Folgendes angezeigt:

3: {'x', 'y', 'f(x,y)'}
2: ε
1: $(\Delta x)_{next}$

Somit wird diese Funktion verwendet, um die angemessene Größe eines Zeitschrittes zu bestimmen, welcher der erforderlichen Toleranz genügt.

Die folgenden Screenshots zeigen den RPN-Speicher vor und nach Anwendung der Funktion RKFSTEP:





Die Ergebnisse zeigen an, dass $(\Delta x)_{next}$ = 0,34049….

Funktion RRKSTEP

Die Eingabeliste dieser Funktion ist ähnlich wie die der Funktion RRK, auch die Toleranz für die Lösung und ein möglicher Schritt Δx sind ähnlich und eine Zahl (LAST), welche die in der Lösung zuletzt verwendete Methode spezifiziert (1, wenn RKF verwendet wurde, oder 2, wenn RRK verwendet wurde). Die Funktion RRKSTEP ergibt dieselbe Eingabeliste, gefolgt von der Toleranz, eine Schätzung des nächsten Schrittes in der unabhängigen Variabel und die aktuelle Methode (CURRENT), die verwendet wird um zum nächsten Schritt zu gelangen. Somit sieht der Eingabespeicher wie folgt aus:

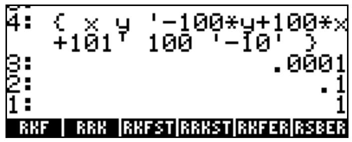
4:	{'x', 'y', 'f(x,y)'}
3:	ε
2:	Δx
1:	LAST

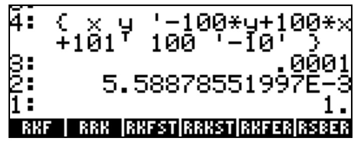
Nachdem Sie die Funktion ausgeführt haben, wird im Speicher Folgendes angezeigt:

4:	{'x', 'y', 'f(x,y)'}
3:	ε
2:	$(\Delta x)_{next}$
1:	CURRENT

Somit wird diese Funktion verwendet, um die angemessene Größe eines Zeitschrittes ($(\Delta x)_{next}$) zu bestimmen, welcher der erforderlichen Toleranz genügt und die Methode, die verwendet wurde, um zu diesem Ergebnis zu kommen (CURRENT).

Die folgenden Screenshots zeigen den RPN-Speicher vor und nach Anwendung der Funktion RRKSTEP:





Die Ergebnisse zeigen, dass $(\Delta x)_{next}$ = 0,00558 und dass die RKF-Methode (CURRENT = 1) verwendet werden sollte.

Funktion RKFERR

Diese Funktion ergibt den absoluten Fehler des Schätzverhaltens für einen bestimmten Schritt, wenn ein ähnliches Problem gelöst wird, wie auch in der Funktion RKF beschrieben. Somit sieht der Eingabespeicher wie folgt aus:

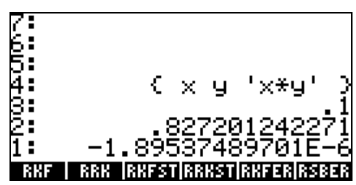
2:	{'x', 'y', 'f(x,y)'}
1:	Δx

Nachdem Sie die Funktion ausgeführt haben, wird im Speicher Folgendes angezeigt:

4:	{'x', 'y', 'f(x,y)'}
3:	ε
2:	Δy
1:	Error

Somit wird diese Funktion verwendet, um die Verminderung in der Lösung Δy, sowie den absoluten Fehler (error) zu bestimmen.

Die folgenden Screenshots zeigen den RPN-Speicher vor und nach Anwendung der Funktion RKFERR:



Die Ergebnisse zeigen, dass Δy = 0,827… und Fehler = -1,89…$\times 10^{-6}$.

Funktion RSBERR

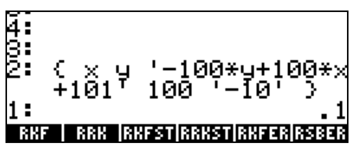
Diese Funktion funktioniert ähnlich wie die Funktion RKERR, aber mit den Eingabeelementen, die für die Funktion RRK aufgelistet sind. Somit sieht der Eingabespeicher für diese Funktion wie folgt aus:

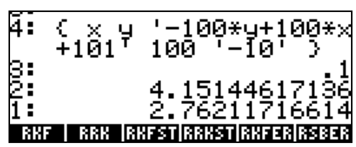
2: {'x', 'y', 'f(x,y)' '∂f/∂x' '∂f/vy' }
1: Δx

Nachdem Sie die Funktion ausgeführt haben, wird im Speicher Folgendes angezeigt:

4: {'x', 'y', 'f(x,y)' '∂f/∂x' '∂f/vy' }:
3: ε
2: Δy
1: Error

Die folgenden Screenshots zeigen den RPN-Speicher vor und nach Anwendung der Funktion RSBERR:





Die Ergebnisse zeigen, dass Δy = 4,1514… und Fehler = 2,762…, für Dx = 0,1. Überprüfen Sie, dass wenn Dx auf 0,01 vermindert ist, Δy = -0,00307… und Fehler = 0,000547.

Anmerkung: Da Sie durchführen, werden die Befehle in den DIFF Menüwerten von x und y als Variablen in Ihrem Rechner produziert und gespeichert. Die Resultate, die durch die Funktionen in diesem Abschnitt bereitgestellt werden, hängen von den gegenwärtigen Werten von x und y ab. Folglich, einige von der Resultate veranschaulicht oben können sich unterscheiden von was Sie in Ihrem Rechner erhalten.

Kapitel 17
Wahrscheinlichkeitsanwendungen

In diesem Kapitel geben wir Beispiele für Anwendungen der Rechnerfunktionen zu Wahrscheinlichkeitsverteilungen.

Das MTH/PROBABILITY.. Untermenü - Teil 1

Das Untermenü MTH/PROBABILITY.. ist über die Tastenkombination ← MTH verfügbar. Wenn das Systemflag 117 auf CHOOSE boxes gesetzt ist, erscheint die folgende MTH-Optionsliste (siehe Abb. auf der linken Seite unten). Wir haben die Option PROBABILITY.. (Option 7) ausgewählt, um die folgenden Funktionen zu zeigen (siehe Abb. auf der rechten Seite unten):

In diesem Abschnitt besprechen wir die Funktionen COMB, PERM, ! (Fakultät), RAND und RDZ.

Fakultäten, Kombinationen und Permutationen

Die Fakultät eines Integers n wird definiert als: n! = n· (n-1) · (n-2)…3·2·1. Definitionsgemäß ist 0! = 1. Fakultäten werden bei der Berechnung von Permutationszahlen und Objektkombinationen verwendet. Beispielsweise ist die Permutationszahl von r Objekten aus einer Menge mit n verschiedenen Objekten

$$_nP_r = n(n-1)(n-1)...(n-r+1) = n!/(n-r)!$$

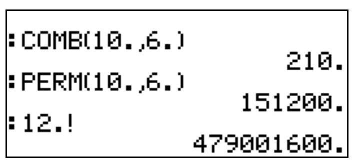
Außerdem beträgt die Anzahl der Kombinationen von n Objekten mit r Elementen auf einmal

$$\binom{n}{r} = \frac{n(n-1)(n-2)...(n-r+1)}{r!} = \frac{n!}{r!(n-r)!}$$

Um das System zu vereinfachen, verwenden Sie P(n, r) für Permutationen und C(n, r) für Kombinationen. Wir können Kombinationen, Permutationen und Fakultäten mit den Funktionen COMB, PERM und ! aus dem Untermenü MTH/PROBABILITY.. berechnen. Die Bedienung dieser Funktionen wird im Folgenden beschrieben:

- COMB(n,r): Kombinationen von n Elementen mit r Elementen auf einmal
- PERM(n,r): Permutationen von n Elementen mit r Elementen auf einmal
- n!: Fakultät eines positiven Integers. Für einen Nicht-Integer gibt x! dann Γ(x+1) zurück, wobei Γ(x) die Gamma-Funktion ist (siehe Kapitel 3). Das Fakultätssymbol (!) kann auch als Tastenkombination ALPHA ↱ 2 eingegeben werden.

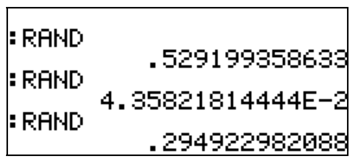
Ein Beispiel für Anwendungen dieser Funktionen wird im Folgenden gezeigt:

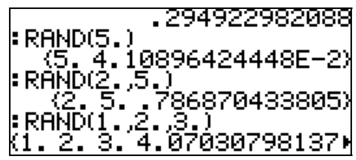


Zufallszahlen

Der Rechner bietet einen Zufallszahlengenerator, der einheitlich verteilte zufällige reelle Zahlen zwischen 0 und 1 ausgibt. Der Generator kann zufällige Zahlensequenzen ausgeben. Nach einer bestimmten Anzahl von Malen jedoch (einer wirklich hohen Anzahl) tendiert die Sequenz dazu, sich selbst zu wiederholen. Aus diesem Grund ist der Zufallszahlengenerator eher ein Pseudozufallsgenerator. Um eine Zufallszahl mit Ihrem Rechner zu erzeugen, verwenden Sie die Funktion RAND aus dem Untermenü MTH/PROBABILITY. Die folgende Anzeige zeigt eine Anzahl zufälliger Zahlen, die mit RAND erstellt wurden. Die Zahlen in der linken Abbildung sind mit der aufrufenden Funktion RAND ohne Argument erstellt worden. Wenn Sie eine

Argumentenliste in die Funktion RAND einsetzen, erhalten Sie die Zahlenliste plus eine zusätzliche Zufallszahl, die wie in der rechten Abbildung gezeigt angehängt wird.





Zufallszahlengeneratoren arbeiten im Allgemeinen so, dass sie einen Wert nehmen, der als “Ausgangszahl” des Generators bezeichnet wird, und einen mathematischen Algorithmus auf diese "Ausgangszahl" ausführen, was eine neue (pseudo)zufällige Zahl erzeugt. Wenn Sie eine Zahlensequenz erzeugen wollen und diese später wiederholen möchten, können Sie die "Ausgangszahl" des Generators durch Verwendung der Funktion RDZ(n) ändern, wobei n die “Ausgangszahl” ist, bevor Sie die Sequenz erzeugen. Zufallszahlengeneratoren beginnen mit einer "Ausgangszahl“, die in die erste Zufallszahl der Serie umgewandelt wird. Die aktuelle Zahl dient als "Ausgangszahl" für die nächste Zahl und so weiter. Durch erneutes Verwenden derselben "Ausgangszahl" für die Sequenz können Sie dieselbe Sequenz mehr als einmal erzeugen. Probieren Sie beispielsweise das Folgende aus:

RDZ(0,25) ENTER	Verwenden Sie 0,25 als "Ausgangszahl".
RAND() ENTER	Erste Zufallszahl = 0,75285…
RAND() ENTER	Zweite Zufallszahl = 0,51109…
RAND() ENTER	Dritte Zufallszahl = 0,085429….
Beginnen Sie die Sequenz erneut:	
RDZ(0,25) ENTER	Verwenden Sie 0,25 als "Ausgangszahl".
RAND() ENTER	Erste Zufallszahl = 0,75285…
RAND() ENTER	Zweite Zufallszahl = 0,51109…
RAND() ENTER	Dritte Zufallszahl = 0,085429….

Um eine Sequenz zufälliger Zahlen zu erzeugen, verwenden Sie die Funktion SEQ. Um beispielsweise eine Liste mit 5 Zufallszahlen zu erzeugen, können

Sie im ALG-Modus: SEQ(RAND(),1,5,1) verwenden. Im RPN-Modus verwenden Sie nachfolgendes Programm:

« → n « 1 n FOR j RND NEXT n →LIST » »

Speichern Sie es in der Variablen RLST (Random LiST) und verwenden Sie VAR 5 RLST, um eine Liste mit 5 Zufallszahlen zu erzeugen.

Die Funktion RNDM(n,m) kann dazu verwendet werden, um eine Matrix mit n Reihen und m Spalten zu erzeugen, deren Elemente aus zufälligen Integeren zwischen -1 und 1 bestehen (siehe Kapitel 10).

Diskrete Wahrscheinlichkeitsverteilung

Eine zufällige Variable wird als diskret bezeichnet, wenn sie nur eine begrenzte Anzahl an Werten hat. So kann beispielsweise die Anzahl der Regentage an einem bestimmten Ort als diskrete Zufallsvariable betrachtet werden, weil wir diese nur als Integerzahlen zählen. Wenn X eine diskrete Zufallsvariable darstellt, wird ihre Wahrscheinlichkeitsverteilung (pmf) durch f(x) = P[X=x] dargestellt, d.h. die Wahrscheinlichkeit, dass die Zufallsvariable X den Wert x annimmt.

Die Wahrscheinlichkeitsverteilung muss die Bedingungen erfüllen, dass

$$f(x) >0, \text{ für alle } x,$$

und

$$\sum_{all\ x} f(x) = 1.0$$

Eine kumulative Verteilungsfunktion (cdf) wird definiert als

$$F(x) = P[X \le x] = \sum_{k \le x} f(k)$$

Als Nächstes definieren wir eine Reihe von Funktionen, um diskrete Wahrscheinlichkeitsverteilungen zu berechnen. Wir schlagen vor, dass Sie ein Unterverzeichnis anlegen, etwa HOME\STATS\DFUN (Diskrete FUNktionen), in dem wir die Wahrscheinlichkeitsverteilung und die Verteilungsfunktion für binomische und Poisson-Verteilungen berechnen.

Binomische Verteilung

Die Wahrscheinlichkeitsverteilung der binomische Verteilung ist gegeben durch

$$f(n,p,x) = \binom{n}{x} \cdot p^x \cdot (1-p)^{n-x}, \quad x = 0,1,2,...,n$$

wobei $\binom{n}{x}$ = C(n,x) die Kombination von n Elementen mit x Elementen auf einmal ist. Die Werte n und p sind die Verteilungsparameter. Der Wert n stellt die Anzahl der Wiederholungen eines Experiments oder eine Beobachtung, die ein von zwei Ergebnissen haben können, z.B. Erfolg oder Misserfolg. Wenn die Zufallsvariable X die Anzahl der Erfolge in den n Wiederholungen darstellt, dann stellt p die Wahrscheinlichkeit, wie häufig ein Erfolg bei n gegebenen Wiederholungen auftreten kann, dar. Die Verteilungsfunktion für die binomische Verteilung ist gegeben durch

$$F(n,p,x) = \sum_{k=0}^{x} f(n,p,x), \quad x = 0,1,2,...,n$$

Poisson-Verteilung

Die Wahrscheinlichkeitsverteilung der Poisson-Verteilung ist gegeben durch

$$f(\lambda,x) = \frac{e^{-\lambda} \cdot \lambda^x}{x!}, \quad x = 0,1,2,...,\infty .$$

Wenn in diesem Ausdruck die Zufallsvariable X die Anzahl der Vorkommen eines Ereignisses oder einer Beobachtung pro Zeiteinheit, Länge, Fläche, Volumen usw. darstellt, dann ist der von l dargestellte Parameter die durchschnittliche Anzahl von Vorkommen pro Zeiteinheit, Länge, Fläche, Volumen usw. Die Verteilungsfunktion für die Poisson-Verteilung ist gegeben durch

$$F(\lambda, x) = \sum_{k=0}^{x} f(\lambda, x), \quad x = 0,1,2,...,\infty$$

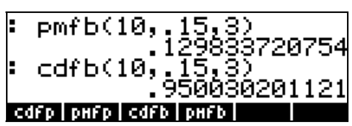
Verwenden Sie als nächstes die Funktion DEFINE (DEF), um die folgenden Wahrscheinlichkeitsverteilungen (pmf) und Verteilungsfunktionen (cdf) zu definieren:

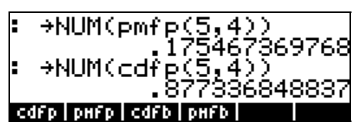
```
DEFINE(pmfb(n,p,x) = COMB(n,x)*p^x*(1-p)^(n-x))
DEFINE(cdfb(n,p,x) = Σ(k=0,x,pmfb(n,p,k)))
DEFINE(pmfp(λ,x) = EXP(-λ)*λ^x/x!)
DEFINE(cdfp(λ,x) = Σ(k=0,x,pmfp(λ,x)))
```

Die Funktionsnamen stehen für:

- pmfb: Wahrscheinlichkeitsverteilung für die binomische Verteilung
- cdfb: Verteilungsfunktion für die binomische Verteilung.
- pmfp: Wahrscheinlichkeitsverteilung für die Poisson-Verteilung
- cdfp: Verteilungsfunktion für die Poisson-Verteilung.

Beispiele für die Berechnungen, die diese Funktionen verwenden, werden im Folgenden gezeigt:





Stetige Wahrscheinlichkeitsverteilungen

Die Wahrscheinlichkeitsverteilung für eine stetige Zufallsvariable X wird durch eine Funktion f(x), die als Wahrscheinlichkeitsdichte (pdf) bekannt ist, charakterisiert. Die pdf hat die folgenden Eigenschaften: f(x) > 0, für alle x und

$$P[X < x] = F(x) = \int_{-\infty}^{x} f(\xi)d\xi.$$

$$\int_{-\infty}^{+\infty} f(x)dx = 1.$$

Wahrscheinlichkeiten werden mit der kumulativen Verteilungsfunktion (cdf), F(x), berechnet und definiert durch $P[X < x] = F(x) = \int_{-\infty}^{x} f(\xi)d\xi$, wobei P[X<x] für "die Wahrscheinlichkeit, dass die Zufallsvariable X kleiner als der Wert x ist" steht.

In diesem Abschnitt beschreiben wir mehrere stetige Wahrscheinlichkeitsverteilungen, einschließlich der Gamma-, Exponential-, Beta- und Weibull-Verteilungen. Diese Verteilungen werden in jedem Statistikbuch beschrieben. Einige dieser Verteilungen verwenden die vorhin definierte Gammafunktion, die im Rechner mit der Fakultätsfunktion als Γ(x) = (x-1)! für jede reelle Zahl x berechnet wird.

Die Gammaverteilung

Die Wahrscheinlichkeitsverteilung (pdf) für die Gammaverteilung ist gegeben durch

$$f(x) = \frac{1}{\beta^{\alpha}\Gamma(\alpha)} \cdot x^{\alpha-1} \cdot \exp(-\frac{x}{\beta}), f \qquad x > 0, \alpha > 0, \beta > 0;$$

Die entsprechende (kumulative) Verteilung (cdf) würde durch ein Integral gegeben werden, dass keine Auflösung in geschlossener Form hat.

Die Exponentialverteilung

Die Exponentialverteilung ist die Gammaverteilung mit α = 1. Ihre pdf wird gegeben durch

$$f(x) = \frac{1}{\beta} \cdot \exp(-\frac{x}{\beta}), f \qquad x > 0, \beta > 0,$$

während ihre cdf durch F(x) = 1 - exp(-x/β), für x>0, β >0 gegeben ist.

Die Betaverteilung

Die pdf für die Gammaverteilung ist gegeben durch

$$f(x) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\cdot\Gamma(\beta)} \cdot x^{\alpha-1} \cdot (1-x)^{\beta-1}, f \qquad 0 < x < 1, \alpha > 0, \beta > 0$$

Wie im Fall der Gammaverteilung ist die entsprechende cdf für die Betaverteilung auch durch ein Integral mit einer Auflösung ohne geschlossene Form gegeben.

Die Weibull-Verteilung

Die pdf für die Weibull-Verteilung ist gegeben durch

$$f(x) = \alpha \cdot \beta \cdot x^{\beta-1} \cdot \exp(-\alpha \cdot x^{\beta}), \quad f \quad x > 0, \alpha > 0, \beta > 0$$

Während die entsprechende cdf gegeben ist durch

$$F(x) = 1 - \exp(-\alpha \cdot x^{\beta}), \quad f \quad x > 0, \alpha > 0, \beta > 0$$

Funktionen für stetige Verteilungen

Um eine Funktionssammlung zu definieren, die den Gamma-, Exponential-, Beta- und Weibull-Verteilungen entsprechen, erstellen Sie zuerst ein Unterverzeichnis namens CFUN (Stetige FUNktionen) und definieren die folgenden Funktionen (wechseln Sie in den Approx-Modus):

Gamma-pdf: `'gpdf(x) = x^(α-1)*EXP(-x/β)/(β^α*GAMMA(α))'`
Gamma-cdf: `'gcdf(x) = ∫(0,x,gpdf(t),t)'`
Beta-pdf:
`'βpdf(x)= GAMMA(α+β)*x^(α-1)*(1-x)^(β-1)/(GAMMA(α)*GAMMA(β))'`
Beta-cdf: `'βcdf(x) = ∫(0,x, βpdf(t),t)'`
Exponential-pdf: `'epdf(x) = EXP(-x/β)/β'`
Exponential-cdf: `'ecdf(x) = 1 - EXP(-x/β)'`
Weibull-pdf: `'Wpdf(x) = α*β*x^(β-1)*EXP(-α*x^β)'`

Weibull-cdf: `'Wcdf(x) = 1 - EXP(-α*x^β)'`

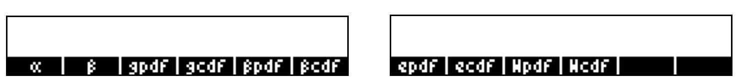
Verwenden Sie die Funktion DEFINE, um diese Funktionen zu definieren. Geben Sie als nächstes die Werte für α und β ein, z.B., 1 STO▶ ALPHA ↱ A ENTER 2 STO▶ ALPHA ↱ B ENTER

Zuletzt müssen Sie für die cdf für Gamma- und Beta-cdfs die Programmdefinitionen bearbeiten, um →NUM zu den durch die Funktion DEFINE erstellten Programme hinzuzufügen. So sollte beispielsweise die Gamma-cdf, d.h., die Funktion gcdf, verändert werden, um wie folgt angezeigt zu werden: « → x '→NUM(∫ (0,x,gpdf(t),t))' » und wieder in gcdf gespeichert werden. Wiederholen Sie diesen Vorgang für βcdf.

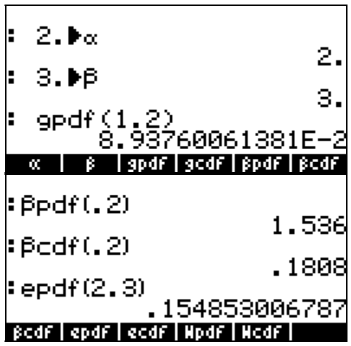
Anders als die zuvor definierten diskreten Funktionen enthalten die in diesem Abschnitt definierten stetigen Funktionen nicht ihre Parameter (α und/oder β) in ihren Definitionen. Deshalb müssen Sie diese nicht im Display eingeben, um die Funktionen zu berechnen. Diese Parameter müssen jedoch vorher definiert werden, indem die entsprechenden Werte in den Variablen α und β gespeichert werden. Wenn alle Funktionen und die Werte α und β gespeichert worden sind, können Sie die Menümarkierungen mit der Funktion ORDER zuordnen. Die Funktion wird nun wie folgt aufgerufen:

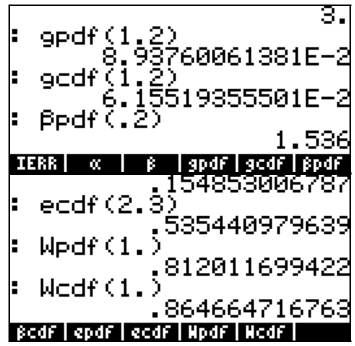
ORDER({'α','β','gpdf','gcdf','βpdf','βcdf','epdf','ecdf','Wpdf','Wcdf'})

Wenn Sie diesem Befehl folgen, zeigen die Menümarkierungen das Folgende (Drücken Sie NXT, um zur zweiten Liste zu gelangen. Drücken Sie noch einmal NXT, um wieder zur ersten Liste zu gelangen):



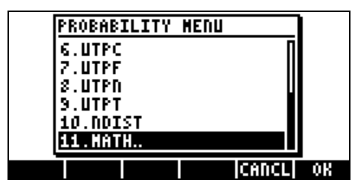
Einige Anwendungsbeispiele dieser Funktionen für die Werte α = 2, β = 3 werden unten gezeigt. Beachten Sie dabei die Variable IERR, welche in der zweiten Abbildung erscheint. Diese stammt aus einer numerischen Integration der Funktion gcdf.





Stetige Verteilungen für statistische Inferenz

In diesem Abschnitt besprechen wir vier stetige Wahrscheinlichkeitsverteilungen, die allgemein für Probleme in Zusammenhang mit statistischer Inferenz verwendet werden. Diese Verteilungen sind die normale Verteilung, die studentische t-Verteilung, die Chi-Quadrat-Verteilung (χ^2) und die F-Verteilung. Die vom Taschenrechner angebotenen Funktionen für diese Verteilungen, um Wahrscheinlichkeiten für diese Verteilungen zu bewerten, sind im Menü MTH/PROBABILITY weiter oben in diesem Kapitel enthalten. Die Funktionen sind NDIST, UTPN, UTPT, UTPC und UTPF. Ihre Anwendung wird in den folgenden Abschnitten beschrieben. Um diese Funktionen zu sehen, aktivieren Sie das Menü MTH: ← MTH und wählen die Option PROBABILITY:



Normale Verteilung pdf

Der Ausdruck für die normale Verteilung pdf lautet:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp[-\frac{(x-\mu)^2}{2\sigma^2}],$$

wobei μ das Mittel ist und σ^2 die Verteilungsvarianz. Um den Wert von $f(\mu,\sigma^2,x)$ für die normale Verteilung zu berechnen, verwenden Sie die Funktion NDIST mit den folgenden Argumenten: das Mittel μ, die Varianz σ^2 und den Wert x, d.h., NDIST(μ,σ^2,x). Überprüfen Sie das beispielsweise für eine normale Verteilung, f(1,0;0,5;2,0) = 0,20755374.

Normale Verteilung cdf

Der Taschenrechner besitzt eine Funktion UTPN, die das obere Ende der normalen Verteilung berechnet, d.h., UTPN(x) = P(X>x) = 1 - P(X<x). Um den Wert des oberen Endes der normalen Verteilung UTPN zu erhalten, müssen wir die folgenden Werte eingeben: das Mittel μ, die Varianz σ^2 und den Wert x, z.B., UTPN((μ,σ^2,x)

Überprüfen Sie das beispielsweise für eine normale Verteilung mit μ = 1,0; σ^2 = 0,5, UTPN(0,75) = 0,638163. Verwenden Sie UTPN(1,0;0,5;0,75) = 0,638163.

Verschiedene Wahrscheinlichkeitsberechnungen für normale Verteilungen [X is N(μ,σ^2)] können unter Verwendung der Funktion UTPN wie folgt definiert werden:

- P(X<a) = 1 - UTPN(μ, σ^2,a)
- P(a<X<b) = P(X<b) - P(X<a) = 1 - UTPN(μ, σ^2,b) - (1 - UTPN(μ, σ^2,a)) = UTPN(μ, σ^2,a) - UTPN(μ, σ^2,b)
- P(X>c) = UTPN(μ, σ^2,c)

Beispiele: Bei Verwendung von μ = 1,5 und σ^2 = 0,5 ergibt sich:
P(X<1,0) = 1 - P(X>1,0) = 1 - UTPN(1,5; 0,5; 1,0) = 0,239750.
P(X>2,0) = UTPN(1,5; 0,5; 2,0) = 0,239750.
P(1,0<X<2,0) = F(1,0) - F(2,0) = UTPN(1,5;0,5;1,0) - UTPN(1,5;0,5;2,0) = 0,7602499 – 0,2397500 = 0,524998.

Die studentische t-Verteilung

Die studentische t- oder einfach die t-Verteilung hat einen Parameter ν, der als der Freiheitsgrad der Verteilung bekannt ist. Die Wahrscheinlichkeitsverteilung (pdf) ist gegeben durch

$$f(t) = \frac{\Gamma(\frac{\nu+1}{2})}{\Gamma(\frac{\nu}{2}) \cdot \sqrt{\pi\nu}} \cdot (1 + \frac{t^2}{\nu})^{-\frac{\nu+1}{2}}, -\infty < t < \infty$$

wobei $\Gamma(\alpha) = (\alpha\text{-}1)!$ die im Kapitel 3 definierte GAMMA-Funktion ist.

Der Rechner sieht Werte für das obere Ende der (kumulativen) Verteilung, für die t-Verteilung, die Funktion UTPT, wenn der Parameter ν und der Wert von t gegeben sind, d.h., UTPT(ν,t), vor. Die Definition dieser Funktion ist deshalb

$$UTPT(\nu, t) = \int_t^{\infty} f(t)dt = 1 - \int_{-\infty}^{t} f(t)dt = 1 - P(T \le t)$$

Zum Beispiel ist UTPT(5;2,5) = 2,7245…E-2. Andere Wahrscheinlichkeitsrechnungen für die t-Verteilung mit der Funktion UTPT können wie folgt definiert werden:

- P(T<a) = 1 - UTPT(ν,a)
- P(a<T<b) = P(T<b) - P(T<a) = 1 - UTPT(ν,b) - (1 - UTPT(ν,a)) = UTPT(ν,a) - UTPT(ν,b)
- P(T>c) = UTPT(ν,c)

Beispiele: Gegeben ist ν = 12, bestimme:

P(T<0,5) = 1-UTPT(12;0,5) = 0,68694..
P(-0,5<T<0,5) = UTPT(12;-0,5)-UTPT(12;0,5) = 0,3738…
P(T> -1,2) = UTPT(12;-1,2) = 0,8733…

Die Chi-Quadrat-Verteilung

Die Chi-Quadrat-Verteilung (χ^2) hat einen Parameter ν, der als Freheitsgrade bekannt ist. Die Wahrscheinlichkeitsverteilung (pdf) ist gegeben durch

$$f(x) = \frac{1}{2^{\frac{\nu}{2}} \cdot \Gamma(\frac{\nu}{2})} \cdot x^{\frac{\nu}{2}-1} \cdot e^{-\frac{x}{2}}, \nu > 0, x > 0$$

Der Taschenrechner sieht Werte für das obere Ende der (kumulativen) Verteilung für die χ^2-Verteilung unter Verwendung von [UTPC] vor, wenn der Wert x und der Parameter ν gegeben sind. Die Definition dieser Funktion ist deshalb

$$UTPC(\nu, x) = \int_t^{\infty} f(x)dx = 1 - \int_{-\infty}^{t} f(x)dx = 1 - P(X \le x)$$

Zur Verwendung dieser Funktion benötigen wir die Freiheitsgrade ν und den Wert der Chi-Quadrat-Variable x, d.h. UTPC(ν,x). Zum Beispiel ist UTPC(5; 2,5) = 0,776495…

Verschiedene Wahrscheinlichkeitsberechnungen können für die Chi-Quadrat-Verteilung mit der Funktion UTPC wie folgt definiert werden:

- P(X<a) = 1 - UTPC(ν,a)
- P(a<X<b) = P(X<b) - P(X<a) = 1 - UTPC(ν,b) - (1 - UTPC(ν,a)) = UTPC(ν,a) - UTPC(ν,b)
- P(X>c) = UTPC(ν,c)

Beispiele: Gegeben ist ν = 6, bestimme:

P(X<5,32) = 1-UTPC(6;5,32) = 0,4965..
P(1,2<X<10,5) = UTPC(6;1,2)-UTPC(6;10,5) = 0,8717…
P(X> 20) = UTPC(6;20) = 2,769..E-3

Die F-Verteilung

Die F-Verteilung hat zwei Parameter νN = Zähler der Freiheitsgrade und νD = Nenner der Freiheitsgrade. Die Wahrscheinlichkeitsverteilung (pdf) ist gegeben durch

$$f(x) = \frac{\Gamma(\frac{\nu N + \nu D}{2}) \cdot (\frac{\nu N}{\nu D})^{\frac{\nu N}{2}} \cdot F^{\frac{\nu N}{2}-1}}{\Gamma(\frac{\nu N}{2}) \cdot \Gamma(\frac{\nu D}{2}) \cdot (1 - \frac{\nu N \cdot F}{\nu D})^{(\frac{\nu N + \nu D}{2})}}$$

Der Taschenrechner sieht Werte für das obere Ende der (kumulativen) Verteilung für die F-Verteilung, die Funktion UTPF, wenn die Parameter νN und νD gegeben sind, und den Wert von F vor. Die Definition dieser Funktion ist deshalb

$$UTPF(\nu N, \nu D, F) = \int_t^{\infty} f(F)dF = 1 - \int_{-\infty}^{t} f(F)dF = 1 - P(\Im \leq F)$$

Zum Beispiel, um UTPF(10;5; 2,5) = 0,161834… zu berechnen

Verschiedene Wahrscheinlichkeitsberechnungen können für die F-Verteilung mit der Funktion UTPF wie folgt definiert werden:

- P(F<a) = 1 - UTPF(νN, νD,a)
- P(a<F<b) = P(F<b) - P(F<a) = 1 -UTPF(νN, νD,b)- (1 - UTPF(νN, νD,a))
  = UTPF(νN, νD,a) - UTPF(νN, νD,b)
- P(F>c) = UTPF(νN, νD,a)

Beispiel: Gegeben ist νN = 10, νD = 5, finde:

P(F<2) = 1-UTPF(10;5;2) = 0,7700…
P(5<F<10) = UTPF(10;5;5) – UTPF(10;5;10) = 3,4693..E-2
P(F>5) = UTPF(10;5;5) = 4,4808..E-2

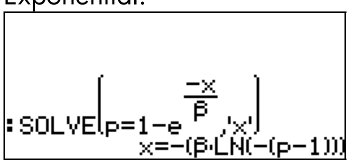
Inverse Verteilungsfunktionen

Für eine stetige Zufallsvariable X mit der kumulativen Dichtefunktion (cdf) F(x) = P(X<x) = p, müssen wir, um die inverse Verteilungsfunktion zu berechnen, den Wert von x finden, sodass x = $F^{-1}(p)$. Dieser Wert ist in den Fällen der <u>Exponential- und der Weibull-Verteilungen</u> relativ einfach zu finden, da ihre cdf's einen Ausdruck mit geschlossener Form haben:

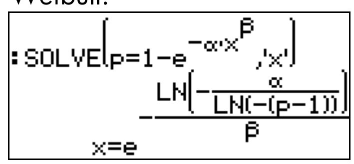
- Exponential, F(x) = 1 - exp(-x/β)
- Weibull, F(x) = 1-exp(-αx^{β})

(Bevor Fortsetzen, vergewissern Sie sich, die Variablen α und β zu löschen). Um die inversen cdf's für diese beiden Verteilungen zu finden, müssen wir nur x aus diesen beiden Ausdrücken auflösen, d.h.,

Exponential:



Weibull:



Für die <u>Gamma- und Beta-Verteilungen</u> sind die aufzulösenden Ausdrücke komplizierter aufgrund der vorhandenen Integrale, d.h.

- Gamma, $p = \int_0^x \frac{1}{\beta^{\alpha}\Gamma(\alpha)} \cdot z^{\alpha-1} \cdot \exp(-\frac{z}{\beta})dz$
- Beta, $p = \int_0^x \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\cdot\Gamma(\beta)} \cdot z^{\alpha-1} \cdot (1-z)^{\beta-1} dz$

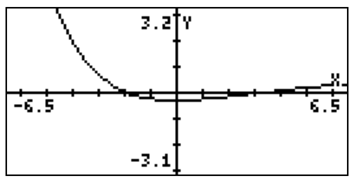
Eine numerische Auflösung mit dem numerischen Auflöser ist wegen des im Ausdruck enthaltenen Integralzeichens nicht machbar. Es ist jedoch eine grafische Lösung möglich. Details, wie Sie die Wurzel eines Graphen finden, werden in Kapitel 12 vorgestellt. Um numerische Ergebnisse zu gewährleisten, ändern Sie die Einstellung CAS auf Approx. Die zu zeichnende Funktion für die Gamma-Verteilung ist

Y(X) = ∫(0,X,z^(α-1)*exp(-z/β)/(β^α*GAMMA(α)),z)-p

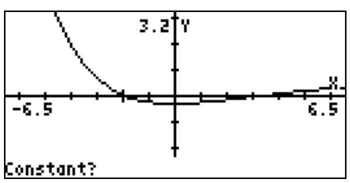
Für die Beta-Verteilung ist die zu zeichnende Funktion

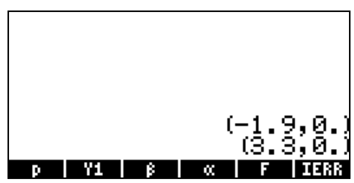
Y(X) =
∫(0,X,z^(α-1)*(1-z)^(β-1)*GAMMA(α+β)/(GAMMA(α)*GAMMA(β)),z)-p

Um den Plot zu erstellen, ist es notwendig, die Werte von α, β und p zu speichern, bevor Sie einen Plot starten. Zum Beispiel ist für α = 2, β = 3 und p = 0,3 der Plot von Y(X) für die Gamma-Verteilung der folgende (Beachten Sie bitte, dass wegen der komplizierten Natur der Funktion Y(X), einige Zeit benötigt wird, bevor der Graph erzeugt wird. Bleiben Sie geduldig).

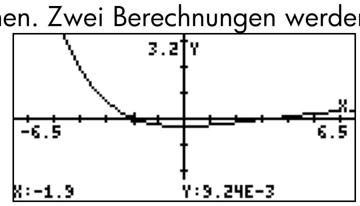


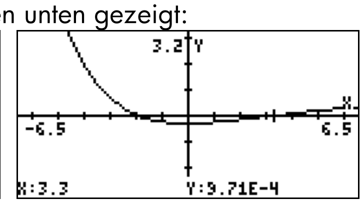
Es gibt zwei Wurzeln für diese Funktion bei Verwendung der Funktion ROOT innerhalb der Plotumgebung. Wegen des Integrals in der Gleichung wird die Wurzel angenähert und nicht in der Plotanzeige gezeigt. Sie erhalten auf der Anzeige nur die Nachricht Constant?. Wenn Sie aber an diesem Punkt ENTER drücken, wird die Annäherungswurzel in der Anzeige aufgeführt. Es werden in der rechten Abbildung unten zwei Wurzeln gezeigt.



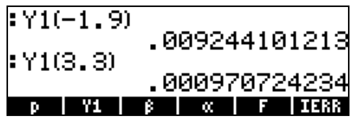


Alternativ können Sie die Funktion TRACE (X,Y) verwenden, um die Wurzeln durch Verfolgen der Kurve nahe ihrer Schnittpunkte mit der x-achse zu berechnen. Zwei Berechnungen werden unten gezeigt:





Diese Berechnungen schlagen die Lösungen x = -1,9 und x = 3,3 vor. Sie können diese “Lösungen” durch Auswertung der Funktion Y1(X) für X = -1,9 und X = 3,3 überprüfen, d.h.,

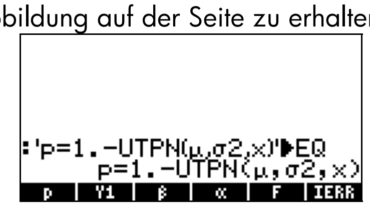


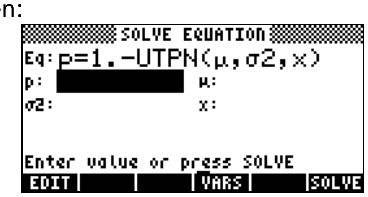
Für die Normal-, studentischen t-, Chi-Quadrat- (χ^2) und F-Verteilungen, die durch die Funktionen UTPN, UTPT, UPTC und UTPF im Taschenrechner dargestellt werden, finden Sie die inverse Antwort durch Lösung einer der folgenden Gleichungen:

- Normal, p = 1 – UTPN(μ,σ2,x)
- studentische t, p = 1 – UTPT(ν,t)
- Chi-Quadrat, p = 1 – UTPC(ν,x)
- F-Verteilung: p = 1 – UTPF(νN,νD,F)

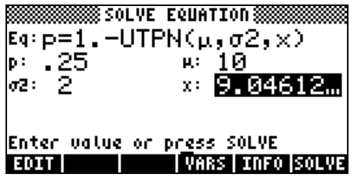
Wir weisen darauf hin, dass der zweite Parameter in der UTPN-Funktion σ2 ist, nicht σ^2, und die Verteilungsvarianz darstellt. Auch ist das Symbol ν (der kleingeschriebene griechische Buchstabe no) im Taschenrechner nicht verfügbar. Sie können beispielsweise das γ (Gamma) statt des ν verwenden. Der Buchstabe γ ist über den Zeichensatz (↱ CHARS) verfügbar.

Um beispielsweise den Wert von x für eine Normalverteilung mit μ = 10, σ^2 = 2, mit p = 0,25 zu erhalten, speichern Sie die Gleichung ‘p=1−UTPN(μ,σ2,x)’ in der Variablen EQ (Abbildung auf der linken Seite unten). Dann starten Sie den numerischen Löser, um das Eingabefeld in der Abbildung auf der Seite zu erhalten:



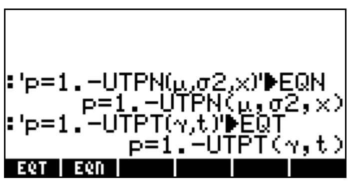


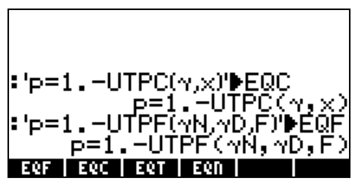
Der nächste Schritt besteht in der Eingabe der Werte von μ, σ^2 und p und der Auflösung für x:



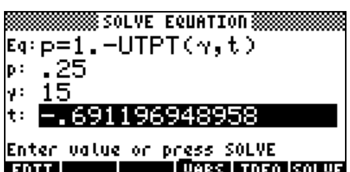
Dieses Eingabefeld kann zur Auflösung jeder der vier Variablen, die in der Gleichung für die Normalverteilung vorkommen, genutzt werden.

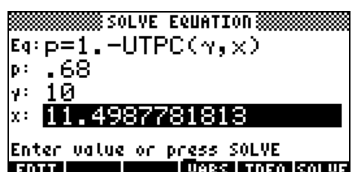
Um die Auflösung der Gleichungen mit den Funktionen UTPN, UTPT, UTPC und UTPF zu erleichtern, können Sie ein Unterverzeichnis UTPEQ anlegen, in das Sie die oben aufgelisteten Gleichungen abspeichern:

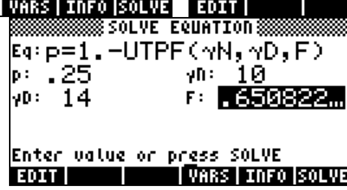




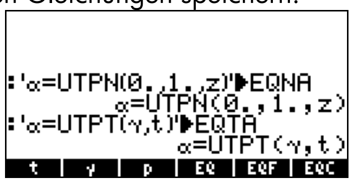
Daher haben Sie an diesem Punkt vier Gleichungen zur Auflösung zur Verfügung. Sie müssen nur eine der Gleichungen in das EQ-Feld in den numerischen Löser laden und mit der Lösung einer der variablen fortfahren. Beispiele für UTPT, UTPC und UPTF werden unten gezeigt:

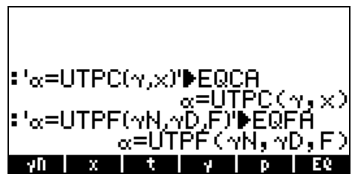




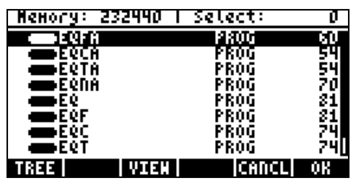


Wir weisen darauf hin, dass wir in allen oben gezeigten Beispielen mit p = P(X<x) arbeiten. Bei vielen statistischen Inferenzproblemen versuchen wir tatsächlich, den Wert von x zu finden, für den P(X>x) = α. Weiterhin arbeiten wir am ehesten für die Normalverteilung mit Standardnormalverteilung, in welcher μ =0 und σ^2 = 1. Auf die Standardnormalvariable wird normalerweise als Z Bezug genommen, sodass das zu lösende Problem P(Z>z) = α ist. Für diese Fälle statistischer Inferenzprobleme können wir die folgenden Gleichungen speichern:

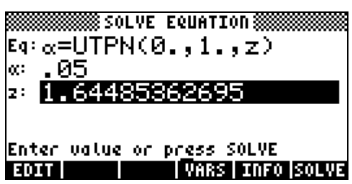


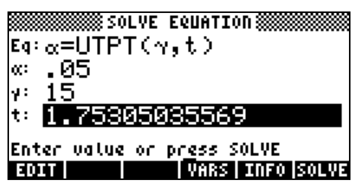


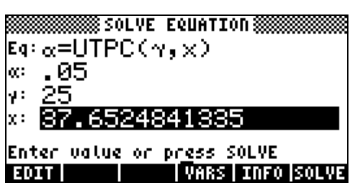
Mit diesen vier Gleichungen haben Sie, immer wenn Sie den numerischen Löser starten, die folgenden Auswahlmöglichkeiten:

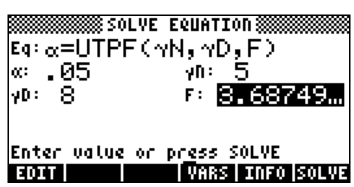


Beispiele für Lösungen der Gleichungen EQNA, EQTA, EQCA und EQFA werden unten gezeigt:







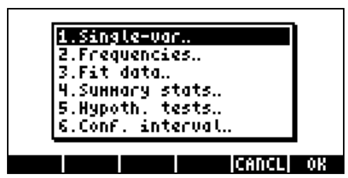


Kapitel 18
Statistikanwendungen

In diesem Kapitel werden statistische Anwendungen des Taschenrechners vorgestellt, z. B. Stichprobenkenngrößen, die Häufigkeitsverteilung von Daten, einfache Regression, Vertrauensbereiche und Hypothesentests.

Vorprogrammierte Statistikfunktionen

Der Taschenrechner enthält vorprogrammierte Statistikfunktionen, auf die über die Tastenkombination ↱ STAT zugegriffen werden kann (mit der Taste für die Zahl 5 identisch). Folgende Statistikanwendungen können mit dem Taschenrechner aufgerufen werden:



Die Anwendungen werden in diesem Kapitel ausführlich dargestellt. Zuerst jedoch zeigen wir, wie Daten für die statistische Analyse eingegeben werden.

Eingeben von Daten

Für die Analyse eines einzelnen Satzes von Daten (Stichprobe) können wir die Anwendungen 1, 2 und 4 in obiger Liste verwenden. Für alle diese Anwendungen ist es erforderlich, dass die Daten als Spalten der Matrix ΣDAT verfügbar sind. Sie können hierzu die Daten mithilfe von MatrixWriter (↰ MTRW) in Spalten eingeben.

Diese Vorgehensweise ist bei einer großen Anzahl von Datenpunkten möglicherweise ermüdend. Es bietet sich an, stattdessen die Daten als Liste (siehe Kapitel 8) einzugeben und die Liste mithilfe des Programms CRMC in einen Spaltenvektor zu konvertieren (siehe Kapitel 10). Stattdessen können Sie auch das folgende Programm eingeben, um eine Liste in einen Spaltenvektor zu konvertieren. Geben Sie das Programm im RPN-Modus ein:

« OBJ→ 1 2 →LIST →ARRY »

Speichern Sie das Programm in einer Variablen mit der Bezeichnung LCX. Nachdem Sie das Programm im RPN-Modus gespeichert haben, können Sie es auch im ALG-Modus verwenden.

Um einen Spaltenvektor in der Variablen ΣDAT zu speichern, verwenden Sie die Funktion STOΣ, die über den Katalog (‚ _CAT) verfügbar ist, z. B. STOΣ (ANS(1)) im ALG-Modus.

Beispiel 1 – Erstellen Sie mithilfe des oben definierten Programms LXC einen Spaltenvektor mit den folgenden Daten: 2,1 1,2 3,1 4,5 2,3 1,1 2,3 1,5 1,6 2,2 1,2 2,5,

Geben Sie im RPG-Modus die Daten in eine Liste ein:

{2,1 1,2 3,1 4,5 2,3 1,1 2,3 1,5 1,6 2,2 1,2 2,5 } ENTER LXC

Speichern Sie die Daten mithilfe der Funktion STOΣ in ΣDAT.

Berechnen von Kenngrößen mit einer einzigen Variablen

Es wird vorausgesetzt, dass der einzelne Datensatz als Spaltenvektor in der Variablen ΣDAT gespeichert wurde. Drücken Sie ‚ _STAT , um auf die einzelnen Programme von STAT zuzugreifen. Drücken Sie OK, um **1. Single-var..** auszuwählen. Anschließend ist eine Eingabemaske mit der Beschriftung **SINGLE-VARIABLE STATISTICS** verfügbar, und die derzeit in der Variablen ΣDAT vorhandenen Daten sind in der Maske als Vektor aufgelistet. Da nur eine Spalte vorhanden ist, muss vor dem Feld **Col:** der Wert 1 stehen. Das Feld **Type** bestimmt, ob Sie mit einer Stichprobe oder einer Grundgesamtheit arbeiten. Die Standardeinstellung ist „Stichprobe". Bewegen Sie den Cursor an die horizontale Linie vor den Feldern **Mean**, **Std Dev**, **Variance**, **Total**, **Maximum** und **Minimum**, und drücken Sie die Menütaste ✓CHK, um die Werte auszuwählen, die von diesem Programm ausgegeben werden sollen. Drücken Sie anschließend OK. Die ausgewählten Werte werden mit der entsprechenden Beschriftung auf dem Bildschirm des Taschenrechners aufgelistet.

Beispiel 1 – Für die im vorherigen Beispiel gespeicherten Daten lauten die Ergebnisse der Kenngröße mit einer einzigen Variablen wie folgt:

Mean: 2,133;	Std Dev: 0,964;	Variance: 0,929
Total: 25,6;	Maximum: 4,5;	Minimum: 1,1

Definitionen

Für diese Größen gelten folgende Definitionen:

Angenommen es sind mehrere Datenpunkte x_1, x_2, x_3, … vorhanden, die unterschiedliche Werte derselben diskreten oder kontinuierlichen Variablen X darstellen. Die Menge aller möglichen Werte der Größe x wird als Grundgesamtheit von x bezeichnet. Eine endliche Grundgesamtheit enthält nur eine bestimmte Anzahl der Elemente x_i. Wenn die Größe x den Wert einer kontinuierlichen Größe darstellt und daher theoretisch eine unendliche Anzahl von Werten annehmen kann, ist die Grundgesamtheit in diesem Fall unendlich. Wenn Sie eine Teilmenge einer Grundgesamtheit auswählen, die durch n Datenwerte $\{x_1, x_2, \ldots, x_n\}$, dargestellt wird, haben Sie eine Stichprobe der Werte von x ausgewählt.

Stichproben sind durch eine Anzahl von Werten oder Kenngrößen gekennzeichnet. Es gibt Lagemaßzahlen, z. B. Mittelwerte, Medianwerte und häufigste Werte, sowie Streuungswerte, z. B. Wertebereich, Varianz und Standardabweichung.

Lagemaßzahlen

Der Mittelwert (bzw. das arithmetische Mittel) $\bar{x}$ der Stichprobe ist als Durchschnittswert der Elemente der Stichprobe definiert:

$$\bar{x} = \frac{1}{n} \cdot \sum_{i=1}^{n} x_i .$$

Dieser durch obige Gleichung ermittelte, mit `Total` beschriftete Wert stellt die Summe der Werte von x oder Σx_i = n· $\bar{x}$ dar. Dies ist der Wert, den der Taschenrechner unter der Überschrift `Mean` ausgibt. Andere in bestimmten Anwendungen verwendete Werte sind das geometrische Mittel x_g bzw. das harmonische Mittel x_h, die wie folgt definiert sind:

$$x_g = \sqrt[n]{x_1 \cdot x_2 \cdots x_n}, \qquad \frac{1}{x_h} = \sum_{i=1}^{n} \frac{1}{x_i}.$$

Beispiele für die Berechnung dieser Werte mithilfe von Listen finden Sie in Kapitel 8.

Der Median ist der Wert, der den Datensatz in der Mitte teilt, wenn die Elemente in aufsteigender Reihenfolge angeordnet sind. Bei einer ungeraden Zahl n von geordneten Elementen ist der Medianwert dieser Stichprobe der Wert an Position (n+1)/2. Bei einer geraden Zahl n von Elementen ist der Medianwert der Durchschnittswert der Elemente an den Positionen n/2 und (n+1)/2. Obwohl die vorprogrammierten Statistikfunktionen des Taschenrechners nicht die Berechnung des Medianwertes umfassen, kann auf sehr einfache Weise ein Programm geschrieben werden, mit dem unter Verwendung von Listen dieser Wert berechnet wird. Wenn Sie beispielsweise den Medianwert mithilfe der Daten in ΣDAT ermitteln möchten, geben Sie im RPN-Modus folgendes Programm ein (Weitere Informationen über das Programmieren in der Sprache User RPL finden Sie in Kapitel 21.):

« → nC « RCLΣ DUP SIZE 2 GET IF 1 > THEN nC COL– SWAP DROP OBJ→ 1 + →ARRY END OBJ→ OBJ→ DROP DROP DUP → n « →LIST SORT IF 'n MOD 2 == 0' THEN DUP 'n/2' EVAL GET SWAP '(n+1)/2' EVAL GET + 2 / ELSE '(n+1)/2' EVAL GET END "Median" →TAG » » »

Speichern Sie dieses Programm unter dem Namen MED. Es folgt ein Beispiel für die Anwendung dieses Programms.

Beispiel 2 – Um das Programm auszuführen, müssen Sie zunächst die Matrix ΣDAT vorbereiten. Geben Sie anschließend die Nummer der Spalte in ΣDAT ein, deren Medianwert Sie ermitteln möchten, und drücken Sie MED. Verwenden Sie für die derzeit in ΣDAT vorhandenen Daten (die in einem vorherigen Beispiel eingegeben wurden) das Programm MED, um anzuzeigen, dass `Median: 2.15`.

Der häufigste Wert einer Stichprobe wird besser durch Histogramme bestimmt, daher wird er in einem späteren Abschnitt definiert.

Werte der Streubreite

Die Varianz (Var) der Stichprobe ist definiert als $s_x^2 = \frac{1}{n-1} \cdot \sum_{i=1}^{n} (x_i - \bar{x})^2$.

Die Standardabweichung (St Dev) der Stichprobe ist einfach die Quadratwurzel der Varianz, d. h. s_x.

Der Wertebereich der Stichprobe ist die Differenz zwischen Maximal- und Minimalwerten der Stichprobe. Da der Taschenrechner mit den vorprogrammierten Statistikfunktionen die Maximal- und Minimalwerte der Stichprobe ausgibt, können Sie den Wertebereich auf einfache Weise berechnen.

Variationskoeffizient

Beim Variationskoeffizienten einer Stichprobe ist der Mittelwert, eine Lagemaßzahl, mit der Standardabweichung, einem Wert der Streubreite, kombiniert und als Prozentzahl definiert durch: $V_x = (s_x/ \bar{x})100$.

Stichprobe und Grundgesamtheit

Die oben verwendeten vorprogrammierten Funktionen für Kenngrößen mit einer einzigen Variablen können auf eine endliche Grundgesamtheit angewendet werden, indem auf dem Bildschirm `SINGLE-VARIABLE STATISTICS` die Option `Type: Grundgesamtheit` ausgewählt wird. Der

Hauptunterschied besteht in den Werten der Varianz und der Standardabweichung, die berechnet werden, indem im Nenner der Varianz n und nicht (n-1) verwendet wird.

Beispiel 3 – Wenn Sie Beispiel 1 dieses Abschnitts wiederholt ausführen und als `Type` nicht `Sample`, sondern `Grundgesamtheit` verwenden, erhalten Sie für Mittelwert, Gesamtwert, Maximum und Minimum dieselben Werte. Für Varianz und Standardabweichung sind jedoch folgende Werte gegeben: Varianz: 0,852; Standardabweichung: 0,923.

Erhalten von Häufigkeitsverteilungen

Die Anwendung **2. Frequencies** im Menü STAT kann zum Erhalten von Häufigkeitsverteilungen für einen Satz von Daten verwendet werden. Die Daten müssen wieder als Spaltenvektor verfügbar sein, der in der Variablen ΣDAT gespeichert ist. Drücken Sie zu Beginn ⮣ STAT ▼ OK. Die anschließend angezeigte Eingabemaske enthält die folgenden Felder:

Σ**DAT**:	die Matrix mit den betreffenden Daten.
Col:	die zu beachtende Spalte von ΣDAT.
X-Min:	die untere Klassengrenze (Standardwert = -6,5).
Bin Count:	die Anzahl der Klassen (Standardwert = 13).
Bin Width:	die einheitliche Breite jeder Klasse (Standardwert = 1).

Definitionen

Folgende Definitionen erleichtern das Verständnis der Bedeutung dieser Parameter: Wenn eine Menge von n Datenwerten $\{x_1, x_2, \ldots, x_n\}$ gegeben ist, die in keiner bestimmten Reihenfolge aufgelistet sind, ist es häufig erforderlich, diese Daten in einer Reihe von Klassen zu gruppieren, indem die Häufigkeit oder Anzahl der Werte jeder Klasse gezählt wird. (Hinweis: Im Taschenrechner werden Klassen als „bins" = „Kästen" bezeichnet.)

Angenommen die Klassen werden ermittelt, indem das Intervall (x_{bot}, x_{top}) in k = Bin Count Klassen unterteilt wird. Dies geschieht durch die Auswahl einer Anzahl von Klassengrenzen, also $\{xB_1, xB_2, \ldots, xB_{k+1}\}$, sodass Klasse 1 durch

xB_1-xB_2 begrenzt ist, Klasse 2 durch xB_2- xB_3 usw. Die letzte Klasse, Klasse k, ist durch xB_k - xB_{k+1} begrenzt.

Der der Mitte jeder Klasse entsprechende Wert x wird als Klassenmittelpunkt bezeichnet und ist für i = 1, 2, …, k durch $xM_i = (xB_i + xB_{i+1})/2$ definiert.

Wenn die Klassen so gewählt werden, dass die Klassengröße identisch ist, können wir die Klassengröße als Bin Width = $\Delta x = (x_{max} - x_{min}) / k$ definieren,

und die Klassengrenzen können mit $xB_i = x_{bot} + (i - 1) * \Delta x$ berechnet werden.

Jeder Datenpunkt x_j, j = 1, 2, …, n gehört zur i-ten Klasse, wenn $xB_i \leq x_j < xB_{i+1}$.

Durch die Anwendung **2. Frequencies..** im Menü STAT wird eine Häufigkeitszählung durchgeführt, und die Werte, die möglicherweise unter dem Minimum oder über dem Maximum der Klassengrenzen (d. h. die Ausreißer) liegen, werden protokolliert.

Beispiel 1 – Um das Ermitteln von Häufigkeitsverteilungen besser veranschaulichen zu können, möchten wir einen relativ umfangreichen Datensatz von 200 Datenpunkten erzeugen, indem wir wie folgt vorgehen:

- Zunächst aktivieren wir den Zufallszahlengenerator durch Verwendung von RDZ(25) im ALG-Modus oder 25 ENTER RDZ im RPN-Modus (siehe Kapitel 17).
- Geben Sie das folgende Programm im RPN-Modus ein:
  « → n « 1 n FOR j RAND 100 * 2 RND NEXT n →LIST » »,
  und speichern Sie es unter dem Namen RDLIST (RanDom number LIST generator, Zufallszahlenlistengenerator).
- Generieren Sie die Liste von 200 Zahlen durch Verwendung von RDLIST(200) im ALG-Modus, oder durch Eingabe von 200 ENTER RDLIST im RPN-Modus.
- Verwenden Sie das Programm LXC (siehe oben), um die auf diese Weise generierte Liste in einen Spaltenvektor zu konvertieren.
- Speichern Sie mithilfe der Funktion STOΣ den Spaltenvektor in ΣDAT.

- Ermitteln Sie über ⊡ STAT OK Informationen zu den einzelnen Variablen. Verwenden Sie als Typ des Datensatzes Sample, und wählen Sie für die Ergebnisse alle Optionen aus. Die Ergebnisse für dieses Beispiel lauten:

Mean: 51,0406, Standardabweichung: 29,5893…, Variance: 875,529…
Total: 10208,12, Maximum: 99,35, Minimum: 0,13

Dies bedeutet, dass die Daten im Bereich von Werten nahe Null bis Werten nahe 100 liegen. Bei Verwendung ganzer Zahlen können wir den Bereich der Abweichung der Daten als (0,100) festlegen. Zum Erzeugen einer Häufigkeitsverteilung verwenden wir das Intervall (10,90) und unterteilen es in 8 Klassen mit der Breite 10.

- Wählen Sie das Programm **2. Frequencies..** aus, indem Sie ⊡ STAT ▼ OK drücken. Die Daten sind bereits in ΣDAT vorhanden, und die Option Col muss den Wert 1 aufweisen, da ΣDAT nur eine einzige Spalte enthält.
- Ändern Sie X-Min in 10, Bin Count in 8 und Bin Width in 10, und drücken Sie OK.

Im RPN-Modus werden die Ergebnisse im Stack als Spaltenvektor auf Ebene 2 des Stacks angezeigt und als Zeilenvektor mit zwei Komponenten auf Ebene 1 des Stacks. Der Vektor auf Ebene 1 des Stacks stellt die Anzahl der Ausreißer außerhalb des Intervalls dar, für das die Häufigkeitszählung ausgeführt wurde. In diesem Fall erhalten Sie die Werte [25. 22.], die angeben, dass im ΣDAT-Vektor 25 Werte kleiner als 10 und 22 Werte größer als 90 vorhanden sind.

- Drücken Sie ◄, um die Vektorausreißer aus dem Stack zu entfernen. Das verbleibende Ergebnis ist die Häufigkeit der Daten. Dieses kann wie unten gezeigt in eine Tabelle übertragen werden.

Die Tabelle wurde anhand der Informationen erstellt, die wir zum Generieren der Häufigkeitsverteilung bereitgestellt hatten, obwohl die einzige vom Taschenrechner ausgegebene Spalte die Spalte f_i (Frequency, Häufigkeit) ist. Die Klassennummern und Klassengrenzen können bei Klassen einheitlicher Größe einfach berechnet werden, und der Klassenmittelpunkt ist einfach der

Durchschnittswert der Klassengrenzen für jede Klasse. Schließlich wird die Summenhäufigkeit ermittelt, indem zu jedem Wert in der letzten Spalte außer dem ersten Wert die Häufigkeit in der nächsten Zeile addiert und das Ergebnis in der letzten Spalte der nächsten Zeile ersetzt wird. Somit ist die Summenhäufigkeit für die zweite Klasse 18+15 = 33, während die Summenhäufigkeit für die dritte Klasse 33+16 = 49 ist usw. Die Summenhäufigkeit stellt die Häufigkeit der Zahlen dar, die kleiner oder gleich der oberen Grenze einer beliebigen Klasse sind.

Klassen-Nr.	Klasse Grenze		Klassen-mittelpunkt	Häufigkeit	Summen-
i	XB_i	XB_{i+1}	Xm_i	f_i	Häufigkeit
$< XB_1$	Aus-reißer	unter Wertebereich		25	
1	10	20	15	18	18
2	20	30	25	14	32
3	30	40	35	17	49
4	40	50	45	17	66
5	50	60	55	22	88
6	60	70	65	22	110
7	70	80	75	24	134
k = 8	80	90	85	19	153
$>XB_k$	Aus-reißer	über Wertebereich		22	

Wenn ein vom Taschenrechner generierter (Spalten-)Vektor der Häufigkeiten vorhanden ist, können Sie den Vektor der Summenhäufigkeit ermitteln, indem Sie im RPN-Modus folgendes Programm verwenden:

« DUP SIZE 1 GET → freq k « {k 1} 0 CON → cfreq « 'freq(1,1)' EVAL 'cfreq(1,1)' STO 2 k FOR j 'cfreq(j-1,1) +freq(j,1)' EVAL 'cfreq (j,1)' STO NEXT cfreq » » »

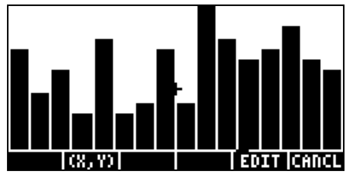
Speichern Sie das Programm unter dem Namen CFREQ. Verwenden Sie das Programm zum Generieren einer Liste von Summenhäufigkeiten (drücken Sie CFREQ, wenn der Spaltenvektor der Häufigkeiten im Stack vorhanden ist). Das Ergebnis für dieses Beispiel ist ein Spaltenvektor, der die letzte Spalte der obigen Tabelle darstellt.

Histogramme

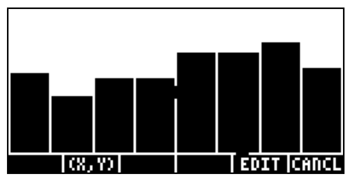
Ein Histogramm ist ein Balkendiagramm, in dem die Häufigkeit als Höhe der Balken und die Klassengrenzen als Sockel der Balken dargestellt werden. Wenn die Ursprungsdaten (d. h. die ursprünglichen Daten vor Ausführung der Häufigkeitszählung) in der Variablen ΣDAT vorhanden sind, können Sie als Diagrammtyp `Histogram` auswählen und den ursprünglichen Wert von x, die Anzahl der Klassen und die Klassenbreite angeben, um das Histogramm zu generieren. Sie können stattdessen auch wie im obigen Beispiel den Spaltenvektor generieren, der die Häufigkeitszählung enthält, diesen Vektor in ΣDAT speichern und als Diagrammtyp `Barplot` auswählen. Im nächsten Beispiel wird gezeigt, wie die erste Methode zum Generieren eines Histogramms verwendet wird.

Beispiel 1 – Generieren Sie unter Verwendung der im obigen Beispiel erzeugten 200 Datenpunkte (in ΣDAT als Spaltenvektor gespeichert) ein Histogramm der Daten, indem Sie X-Min = 10, Bin Count = 16 und Bin Width = 5 verwenden.

- Drücken Sie zunächst ← 2D/3D (gleichzeitig, sofern im RPN-Modus), um die Eingabe im Fenster PLOT SETUP zu aktivieren. Ändern Sie in diesem Fenster Type: in Histogramm, und stellen Sie sicher, dass die Option Col: 1 ausgewählt ist. Drücken Sie anschließend NXT OK.
- Drücken Sie dann ← WIN (gleichzeitig, sofern im RPN-Modus), um die Eingabe im Fenster PLOT WINDOW – HISTOGRAM zu aktivieren. Ändern Sie die Informationen in diesem Fenster in H-View: 10 90, V-View: 0 15, Bar Width: 5.
- Drücken Sie ERASE DRAW, um das folgende Histogramm zu generieren:



- Drücken Sie CANCEL, um zum vorherigen Fenster zurückzukehren. Ändern Sie die Werte für V-View und Bar Width erneut, sodass diese nun wie folgt lauten: V-View: 0 30, Bar Width: 10. Das neue Histogramm, das auf demselben Datensatz beruht, wird nun wie folgt dargestellt:



Die Darstellung der Häufigkeit f_i gegen die Klassenmittelpunkte xM_i wird als Häufigkeitspolygon bezeichnet. Die Darstellung der Summenhäufigkeit gegen die oberen Grenzen wird als Häufigkeitsverteilungskurve der Summenhäufigkeit bezeichnet. Sie können Punktwolken erzeugen, die diese beiden Darstellungen simulieren, indem Sie die entsprechenden Daten in die Spalten 1 und 2 einer neuen ΣDAT-Matrix eingeben und im Fenster PLOT SETUP `Type`: in `SCATTER` ändern.

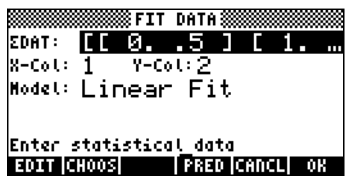
Anpassen von Daten an die Funktion y = f(x)

Mit dem Programm **3. Fit data..**, das im Menü STAT als Option 3 verfügbar ist, können lineare, logarithmische, exponentielle und Potenzfunktionen an Datensätze (x,y) angepasst werden, die in Spalten der ΣDAT-Matrix gespeichert sind. Damit dieses Programm effektiv eingesetzt werden kann, müssen in der Variablen ΣDAT mindestens zwei Spalten vorhanden sein.

Beispiel 1 – Anpassen einer linearen Funktion an die Daten in der folgenden Tabelle:

x	0	1	2	3	4	5
y	0.5	2.3	3.6	6.7	7.2	11

- Geben Sie zunächst die Daten in den beiden Zeilen in die Spalten der Variablen ΣDAT ein, indem Sie MatrixWriter und die Funktion STOΣ verwenden.
- Verwenden Sie zum Aufrufen des Programms **3. Fit data..** die folgende Tastenkombination: ↱ STAT ▼ ▼ OK. In der Eingabemaske wird die aktuelle Variable ΣDAT angezeigt, die bereits geladen ist. Legen Sie ggf. für eine lineare Anpassung im Einrichtungsfenster die folgenden Parameter fest:



- Drücken Sie OK, um die Datenanpassung auszuführen. Die unten für unseren Datensatz dargestellte Ausgabe dieses Programms besteht im RPN-Modus aus den folgenden drei Zeilen:

3: '0,195238095238 + 2,00857142857*X'
2: Correlation: 0,983781424465
1: Covariance: 7,03

Auf Ebene 3 wird die Form der Gleichung dargestellt, in diesem Fall y = 0.06924 + 0.00383 x. Auf Ebene 2 wird der Stichprobenkorrelationskoeffizient der Stichprobe dargestellt, und auf Ebene 1 die Kovarianz von x-y.

Definitionen

Für eine Stichprobe von Datenpunkten (x,y) definieren wir die Stichprobenkovarianz als

$$s_{xy} = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})$$

Der Stichprobenkorrelationskoeffizient für x,y wird definiert als

$$r_{xy} = \frac{s_{xy}}{s_x \cdot s_y}.$$

Hierbei stellen s_x, s_y die Standardabweichungen von x bzw. y dar, d. h.

$$s_x^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \overline{x})^2 \qquad s_y^2 = \frac{1}{n-1}\sum_{i=1}^{n}(y_i - \overline{y})^2$$

Bei den Werten s_{xy} und r_{xy} handelt es sich um die Werte für „Covariance" bzw. „Correlation", die mit der Funktion „Fit data" des Taschenrechners ermittelt wurden.

Linearisierte Funktionen

Zahlreiche gekrümmte Funktionen können zu einer linearen Form abgeflacht werden. Beispielsweise können die durch den Taschenrechner bereitgestellten unterschiedlichen Modelle für die Datenanpassung wie in der folgenden Tabelle dargestellt linearisiert werden.

Art der Anpas-sung	Tatsächliches Modell	Linearisiertes Modell	Unabh. Variable ξ	Abh. Variable η	Kovar. $s_{\xi\eta}$
Linear	y = a + bx	[dito]	x	y	s_{xy}
Log.	y = a + b ln(x)	[dito]	ln(x)	y	$s_{\ln(x),y}$
Exp.	$y = a e^{bx}$	ln(y) = ln(a) + bx	x	ln(y)	$s_{x,\ln(y)}$
Potenz.	$y = a x^b$	ln(y) = ln(a) + b ln(x)	ln(x)	ln(y)	$s_{\ln(x),\ln(y)}$

Die Stichprobenkovarianz von ξ,η ist durch $s_{\xi\eta} = \frac{1}{n-1}\sum(\xi_i - \overline{\xi})(\eta_i - \overline{\eta})$

definiert.

Wir definieren außerdem die Stichprobenvarianz von ξ bzw. η als

$$s_{\xi}^2 = \frac{1}{n-1}\sum_{i=1}^{n}(\xi_i - \overline{\xi})^2 \qquad s_{\eta}^2 = \frac{1}{n-1}\sum_{i=1}^{n}(\eta_i - \overline{\eta})^2$$

Der Stichprobenkorrelationskoeffizient $r_{\xi\eta}$ lautet $r_{\xi\eta} = \frac{s_{\xi\eta}}{s_{\xi} \cdot s_{\eta}}$.

Die allgemeine Form der Regressionsgleichung lautet η = A + Bξ.

Optimale Datenanpassung

Der Taschenrechner kann bestimmen, welche der linearen oder linearisierten Funktionen die beste Anpassung für eine Menge von (x,y) Datenpunkten ergibt. Wir veranschaulichen die Verwendung dieser Funktion mit einem Beispiel. Angenommen Sie möchten ermitteln, welche der Datenanpassungsfunktionen die beste Anpassung für die folgenden Daten ergibt:

x	0.2	0.5	1	1.5	2	4	5	10
y	3.16	2.73	2.12	1.65	1.29	0.47	0.29	0.01

Geben Sie zunächst die Daten als Matrix ein, indem Sie entweder die Daten unter Verwendung des Matrix-Editors eingeben oder mit dem in Kapitel 10 entwickelten Programm CRMC zwei Listen von Daten für x und y eingeben. Speichern Sie dann diese Matrix mit der Funktion STOΣ in der Statistikmatrix ΣDAT.

Starten Sie anschließend mit ↱ STAT ▼ ▼ OK die Anwendung zur Datenanpassung. Es wird die aktuelle Variable ΣDAT angezeigt, die bereits geladen ist. Legen Sie ggf. im Einrichtungsfenster die folgenden Parameter fest:

Drücken Sie OK, um die folgende Ausgabe zu erhalten:

1: '3,99504833324*EXP(-,579206831203*X)'
2: Correlation: -0,996624999526
3: Covariance: -6,23350666124

Die beste Anpassung für die Daten lautet daher $y = 3{,}995\ e^{-0.58 \cdot x}$.

Ermitteln zusätzlicher Summenkenngrößen

Für einige Berechnungen von Stichprobenkenngrößen bietet sich die Anwendung **4. Summary stats..** im Menü STAT an. Drücken Sie zunächst erneut ↱ STAT, wechseln Sie mit der Nach-Unten-Taste ▼ zur vierten Option, und drücken Sie OK. Die anschließend angezeigte Eingabemaske enthält die folgenden Felder:

ΣDAT: Die Matrix mit den betreffenden Daten.

X-Col, Y-Col: Die Verwendung dieser Optionen ist nur sinnvoll, wenn die Matrix ΣDAT mehr als zwei Spalten enthält. Standardmäßig wird für Spalte x die Spalte 1 und für Spalte y die Spalte 2 verwendet.

_Σ**X** _ Σ**Y...**: Summenkenngrößen, die Sie als Ergebnisse dieses Programms auswählen können, indem Sie das entsprechende Feld mithilfe von [✓CHK] mit einem Häkchen versehen, wenn dieses Feld ausgewählt ist.

Viele dieser Summenkenngrößen werden zum Berechnen von Kenngrößen mit zwei Variablen (x,y) verwendet, die einen Bezug zur Funktion y = f(x) aufweisen. Daher gehört dieses Programm gewissermaßen zum Programm **3. Fit data..**

<u>Beispiel 1</u> – Ermitteln Sie für die gegenwärtig in ΣDAT vorhandenen x-y-Daten alle Summenkenngrößen.

- Rufen Sie die Option **summary stats...** mit ⟨→⟩ STAT ▼ ▼ ▼ OK auf.
- Wählen Sie die den x- und y-Daten entsprechenden Spaltennummern aus, d. h. X-Col: 1 und Y-Col: 2.
- Wählen Sie mit der Taste ✓CHK alle Optionen für die Ausgabe aus, d. h. _ΣX, _ΣY usw.
- Drücken Sie OK, um die folgenden Ergebnisse zu erhalten:

  ΣX: 24,2; ΣY: 11,72; ΣX2: 148,54; ΣY2: 26,6246; ΣXY: 12,602; NΣ:8

> **Hinweis:** Das Menü STAT enthält zwei weitere Anwendungen, nämlich **5. Hypth. Tests..** und **6. Conf. Interval..** Diese beiden Anwendungen werden weiter unten in diesem Kapitel erläutert.

Berechnung von Perzentilen

Durch Perzentile wird ein Datensatz in 100 Teile unterteilt. Das grundlegende Verfahren zum Berechnen des 100 p-ten Perzentils ($0 < p < 1$) in einer Stichprobe der Größe n lautet wie folgt:

1. Ordnen Sie die n Größen in aufsteigender Reihenfolge an.
2. Bestimmen Sie das Produkt $n \cdot p$.
   A. Wenn $n \cdot p$ keine Ganzzahl ist, runden Sie das Produkt auf die nächste Ganzzahl, und ermitteln Sie den entsprechenden Wert in dieser Reihenfolge.
   B. Wenn $n \cdot p$ eine Ganzzahl ist, z. B. k, berechnen Sie den Mittelwert der k-th und (k-1) th geordneten Werte.

> **Hinweis**: Regel für das Runden auf Ganzzahlen: Wenn für einen nicht ganzzahligen Wert x.yz… der Wert $y \geq 5$, runden Sie auf x+1. Wenn $y < 5$, runden Sie auf x.

Dieser Algorithmus kann durch das folgende, im RPN-Modus eingegebene Programm implementiert werden (Informationen über das Programmieren erhalten Sie in Kapitel 21):

« SORT DUP SIZE → p X n « n p * → k « IF k CEIL k FLOOR - NOT THEN X k GET X k 1 + GET + 2 / ELSE k 0 RND X SWAP GET END » » »

Wir speichern dieses Programm in der Variablen %TILE (percent-tile bzw. Perzentil). Für dieses Programm ist als Eingabe ein Wert p zwischen 0 und 1 erforderlich, der das 100p-Perzentil darstellt, sowie eine Liste von Werten. Das Programm gibt das 100p-Perzentil der Liste zurück.

Beispiel 1 – Bestimmen Sie das 37%-Perzentil der Liste { 2 1 0 1 3 5 1 2 3 6 7 9}. Geben Sie im RPN-Modus 0,27 ENTER { 2 1 0 1 3 5 1 2 3 6 7 9} ENTER %TILE ein. Geben Sie im ALG-Modus %TILE(0,27,{2,1,0,1,3,5,1,2,3,6,7,9} ein. Das Ergebnis lautet 1.

Das Menü STAT

Über das Menü STAT kann auf sämtliche oben beschriebenen vorprogrammierten Statistikfunktionen zugegriffen werden. Sie können das Menü STAT aufrufen, indem Sie im RPN-Modus folgenden Befehl verwenden: 96 MENU

Sie können ein eigenes Programm erstellen, z. B. STAT, um das Menü STAT direkt zu aktivieren. Der Inhalt dieses Programms ist lediglich: « 96 MENU ».

Das Menü STAT enthält die folgenden Funktionen:

Wenn Sie eine der diesen Menüs entsprechenden Tasten drücken, erhalten Sie wie unten beschrieben Zugriff auf unterschiedliche Funktionen.

Das Untermenü DATA

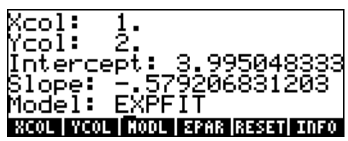
Das Untermenü DATA enthält Funktionen zum Bearbeiten der Statistikmatrix ΣDATA:

Diese Funktionen bewirken Folgendes:

Σ+ : fügt dem unteren Rand der Matrix ΣDATA eine Zeile auf Ebene 1 hinzu.
Σ- : entfernt die letzte Zeile in der Matrix ΣDATA und legt diese auf Ebene 1 des Stacks ab.
Die geänderte Matrix ΣDATA wird gespeichert.
CLΣ : löscht die aktuelle Matrix ΣDATA.
ΣDAT: legt Inhalte der aktuellen Matrix ΣDATA auf Ebene 1 des Stacks ab.
↰ ΣDAT: speichert die Matrix auf Ebene 1 des Stacks in der Matrix ΣDATA.

Das Untermenü ΣPAR

Das Untermenü ΣPAR enthält Funktionen zum Ändern von Statistikparametern. Die dargestellten Parameter entsprechen dem letzten Beispiel für die Datenanpassung.



Die auf dem Bildschirm angezeigten Parameter lauten:

Xcol: gibt die Spalte von ΣDATA an, die x darstellt (Standardwert: 1)
Ycol: gibt die Spalte von ΣDATA an, die y darstellt (Standardwert: 2)
Intercept: zeigt einen Abschnitt der letzten Datenanpassung an (Standardwert: 0)
Slope: zeigt die Steigung der letzten Datenanpassung an (Standardwert: 0)
Model: zeigt das aktuelle Datenanpassungsmodell an (Standardwert: LINFIT)

Die mit den Menütasten aufgerufenen Funktionen bewirken Folgendes:
XCOL: Aufruf mit n XCOL, ändert Spalte X in n.
YCOL: Aufruf mit n YCOL, ändert Spalte Y in n.
ΣPAR: zeigt Statistikparameter an.
RESET: setzt Parameter auf die Standardwerte zurück.
INFO: zeigt Statistikparameter an.

Das Untermenü MODL in Σ**PAR**
Dieses Untermenü enthält Funktionen, mit denen Sie durch Drücken der entsprechenden Taste das Datenanpassungsmodell in LINFIT, LOGFIT, EXPFIT, PWRFIT oder BESTFIT ändern können.

Das Untermenü 1VAR

Das Untermenü 1VAR enthält Funktionen zum Berechnen der Kenngrößen für die Spalten in der Matrix ΣDATA.

Folgende Funktionen sind verfügbar:

TOT : zeigt die Summe jeder Spalte in der Matrix ΣDATA an.
MEAN : zeigt den arithmetischen Mittelwert jeder Spalte in der Matrix ΣDATA an.
SDEV : zeigt die Standardabweichung jeder Spalte in der Matrix ΣDATA an.
MAXΣ : zeigt den Maximalwert jeder Spalte in der Matrix ΣDATA an.
MINΣ : zeigt den Minimalwert jeder Spalte in der Matrix ΣDATA an.
BINS : stellt bei Verwendung von x_s, Δx, n [BINS] die Häufigkeitsverteilung für Daten in der Spalte Xcol der Matrix ΣDATA mit den durch $[x_s,x_s+\Delta x]$, $[x_s,x_s+2\Delta x]$,…, $[x_s,x_s+n\Delta x]$ definierten Häufigkeitsklassen bereit.
VAR : zeigt die Varianz jeder Spalte in der Matrix ΣDATA an.
PSDEV : zeigt die Grundgesamtheitsstandardabweichung (anhand von n statt von (n-1)) jeder Spalte in der Matrix ΣDATA an.
PVAR : zeigt die Grundgesamtheitsvarianz jeder Spalte in der Matrix ΣDATA an.
MINΣ : zeigt den Mindestwert jeder Spalte in der Matrix ΣDATA an.

Das Untermenü PLOT

Das Untermenü PLOT enthält Funktionen zum Erzeugen grafischer Darstellungen der Daten in der Matrix ΣDATA.

Diese Funktionen lauten:

BARPL : erzeugt ein Balkendiagramm mit den Daten in der Spalte Xcol der Matrix ΣDATA.

HISTP : erzeugt ein Histogramm der Daten in der Spalte Xcol der Matrix ΣDATA, wobei die 13 Klassen entsprechende Standardbreite verwendet wird, sofern die Klassengröße nicht mit der Funktion BINS im Untermenü 1VAR (siehe oben) geändert wird.

SCATR : erzeugt eine Punktwolke der Daten in der Spalte Ycol der Matrix ΣDATA gegen die Daten in der Spalte Xcol der Matrix ΣDATA. Die angepasste Gleichung wird in der Variablen EQ gespeichert.

Das Untermenü FIT

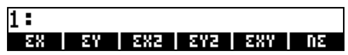
Das Untermenü FIT enthält Funktionen zum Anpassen von Gleichungen an die Daten in den Spalten Xcol und Ycol der Matrix ΣDATA.

Die in diesem Untermenü verfügbaren Funktionen lauten:

ΣLINE : stellt die der letzten Anpassung entsprechende Gleichung bereit.
LR : stellt Abschnitt und Steigung der letzten Anpassung bereit.
PREDX : Aufruf mit y PREDX, Ermitteln von x bei gegebenem y für die Anpassung y = f(x).
PREDY : Aufruf mit x PREDY, Ermitteln von y bei gegebenem x für die Anpassung y = f(x).
CORR : stellt den Korrelationskoeffizienten für die letzte Anpassung bereit.
COV : stellt die Stichprobenkovarianz für die letzte Anpassung bereit.
PCOV : zeigt die Grundgesamtheitskovarianz für die letzte Anpassung an.

Das Untermenü SUMS

Das Untermenü SUMS enthält Funktionen zum Ermitteln von Summenkenngrößen der Daten in den Spalten Xcol und Ycol der Matrix ΣDATA.



ΣX : stellt die Summe der Werte in der Spalte Xcol bereit.
ΣY : stellt die Summe der Werte in der Spalte Ycol bereit.
ΣX^2 : stellt die Summe der Quadrate der Werte in der Spalte Xcol bereit.
ΣY^2 : stellt die Summe der Quadrate der Werte in der Spalte Ycol bereit.
ΣX*Y : stellt die Summe von x·y bereit, d. h. der Produkte der Daten in den Spalten Xcol und Ycol.
NΣ : stellt die Anzahl der Spalten in der Matrix ΣDATA bereit.

Beispiel für Operationen des Menüs STAT

ΣDATA sei die auf der nächsten Seite dargestellte Matrix.

- Geben Sie mit dem Matrix-Editor die Matrix auf Ebene 1 des Stacks ein.
- Speichern Sie die Matrix in ΣDATA mit dem Befehl DATA ← ΣDAT
- Berechnen Sie die Kenngröße für jede Spalte: STAT 1VAR:

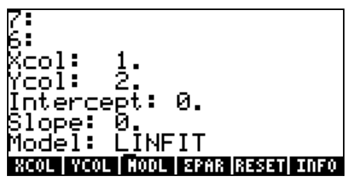
TOT	ergibt [38,5 87,5 82799,8]
MEAN	ergibt [5,5, 12,5 11828,54…]
SDEV	ergibt [3,39… 6,78… 21097,01…]
MAXΣ	ergibt [10 21,5 55066]
MINΣ	ergibt [1,1 3,7 7,8]
NXT VAR	ergibt [11,52 46,08 445084146,33]
PSDEV	ergibt [3,142… 6,284… 19532,04…]
PVAR	ergibt [9,87… 39,49… 381500696,85…]

- Daten:

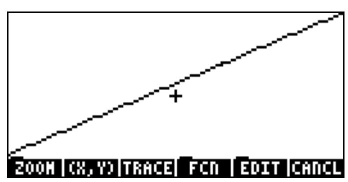
$$\begin{bmatrix} 1.1 & 3.7 & 7.8 \\ 3.7 & 8.9 & 101 \\ 2.2 & 5.9 & 25 \\ 5.5 & 12.5 & 612 \\ 6.8 & 15.1 & 2245 \\ 9.2 & 19.9 & 24743 \\ 10.0 & 21.5 & 55066 \end{bmatrix}$$

- Erzeugen Sie eine Punktwolke für die Daten in den Spalten 1 und 2, und zeichnen Sie eine entsprechende gerade Linie:

STAT ΣPAR RESET — setzt die Statistikparameter zurück



NXT STAT PLOT SCATR — erzeugt Punktwolke
STATL — zeichnet die Datenanpassung als gerade Linie



CANCL — wechselt zum Hauptbildschirm

- Bestimmen Sie die Anpassungsgleichung und einige ihrer Kenngrößen:

STAT FIT ΣLINE — ergibt `'1,5+2*X'`

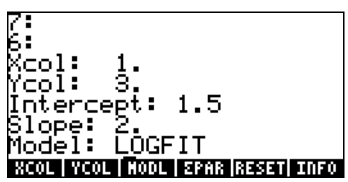
LR	ergibt `Intercept: 1,5, Slope: 2`
3 PREDX	ergibt 0,75
1 PREDY	ergibt 3,50
CORR	ergibt 1,0
COV	ergibt 23,04
NXT PCOV	ergibt 19,74...

- Ermitteln Sie Summenkenngrößen für die Daten in den Spalten 1 und 2: STAT SUMS:

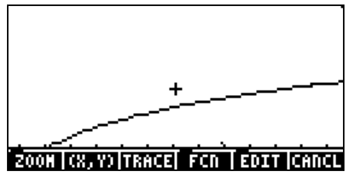
ΣX	ergibt 38,5
ΣY	ergibt 87,5
ΣX2	ergibt 280,87
ΣY2	ergibt 1370,23
ΣXY	ergibt 619,49
NΣ	ergibt 7

- Passen Sie die Daten in den Spalten 1 (x) und 3 (y) mit einer logarithmischen Anpassung an:

NXT STAT ΣPAR 3 YCOL	legt Ycol = 3 und
MODL LOGFI	legt Model = Logfit fest



NXT STAT PLOT SCATR	erzeugt ein Streuungsdiagramm von y gegen x
STATL	zeigt die Linie für logarithmische Anpassung an

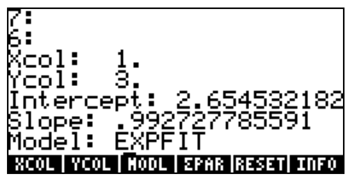


Offensichtlich ist die logarithmische Anpassung keine gute Lösung.

CANCL wechselt zum Hauptbildschirm

- Wählen Sie mit folgendem Befehl die beste Anpassung aus:

STAT ΣPAR MODL BESTF zeigt EXPFIT als beste Anpassung für diese Daten an



NXT STAT FIT ΣLINE	ergibt '2,6545*EXP(0,9927*X)'
CORR	ergibt 0,99995… (gute Korrelation)
2300 PREDX	ergibt 6,8139
5,2 PREDY	ergibt 463,37
NXT STAT PLOT SCATR	erzeugt ein Streuungsdiagramm von y gegen x
STATL	zeigt die Linie für logarithmische Anpassung an

- Um zum Menü STAT zurückzukehren, verwenden Sie NXT STAT.
- Um wieder zum Variablenmenü zurückzukehren, verwenden Sie VAR.

Vertrauensbereiche

Bei der statistischen Inferenz handelt es sich um Schlussfolgerungen in Bezug auf eine Grundgesamtheit anhand der aus den Stichprobendaten gewonnenen Informationen. Damit die Stichprobendaten aussagekräftig sind, muss die Stichprobe zufällig sein, d. h., die Auswahl einer bestimmten Stichprobe muss mit derselben Wahrscheinlichkeit wie die Auswahl jeder anderen möglichen Stichprobe aus einer bestimmten Grundgesamtheit erfolgen. Im Folgenden werden einige für das Konzept der Zufallsstichprobe relevanten Begriffe erläutert:

- Grundgesamtheit : die Menge aller denkbaren Werte eines Prozesses oder Attributs einer Komponente.
- Stichprobe : Teilmenge einer Grundgesamtheit.
- Zufallsstichprobe : eine für die Grundgesamtheit repräsentative Stichprobe.
- Zufallsvariable : für einen Stichprobenraum definierte reellwertige Funktion. Kann diskret oder kontinuierlich sein.

Falls eine bestimmte Wahrscheinlichkeitsverteilung auf die Grundgesamtheit zutrifft, die von einem Parameter θ abhängt, kann zum Schätzen von θ eine Zufallsstichprobe von Werten (X_1, X_2, X_3, …, X_n) der Größe n verwendet werden.

- Stichprobenverteilung : die gemeinsame Wahrscheinlichkeitsverteilung von $X_1,X_2,X_3,\ldots, X_n$.
- Kenngröße : jede Funktion der Werte, die quantifizierbar ist und keine unbekannten Parameter enthält. Eine Kenngröße ist eine Zufallsvariable, die für Schätzungen verwendet werden kann.
- Punktschätzung : wenn für den Parameter θ ein einziger Wert bereitgestellt wird.
- Vertrauensbereich : ein numerisches Intervall, das mit einer bestimmten Wahrscheinlichkeit den Parameter θ enthält.
- Schätzfunktion : Regel oder Methode zum Schätzen des Parameters θ.

- Schätzwert : der Wert, den die Schätzfunktion in einer bestimmten Anwendung zurückgibt.

Beispiel 1 – X stelle die Zeit (Stunden) dar, die für die Ausführung eines bestimmten Fertigungsprozesses erforderlich ist. Gegeben sei folgende Stichprobe der Werte von X: 2,2 2,5 2,1 2,3 2,2. Die Grundgesamtheit, der die Stichprobe entnommen wurde, ist die Menge aller möglichen Werte für die Fertigungszeit und daher eine unendliche Grundgesamtheit. Angenommen der zu schätzende Grundgesamtheitsparameter ist der Mittelwert μ. Wir verwenden als Schätzfunktion den Mittelwert $\overline{X}$ der Stichprobe, der durch folgende Gleichung (Regel) definiert ist: $\overline{X} = \frac{1}{n} \cdot \sum_{i=1}^{n} X_i$.

Für die betreffende Stichprobe ist der Schätzwert für μ die Stichprobenkenngröße $\overline{x}$ = (2,2+2,5+2,1+2,3+2,2)/5 = 2,36. Dieser einzelne Wert von $\overline{X}$, also $\overline{x}$ = 2,36, stellt eine Punktschätzung des Grundgesamtheitsparameters μ dar.

Schätzung von Vertrauensbereichen

Die nächste Stufe der Inferenz nach der Punktschätzung ist die Intervallschätzung. Dies bedeutet, dass wir nicht einen einzelnen Wert einer Schätzfunktion ermitteln, sondern zwei Kenngrößen a und b angeben, durch die ein Intervall definiert wird, das mit einer bestimmten Wahrscheinlichkeit den Parameter θ enthält. Die Endpunkte des Intervalls werden als Vertrauensgrenzen und das Intervall (a,b) als Vertrauensbereich bezeichnet.

Definitionen

(C_l,C_u) sei ein Vertrauensbereich, der den unbekannten Parameter θ enthält.

- Die statistische Sicherheit bzw. Aussagewahrscheinlichkeit ist die Menge $(1-\alpha)$ mit $0 < \alpha < 1$, sodass $P[C_l < \theta < C_u] = 1 - \alpha$, wobei P[ ] eine Wahrscheinlichkeit darstellt (siehe Kapitel 17). Der vorherige Ausdruck definiert die so genannten zweiseitigen Vertrauensgrenzen.
- Ein unterer einseitiger Vertrauensbereich wird durch $Pr[C_l < \theta] = 1 - \alpha$ definiert.

- Ein oberer einseitiger Vertrauensbereich wird durch $\Pr[\theta < C_u] = 1 - \alpha$ definiert.
- Der Parameter α wird als Signifikanzniveau bezeichnet. Typische Werte für α sind 0,01, 0,05 und 0,1, die den Aussagewahrscheinlichkeiten 0,99, 0,95 bzw. 0,90 entsprechen.

Vertrauensbereiche für den Grundgesamtheitsmittelwert bei bekannter Grundgesamtheitsvarianz

$\bar{X}$ sei der Mittelwert einer Zufallsstichprobe der Größe n, die einer unendlichen Grundgesamtheit mit der bekannten Standardabweichung σ entnommen wurde. Der zentrale zweiseitige Vertrauensbereich $100(1-\alpha)\%$ [d. h. 99 %, 95 %, 90 %, usw.] für den Grundgesamtheitsmittelwert μ ist $(\bar{X}-z_{\alpha/2}\cdot\sigma/\sqrt{n}\ ,\ \bar{X}+z_{\alpha/2}\cdot\sigma/\sqrt{n})$, wobei $z_{\alpha/2}$ eine normalverteilte Zufallsvariable darstellt, die mit der Wahrscheinlichkeit $\alpha/2$ überschritten wird. Der Standardfehler des Stichprobenmittelwertes $\bar{X}$ ist $\cdot\sigma/\sqrt{n}$.

Die obere und untere einseitige Vertrauensgrenze $100(1-\alpha)\%$ für den Grundgesamtheitsmittelwert μ lautet $X+z_\alpha\cdot\sigma/\sqrt{n}$ bzw. $\bar{X}-z_\alpha\cdot\sigma/\sqrt{n}$. Somit ist ein unterer einseitiger Vertrauensbereich durch $(-\infty\ ,\ X+z_\alpha\cdot\sigma/\sqrt{n})$ und ein oberer einseitiger Vertrauensbereich durch $(X-z_\alpha\cdot\sigma/\sqrt{n},+\infty)$ definiert. Beachten Sie, dass wir in diesen letzten beiden Vertrauensbereichen nicht den Wert $z_{\alpha/2}$, sondern z_α verwenden.

Im Allgemeinen ist der Wert z_k in der Standardnormalverteilung als der Wert von z definiert, dessen Überschreitungswahrscheinlichkeit k ist, d. h. $\Pr[Z>z_k] = k$ oder $\Pr[Z<z_k] = 1 - k$. Die Normalverteilung wurde in Kapitel 17 erläutert.

Vertrauensbereiche für den Grundgesamtheitsmittelwert bei unbekannter Grundgesamtheitsvarianz

$\bar{X}$ und S sei der Mittelwert bzw. die Standardabweichung einer Zufallsstichprobe der Größe n, die einer unendlichen Grundgesamtheit mit Normalverteilung und der unbekannten Standardabweichung σ entnommen wurde. Der zentrale zweiseitige Vertrauensbereich $100\cdot(1-\alpha)\%$ [d. h. 99 %, 95 %, 90 %, usw.] für den Grundgesamtheitsmittelwert μ ist $(\bar{X}-t_{n-1,\alpha/2}\cdot S$

$/\sqrt{n}$, $\bar{X}+ t_{n-1, \alpha/2} \cdot S/\sqrt{n}$), wobei $t_{n-1, \alpha/2}$ eine Studentsche t-Verteilung mit dem Freiheitsgrad ν = n-1 und der Überschreitungswahrscheinlichkeit $\alpha/2$ darstellt.

Die obere und untere einseitige Vertrauensgrenze 100 · (1-α)% für den Grundgesamtheitsmittelwert μ lautet

$$X + t_{n-1, \alpha/2} \cdot S/\sqrt{n} \text{ bzw. } \bar{X} - t_{n-1, \alpha/2} \cdot S /\sqrt{n}.$$

Kleine und große Stichproben

Für die Studentsche t-Verteilung gilt, dass sie für n>30 nicht von der Standardnormalverteilung zu unterscheiden ist. Wenn daher bei Stichproben mit mehr als 30 Elementen die Grundgesamtheitsvarianz nicht bekannt ist, können Sie denselben Vertrauensbereich wie bei bekannter Grundgesamtheitsvarianz verwenden, müssen jedoch σ durch S ersetzen. Stichproben mit n>30 werden üblicherweise als große Stichproben bezeichnet, andernfalls als kleine Stichproben.

Vertrauensbereich für eine Quote

Eine diskrete Zufallsvariable X folgt einer Bernoulli-Verteilung, wenn X nur zwei Werte annehmen kann: X = 0 (Fehlschlag) und X = 1 (Erfolg). Wenn X ~ Bernoulli(p) und p die Erfolgswahrscheinlichkeit ist, ist der Mittelwert bzw. Erwartungswert von X gleich E[X] = p, und die Varianz lautet Var[X] = p(1-p).

Wenn ein Experiment mit X n Mal wiederholt wird und k erfolgreiche Ergebnisse aufgezeichnet werden, ist der Schätzwert p durch p' = k/n definiert, und der Standardfehler p' ist $\sigma_{p'} = \sqrt{(p \cdot (1-p)/n)}$. In der Praxis wird in der Formel für den Standardfehler p durch den Stichprobenschätzwert für p, d. h. p', ersetzt.

Bei einer großen Stichprobe mit n>30, n·p>5 und n·(1-p)>5 ist die Stichprobenverteilung nahezu eine Normalverteilung. Daher ist der zentrale zweiseitige Vertrauensbereich 100(1-α)% für den Grundgesamtheitsmittelwert $(p'+z_{\alpha/2} \cdot \sigma_{p'},\ p'+z_{\alpha/2} \cdot \sigma_{p'})$. Bei einer kleinen Stichprobe (n<30) kann mit $(p'-t_{n-1,\alpha/2} \cdot \sigma_{p'}, p'+t_{n-1,\alpha/2} \cdot \sigma_{p'})$ ein Schätzwert für den Vertrauensbereich ermittelt werden.

Stichprobenverteilung für Differenzen und Summen von Kenngrößen

S_1 und S_2 seien unabhängige Kenngrößen auf der Grundlage von zwei Stichproben der Größe n_1 bzw. n_2 aus zwei Grundgesamtheiten. Außerdem seien die jeweiligen Mittelwerte und Standardfehler der Stichprobenverteilungen dieser Kenngrößen μ_{S1} und μ_{S2} bzw. σ_{S1} und σ_{S2}. Die Differenz der aus den beiden Grundgesamtheiten S_1-S_2 weist eine Stichprobenverteilung mit dem Mittelwert $\mu_{S1-S2} = \mu_{S1} - \mu_{S2}$ und dem Standardfehler $\sigma_{S1-S2} = (\sigma_{S1}^2 + \sigma_{S2}^2)^{1/2}$ auf. Außerdem weist die Summe der Kenngrößen T_1+T_2 den Mittelwert $_{S1+S2} = \mu_{S1} + \mu_{S2}$ und den Standardfehler $\sigma_{S1+S2} = (\sigma_{S1}^2 + \sigma_{S2}^2)^{1/2}$ auf.

Schätzfunktionen für den Mittelwert und die Standardabweichung von Differenz und Summe der Kenngrößen S_1 und S_2 sind durch folgende Gleichungen definiert:

$$\hat{\mu}_{S_1 \pm S_2} = \overline{X}_1 \pm \overline{X}_2, \qquad \hat{\sigma}_{S_1 \pm S_2} = \sqrt{\frac{\sigma_{S1}^2}{n_1} + \frac{\sigma_{S2}^2}{n_2}}$$

In diesen Ausdrücken sind $\overline{X}_1$ und $\overline{X}_2$ die Werte der Kenngrößen S_1 und S_2 von den aus den beiden Grundgesamtheiten entnommenen Stichproben, und σ_{S1}^2 und σ_{S2}^2 sind die Varianzen der Grundgesamtheiten der Kenngrößen S_1 und S_2, aus denen die Stichproben entnommen wurden.

Vertrauensbereiche für Summen und Differenzen von Mittelwerten

Wenn die Grundgesamtheitsvarianzen σ_1^2 und σ_2^2 bekannt sind, werden die Vertrauensbereiche für Differenz und Mittelwerte der Grundgesamtheiten, d. h. $\mu_1 \pm \mu_2$, durch folgenden Ausdruck definiert:

$$\left((\overline{X}_1 \pm X_2) - z_{\alpha/2} \cdot \sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}},\ (\overline{X}_1 \pm X_2) + z_{\alpha/2} \cdot \sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}} \right)$$

Bei großen Stichproben, d. h. $n_1>30$ und $n_2>30$, und unbekannten, jedoch gleichen Grundgesamtheitsvarianzen $\sigma_1^2 = \sigma_2^2$ werden die Vertrauensbereiche für Differenz und Mittelwerte der Grundgesamtheiten, d. h. $\mu_1\pm\mu_2$, durch folgenden Ausdruck definiert:

$$\left((\overline{X}_1 \pm X_2) - z_{\alpha/2} \cdot \sqrt{\frac{S_1^2}{n_1} + \frac{S_2^2}{n_2}}, (\overline{X}_1 \pm X_2) + z_{\alpha/2} \cdot \sqrt{\frac{S_1^2}{n_1} + \frac{S_2^2}{n_2}} \right).$$

Wenn eine der Stichproben klein ist, d. h. $n_1<30$ oder $n_2<30$, und die Grundgesamtheitsvarianzen $\sigma_1^2 = \sigma_2^2$ unbekannt, jedoch gleich sind, können wir für die Abweichung $\mu_1\pm\mu_2$ den „zusammengefassten" Schätzwert $s_p^2 = [(n_1-1)\cdot s_1^2+(n_2-1)\cdot s_2^2]/(n_1+n_2-2)$ ermitteln.

In diesem Fall sind die zentralen Vertrauensbereiche für die Summe und Differenz der Mittelwerte der Grundgesamtheiten, d. h. $\mu_1\pm\mu_2$, durch folgenden Ausdruck definiert:

$$\left((\overline{X}_1 \pm X_2) - t_{\nu,\alpha/2} \cdot s_p^2, (\overline{X}_1 \pm X_2) + t_{\nu,\alpha/2} \cdot s_p^2 \right)$$

Hierbei stellt $\nu = n_1+n_2-2$ den Freiheitsgrad in der Studentschen t-Verteilung dar.

Für die letzten beiden Optionen geben wir an, dass die Varianzen der Grundgesamtheit gleich sein müssen, obwohl sie nicht bekannt sind. Dies ist der Fall, wenn die beiden Stichproben derselben Grundgesamtheit oder zwei Grundgesamtheiten entnommen wurden, von denen wir annehmen, dass sie dieselbe Varianz der Grundgesamtheit aufweisen. Wenn wir jedoch Grund zu der Annahme haben, dass die beiden unbekannten Varianzen der Grundgesamtheit voneinander abweichen, können wir folgenden Vertrauensbereich verwenden:

$$\left((\overline{X}_1 \pm X_2) - t_{\nu,\alpha/2} \cdot s^2_{\overline{X}_1 \pm \overline{X}_2}, (\overline{X}_1 \pm X_2) + t_{\nu,\alpha/2} \cdot s^2_{\overline{X}_1 \pm \overline{X}_2} \right)$$

Hierbei ist die geschätzte Standardabweichung für Summe oder Differenz durch

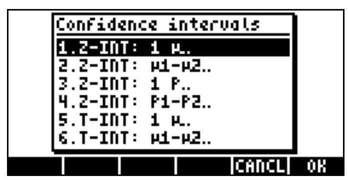
$$s_{\bar{X}_1 \pm \bar{X}_2} = \sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}$$

definiert und ν, der Freiheitsgrad der t-Abweichung, wird mit folgender Formel berechnet (das Ergebnis wird auf die nächste Ganzzahl gerundet):

$$\nu = \frac{[(S_1^2/n_1) + (S_2^2/n_2)]^2}{[(S_1^2/n_1)/(n_1-1)] + [(S_2^2/n_2)/(n_2-1)]}$$

Bestimmen von Vertrauensbereichen

Die Anwendung **6. Conf Interval** kann mit ↱ STAT ▲ OK aufgerufen werden. Die Anwendung enthält die folgenden Optionen:



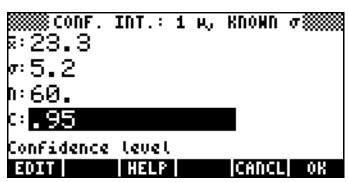
Diese Optionen werden im Folgenden erläutert:

1. Z-INT: 1 μ : Vertrauensbereich einer einzelnen Stichprobe für den Mittelwert der Grundgesamtheit μ mit bekannter Varianz der Grundgesamtheit oder bei großen Stichproben mit unbekannter Varianz der Grundgesamtheit.
2. Z-INT: μ1−μ2: Vertrauensbereich für die Differenz der Mittelwerte der Grundgesamtheit μ_1- μ_2 mit entweder bekannten Varianzen der Grundgesamtheit oder bei großen Stichproben mit unbekannten Varianzen der Grundgesamtheit.

3. Z-INT : 1 p   : Vertrauensbereich einer einzelnen Stichprobe für die Quote p bei großen Stichproben mit unbekannter Varianz der Grundgesamtheit.
4. Z-INT : p1– p2: Vertrauensbereich für die Differenz zweier Quoten p_1-p_2 für große Stichproben mit unbekannten Varianzen der Grundgesamtheit.
5. T-INT : 1 μ   : Vertrauensbereich einer einzelnen Stichprobe für den Grundgesamtheitsmittelwert μ bei kleinen Stichproben mit unbekannter Varianz der Grundgesamtheit.
6. T-INT : μ1–μ2: Vertrauensbereich für die Differenz zweier Grundgesamtheitsmittelwerte μ_1-μ_2 für kleine Stichproben mit unbekannten Varianzen der Grundgesamtheit.

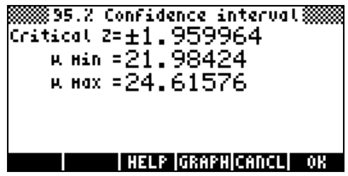
Beispiel 1 – Bestimmen Sie den zentralen Vertrauensbereich für den Mittelwert einer Grundgesamtheit, wenn eine Stichprobe mit 60 Elementen bedeutet, dass der Mittelwert der Stichprobe $\bar{x}$ = 23,2 und die Standardabweichung s = 5,2 beträgt. Verwenden Sie α = 0,05. Die statistische Sicherheit beträgt C = 1-α = 0,95.

Wählen Sie die erste Option im oben abgebildeten Menü aus, indem Sie OK drücken. Geben Sie die erforderlichen Werte wie dargestellt in die Eingabemaske ein:



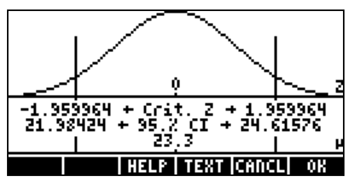
Drücken Sie HELP, um ein Fenster aufzurufen, in dem der Vertrauensbereich anhand vom Taschenrechner generierter Zufallszahlen erläutert wird. Blättern Sie im dann angezeigten Fenster mithilfe der Taste mit dem nach unten weisenden Pfeil ▽ nach unten. Drücken Sie OK, wenn Sie das Hilfefenster schließen möchten. Dadurch wird das oben abgebildete Fenster erneut angezeigt.

Drücken Sie OK, um den Vertrauensbereich zu berechnen. Das vom Taschenrechner angezeigte Ergebnis lautet:



Das Ergebnis bedeutet, dass ein Vertrauensbereich von 95 % berechnet wurde. Der im obigen Fenster angezeigte Wert für Critical z entspricht den Werten $\pm z_{\alpha/2}$ in der Formel des Vertrauensbereichs ($\overline{X}-z_{\alpha/2}\cdot\sigma/\sqrt{n}$, $\overline{X}+z_{\alpha/2}\cdot\sigma/\sqrt{n}$). Die Werte μ Min und μ Max stellen die obere bzw. untere Grenze dieses Intervalls dar, d. h μ Min = $\overline{X}-z_{\alpha/2}\cdot\sigma/\sqrt{n}$ und μ Max = $\overline{X}+z_{\alpha/2}\cdot\sigma/\sqrt{n}$.

Drücken Sie GRAPH, um eine grafische Darstellung des Vertrauensbereichs anzuzeigen:

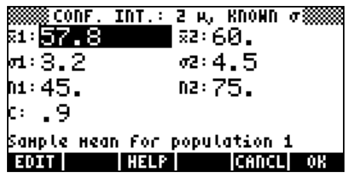


Das Diagramm stellt die Wahrscheinlichkeitsdichtefunktion, die Position der kritischen Punkte $\pm z_{\alpha/2}$, den Mittelwert (23,2) und die entsprechenden Bereichsgrenzen (21,88424 und 24,51576) dar. Drücken Sie TEXT, um zum vorherigen Ergebnisfenster zurückzukehren und/oder drücken Sie OK, um die Vertrauensbereichsumgebung zu schließen. Die Ergebnisse werden auf dem Bildschirm des Taschenrechners angezeigt.

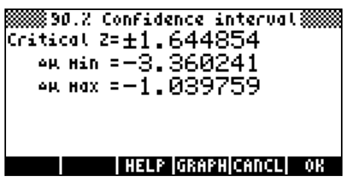
<u>Beispiel 2</u> – Die Daten aus zwei Stichproben (Stichprobe 1 und 2) geben an, dass $\bar{x}_1$ = 57,8 und $\bar{x}_2$ = 60,0. Die Stichprobengrößen betragen n_1 = 45 und n_2 = 75. Wenn bekannt ist, dass die Standardabweichungen der Grundgesamtheiten σ_1 = 3,2 und σ_2 = 4,5 betragen, bestimmen Sie den

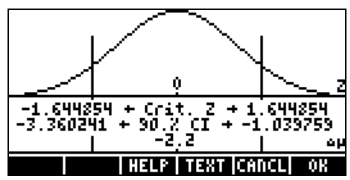
Vertrauensbereich von 90 % für die Differenz der Grundgesamtheitsmittelwerte, d. h. μ_1- μ_2.

Drücken Sie ↱ STAT ▲ OK, um die Vertrauensbereichsfunktion des Taschenrechners aufzurufen. Drücken Sie ▼ OK, um Option 2. Z-INT: μ1–μ2.. auszuwählen. Geben Sie folgende Werte ein:



Drücken Sie anschließend OK. Die Ergebnisse werden unten in Text- und Diagrammform dargestellt:

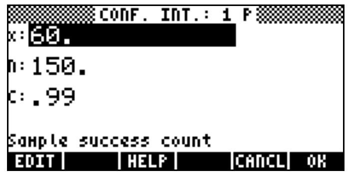




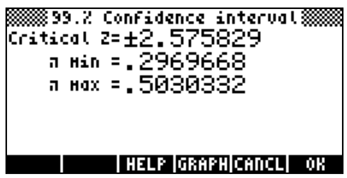
Die Variable Δμ stellt μ1–μ2 dar.

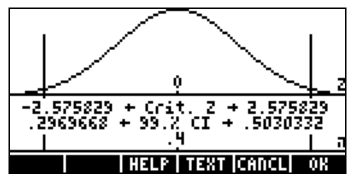
<u>Beispiel 3</u> – Eine Meinungsumfrage gibt an, dass 60 Personen aus einer Stichprobe von 150 Personen höhere Vermögenssteuern zur Finanzierung öffentlicher Projekte befürworten. Bestimmen Sie einen Vertrauensbereich von 99 % für den Grundgesamtheitsanteil, der höhere Steuern befürwortet.

Drücken Sie ↱ STAT ▲ OK, um die Vertrauensbereichsfunktion des Taschenrechners aufzurufen. Drücken Sie ▼▼ OK, um Option 3. Z-INT: μ1–μ2.. auszuwählen. Geben Sie folgende Werte ein:



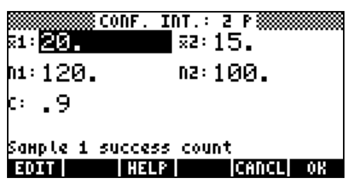
Drücken Sie anschließend OK. Die Ergebnisse werden unten als Text- und Diagramm dargestellt:



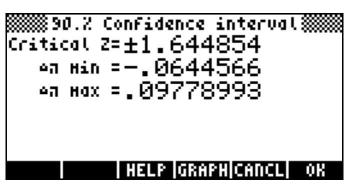


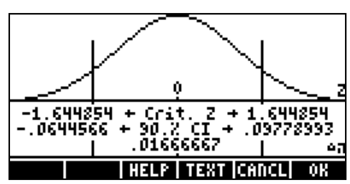
Beispiel 4 – Bestimmen Sie einen Vertrauensbereich von 90 % für die Differenz der beiden Anteile, wenn Stichprobe 1 unter 120 Versuchen 20 Erfolge aufweist und Stichprobe 2 unter 100 Versuchen 15 Erfolge aufweist.

Drücken Sie STAT OK, um die Vertrauensbereichsfunktion des Taschenrechners aufzurufen. Drücken Sie OK, um Option 4. Z-INT: p1–p2.. aufzurufen. Geben Sie folgende Werte ein:



Drücken Sie anschließend OK. Die Ergebnisse werden unten in Text- und Diagrammform dargestellt:

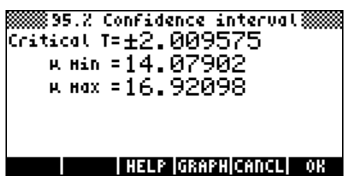


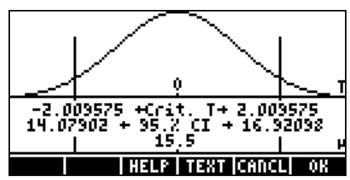


Beispiel 5 – Bestimmen Sie einen Vertrauensbereich von 95 % für den Mittelwert der Grundgesamtheit, wenn eine Stichprobe von 50 Elementen den Mittelwert 15,5 und die Standardabweichung 5 aufweist. Die Standardabweichung der Grundgesamtheit ist nicht bekannt.

Drücken Sie ↱ STAT ▲ OK, um die Vertrauensbereichsfunktion des Taschenrechners aufzurufen. Drücken Sie ▲ ▲ OK, um Option 5. T-INT: μ aufzurufen. Geben Sie folgende Werte ein:

Drücken Sie anschließend OK. Die Ergebnisse werden unten in Text- und Diagrammform dargestellt:

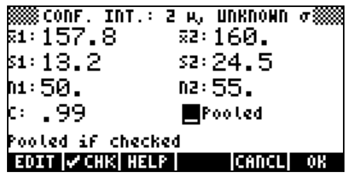




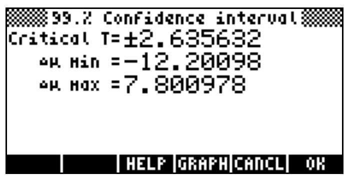
Die Abbildung stellt die Wahrscheinlichkeitsdichtefunktion der Studentschen t-Verteilung für den Freiheitsgrad $\nu = 50-1 = 49$ dar.

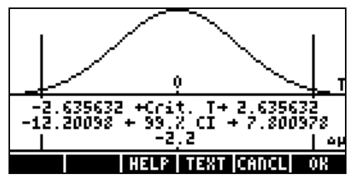
Beispiel 6 – Bestimmen Sie einen Vertrauensbereich von 99 % für die Differenz der Mittelwerte zweier Grundgesamtheiten mit den Stichprobendaten $\bar{x}_1 = 157{,}8$, $\bar{x}_2 = 160{,}0$ und $n_1 = 50$, $n_2 = 55$. Die Standardabweichungen der Grundgesamtheiten betragen $s_1 = 13{,}2$ und $s_2 = 24{,}5$.

Drücken Sie ↱ STAT ▲ OK, um die Vertrauensbereichsfunktion des Taschenrechners aufzurufen. Drücken Sie ▲ OK, um Option 6. T-INT: μ1–μ2 aufzurufen. Geben Sie folgende Werte ein:

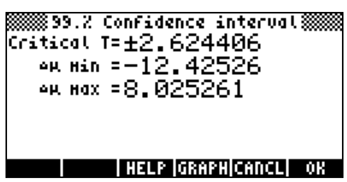


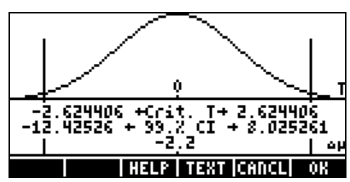
Drücken Sie anschließend OK. Die Ergebnisse werden unten in Text- und Diagrammform dargestellt:





Bei diesen Ergebnissen wird vorausgesetzt, dass die Werte s_1 und s_2 die Standardabweichungen der Grundgesamtheiten darstellen. Wenn diese Werte jedoch die Standardabweichungen der Stichproben darstellen, müssen Sie dieselben Werte wie zuvor, jedoch mit Auswahl der Option `_pooled` eingeben. Die Ergebnisse lauten nun:





Vertrauensbereiche für die Varianz

Zum Erstellen einer Formel für den Vertrauensbereich der Varianz stellen wir zuerst die Stichprobenverteilung der Varianz vor: Gegeben sei eine Zufallsstichprobe X_1, X_2 ..., X_n unabhängiger normalverteilter Variablen mit dem Mittelwert μ, der Varianz σ^2 und dem Stichprobenmittelwert $\bar{X}$. Die Kenngröße

$$\hat{S}^2 = \frac{1}{n-1} \cdot \sum_{i=1}^{n} (X_i - \bar{X})^2,$$

ist eine erwartungstreue Schätzfunktion der Varianz σ^2.

Die Menge $(n-1)\cdot\frac{\hat{S}^2}{\sigma^2} = \sum_{i=1}^{n}(X_i - \overline{X})^2$, weist eine Chi-Quadrat-Verteilung χ_{n-1}^2 mit dem Freiheitsgrad ν = n-1 auf. Der beidseitige Vertrauensbereich (1-α)·100 % wird durch

$$\Pr[\chi^2_{n-1,1-\alpha/2} < (n-1)\cdot S^2/\sigma^2 < \chi^2_{n-1,\alpha/2}] = 1-\alpha \text{ ermittelt.}$$

Der Vertrauensbereich für die Grundgesamtheitsvarianz σ^2 lautet daher

$$[(n-1)\cdot S^2/\ \chi^2_{n-1,\alpha/2}\ ;\ (n-1)\cdot S^2/\ \chi^2_{n-1,1-\alpha/2}].$$

Hierbei stellen $\chi^2_{n-1,\alpha/2}$ und $\chi^2_{n-1,1-\alpha/2}$ die Werte dar, um die eine Variable χ^2 mit dem Freiheitsgrad ν = n-1 und den Überschreitungswahrscheinlichkeiten $\alpha/2$ bzw. 1- $\alpha/2$ von den Erwartungswerten abweichen kann.

Die obere einseitige Vertrauensbereichsgrenze für σ^2 ist durch $(n-1)\cdot S^2/\ \chi^2_{n-1,1-\alpha}$ definiert.

Beispiel 1 – Bestimmen Sie anhand der Ergebnisse aus einer Stichprobe der Größe n = 25, die eine Stichprobenvarianz s^2 = 12,5 angibt, für die Grundgesamtheitsvarianz σ^2 einen Vertrauensbereich von 95 %.

In Kapitel 17 wird zum Lösen der Gleichung α = UTPC(γ,x) der numerische Gleichungslöser verwendet. In diesem Programm stellt γ den Freiheitsgrad (n-1) und α die Wahrscheinlichkeit für das Überschreiten eines bestimmten Wertes von x (χ^2) dar, d. h. $\Pr[\chi^2 > \chi_\alpha^2] = \alpha$.

Für das vorliegende Beispiel gilt α = 0,05, γ = 24 und α = 0,025. Die Lösung der oben dargestellten Gleichung lautet $\chi^2_{n-1,\alpha/2} = \chi^2_{24,0.025}$ = 39.3640770266.

Der Wert $\chi^2_{n-1,\alpha/2} = \chi^2_{24,0.975}$ wird hingegen anhand der Werte γ = 24 und α = 0,975 berechnet. Das Ergebnis lautet $\chi^2_{n-1,1-\alpha/2} = \chi^2_{24,0.975}$ = 12.4011502175.

Die oberen und unteren Grenzen des Vertrauensbereichs lauten wie folgt (führen Sie diese Berechnungen im ALG-Modus aus):

$$(n-1)\cdot S^2/\ \chi^2_{n-1,\alpha/2} = (25-1)\cdot 12{,}5/39{,}3640770266 = 7{,}62116179676$$

$$(n-1)\cdot S^2/\ \chi^2_{n-1,1-\alpha/2} = (25-1)\cdot 12{,}5/12{,}4011502175 = 24{,}1913044144$$

Der Vertrauensbereich von 95 % lautet für dieses Beispiel somit

$$7{,}62116179676 < \sigma^2 < 24{,}1913044144.$$

Hypothesentest

Eine Hypothese ist eine Aussage über eine Grundgesamtheit (beispielsweise über ihren Mittelwert). Die Billigung dieser Aussage beruht auf einer statistischen Überprüfung einer der Grundgesamtheit entnommenen Stichprobe. Der anschließende Vorgang und die anschließende Entscheidungsfindung werden als Hypothesentest bezeichnet.

Der Hypothesentest besteht aus dem Entnehmen einer Zufallsstichprobe aus der Grundgesamtheit und dem Erstellen einer statistischen Hypothese über die Grundgesamtheit. Wenn das postulierte Modell oder die postulierte Theorie durch die Werte nicht gestützt werden, wird die Hypothese verworfen. Wenn die Werte jedoch mit der Hypothese übereinstimmen, wird sie nicht verworfen, aber nicht notwendigerweise übernommen. Dieser Entscheidung ist ein Signifikanzniveau α zugeordnet.

Vorgehensweise beim Testen von Hypothesen

Die Vorgehensweise beim Hypothesentest besteht aus den folgenden sechs Schritten:

1. Geben Sie eine Nullhypothese H_0 an. Dies ist die zu testende Hypothese. Beispiel: H_0: μ_1-μ_2 = 0, d. h. wir nehmen an, dass die Mittelwerte von Grundgesamtheit 1 und Grundgesamtheit 2 identisch sind. Wenn H_0 wahr ist, wird jede Differenz der Mittelwerte Fehlern bei der Zufallsstichprobe zugeschrieben.

2. Geben Sie eine Alternativhypothese H_1 an. Diese kann für das vorliegende Beispiel H_1: μ_1-$\mu_2 \neq 0$ lauten. [Hinweis: Dies ist es, was wir eigentlich testen möchten.]
3. Bestimmen Sie eine Testkenngröße T, oder geben Sie diese an. Im vorliegenden Beispiel beruht T auf der Differenz der Mittelwerte $\bar{X}_1$- $\bar{X}_2$.
4. Verwenden Sie die bekannte (oder vermutete) Verteilung der Testkenngröße T.
5. Definieren Sie anhand des zuvor zugewiesenen Signifikanzniveaus α einen Zurückweisungsbereich (die kritische Region R) für die Testkenngröße.
6. Bestimmen Sie anhand der ermittelten Daten, ob der berechnete Wert der Testkenngröße innerhalb oder außerhalb des kritischen Bereichs liegt. Wenn sich die Testkenngröße innerhalb des kritischen Bereichs befindet, sagen wir, dass die getestete Menge ein Signifikanzniveau von 100α Prozent aufweist.

Hinweise:

1. Für das vorliegende Beispiel ergibt die Alternativhypothese H_1: μ_1-$\mu_2 \neq 0$ einen so genannten zweiseitigen Test. Wenn die Alternativhypothese H_1: μ_1-$\mu_2 > 0$ oder H_1: μ_1-$\mu_2 < 0$ lautet, liegt ein einseitiger Test vor.

2. Die Wahrscheinlichkeit des Zurückweisens der Nullhypothese ist gleich dem Signifikanzniveau, d. h. $\Pr[T\in R|H_0] = \alpha$. Die Notation $\Pr[A|B]$ stellt die bedingte Wahrscheinlichkeit von Ereignis A unter der Voraussetzung des Eintretens von Ereignis B dar.

Fehler beim Hypothesentest

Für Hypothesentests verwenden wir die Begriffe „Fehler vom Typ I" bzw. „Fehler vom Typ II", um Fälle zu definieren, in denen eine wahre Hypothese zurückgewiesen oder eine falsche Hypothese akzeptiert (nicht zurückgewiesen) wird. Es sei T = Wert der Testkenngröße, R = Zurückweisungsbereich, A = Beibehaltungsbereich, sodass $R\cap A = \varnothing$ und $R\cup A = \Omega$, wobei Ω = der Parameterraum für T und $\varnothing$ = die leere Menge ist. Die Wahrscheinlichkeiten für Fehler vom Typ I oder Typ II sind wie folgt definiert:

Zurückweisen einer wahren Hypothese

$Pr[\text{Fehler Typ I}] = Pr[T \in R | H_0] = \alpha$

Nicht Zurückweisen einer falschen Hypothese
$Pr[\text{Fehler Typ II}] = Pr[T \in A | H_1] = \beta$

Betrachten wir nun die Fälle, in denen wir die richtige Entscheidung treffen:

Nicht Zurückweisen einer wahren Hypothese
$Pr[\text{Not(Fehler Typ I)}] = Pr[T \in A | H_0] = 1 - \alpha$

Zurückweisen einer falschen Hypothese
$Pr[\text{Not(Fehler Typ II)}] = Pr\ [T \in R | H_1] = 1 - \beta$

Das Komplement von β wird als Mächtigkeit des Tests der Nullhypothese H_0 gegen die Alternativhypothese H_1 bezeichnet. Anhand der Mächtigkeit eines Tests wird beispielsweise die Mindestgröße einer Stichprobe bestimmt, um die Fehlerwahrscheinlichkeit zu verringern.

Auswählen der Werte von α und β

Ein typischer Wert des Signifikanzniveaus (oder der Wahrscheinlichkeit eines Fehlers vom Typ I) ist $\alpha = 0{,}05$ (d. h. durchschnittlich eine falsche Zurückweisung pro 20 Tests). Wenn ein Fehler vom Typ I ernsthafte Folgen hat, wählen Sie für α kleinere Werte aus, z. B. 0,01 oder sogar 0,001.

Der Wert von β, d. h. die Wahrscheinlichkeit eines Fehlers vom Typ II, hängt von α, der Stichprobengröße n und dem tatsächlichen Wert des getesteten Parameters ab. Daher wird der Wert von β nach dem Ausführen des Hypothesentests bestimmt. Üblicherweise werden Diagramme gezeichnet, die β bzw. die Mächtigkeit des Tests $(1-\beta)$ als Funktion des tatsächlichen Wertes des getesteten Parameters darstellen. Diese Diagramme werden als Operationscharakteristik bzw. Gütefunktion bezeichnet.

Inferenzen in Bezug auf einen einzigen Mittelwert

Zweiseitige Hypothese

Das Problem besteht im Testen der Nullhypothese H_o: $\mu = \mu_o$ gegen die Alternativhypothese H_1: $\mu \neq \mu_o$ bei einer statistischen Sicherheit von $(1-\alpha)100\%$ oder einem Signifikanzniveau α bei einer Stichprobe der Größe n mit einem Mittelwert $\bar{x}$ und einer Standardabweichung s. Dieser Test wird als zweiseitiger Test bezeichnet. Der Test wird in folgenden Schritten ausgeführt:

Zunächst berechnen wir die entsprechende Kenngröße für den Test (t_o oder z_o) wie folgt:

- Wenn n < 30 und die Standardabweichung σ der Grundgesamtheit bekannt ist, verwenden Sie die z-Kenngröße: $z_o = \dfrac{\bar{x} - \mu_o}{\sigma / \sqrt{n}}$
- Wenn n > 30 und σ bekannt ist, verwenden Sie z_o wie oben dargestellt. Wenn σ nicht bekannt ist, ersetzen Sie in z_o σ durch s, d. h. $z_o = \dfrac{\bar{x} - \mu_o}{s / \sqrt{n}}$
- Wenn n < 30 und σ nicht bekannt ist, verwenden Sie die t-Kenngröße $t_o = \dfrac{\bar{x} - \mu_o}{s / \sqrt{n}}$, mit dem Freiheitsgrad $\nu = n - 1$.

Berechnen Sie dann den entweder z_o oder t_o zugeordneten P-Wert (eine Wahrscheinlichkeit) und vergleichen Sie ihn mit α, um zu bestimmen, ob die Nullhypothese zurückgewiesen werden soll. Der P-Wert für einen zweiseitigen Test ist entweder durch

P-Wert = $P(|z|>|z_o|)$ oder durch P-Wert = $P(|t|>|t_o|)$ definiert.

Die beim Hypothesentest zu verwendenden Kriterien lauten:

- H_o zurückweisen, wenn P-Wert < α
- H_o nicht zurückweisen, wenn P-Wert > α

Der P-Wert für einen zweiseitigen Test kann mithilfe der Wahrscheinlichkeitsfunktionen des Taschenrechners wie folgt berechnet werden:

- Bei Verwendung von z: P-Wert = 2·UTPN(0,1,$|z_o|$)
- Bei Verwendung von t: P-Wert = 2·UTPT(ν,$|t_o|$)

Beispiel 1 – Testen Sie die Nullhypothese H_o: μ = 22,5 (= μ_o) gegen die Alternativhypothese H_1: $\mu \neq 22{,}5$ bei einer statistischen Sicherheit von 95 %, d. h. α = 0,05, und verwenden Sie hierfür eine Stichprobe der Größe n = 25 mit dem Mittelwert $\bar{x}$ = 22,0 und der Standardabweichung s = 3,5. Wir setzen voraus, dass wir den Wert der Standardabweichung für die Grundgesamtheit nicht kennen. Daher berechnen wir die t-Kenngröße wie folgt:

$$t_o = \frac{\bar{x}-\mu_o}{s/\sqrt{n}} = \frac{22.0-22.5}{3.5/\sqrt{25}} = -0.7142$$

Der entsprechende P-Wert für den Freiheitsgrad ν = 25 - 1 = 24 lautet

P-Wert = 2·UTPT(24,-0.7142) = 2·0.7590 = 1.5169,

Da 1,5169 > 0,05, d. h. P-Wert > α, können wir die Nullhypothese H_o: μ = 22,0 nicht zurückweisen.

Einseitige Hypothese

Das Problem besteht im Testen der Nullhypothese H_o: $\mu = \mu_o$ gegen die Alternativhypothese H_1: $\mu > \mu_o$ oder H_1: $\mu < \mu_o$ bei einer statistischen Sicherheit von (1-α)100% oder einem Signifikanzniveau α anhand einer Stichprobe der Größe n mit einem Mittelwert x und einer Standardabweichung s. Dieser Test wird als einseitiger Test bezeichnet. Die Ausführung eines einseitigen Tests beginnt wie der zweiseitige Test und wie oben dargestellt mit der Berechnung der entsprechenden Kenngröße für den Test (t_o oder z_o).

Anschließend berechnen wir den entweder z_o oder t_o zugeordneten P-Wert und vergleichen diesen mit α, um zu bestimmen, ob die Nullhypothese zurückgewiesen werden soll. Der P-Wert für einen zweiseitigen Test ist entweder definiert durch

$$\text{P-Wert} = P(z > |z_o|) \text{ oder durch P-Wert} = P(t > |t_o|).$$

Die beim Hypothesentest zu verwendenden Kriterien lauten:

- H_o zurückweisen, wenn P-Wert < α
- H_o nicht zurückweisen, wenn P-Wert > α

Beachten Sie, dass es sich um dieselben Kriterien wie beim zweiseitigen Test handelt. Der Hauptunterschied liegt in der Art der Berechnung des P-Wertes. Der P-Wert für einen einseitigen Test kann mithilfe der Wahrscheinlichkeitsfunktionen des Taschenrechners wie folgt berechnet werden:

- Bei Verwendung von z: P-Wert = UTPN(0,1,z_o)
- Bei Verwendung von t : P-Wert = UTPT(ν,t_o)

<u>Beispiel 2</u> – Testen Sie die Nullhypothese H_o: μ = 22,0 (= μ_o) gegen die Alternativhypothese H_1: μ >22,5 bei einer statistischen Sicherheit von 95 %, d. h. α = 0,05, und verwenden Sie hierfür eine Stichprobe der Größe n = 25 mit dem Mittelwert $\bar{x}$ = 22,0 und der Standardabweichung s = 3,5. Wir gehen wieder davon aus, dass wir den Wert der Grundgesamtheitsstandardabweichung nicht kennen. Daher ist der Wert der t-Kenngröße mit dem entsprechenden Wert des oben dargestellten zweiseitigen Tests identisch, d. h. t_o = -0,742 und der P-Wert für Freiheitsgrad ν = 25 - 1 = 24 lautet

P-Wert = UTPT(24; |-0,7142|) = UTPT(24;0,7124) = 0,2409.

Da 0,2409 > 0,05, d. h. P-Wert > α, können wir die Nullhypothese H_o: μ = 22,0 nicht zurückweisen.

Inferenzen in Bezug auf zwei Mittelwerte

Die zu testende Nullhypothese lautet H_o: μ_1-μ_2 = δ bei einer statistischen Sicherheit von (1-α)100% oder dem Signifikanzniveau α und Verwendung zweier Stichproben mit den Größen n_1 und n_2, den Mittelwerten $\bar{x}_1$ und $\bar{x}_2$ sowie den Standardabweichungen s_1 und s_2. Wenn die den Stichproben entsprechenden Grundgesamtheitsstandardabweichungen σ_1 und σ_2 bekannt sind oder wenn $n_1 > 30$ und $n_2 > 30$ (große Stichproben), lautet die zu verwendende Testkenngröße

$$z_o = \frac{(\bar{x}_1 - \bar{x}_2) - \delta}{\sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}}}$$

Wenn $n_1 < 30$ oder $n_2 < 30$ (mindestens eine kleine Stichprobe), verwenden Sie folgende Testkenngröße:

$$t = \frac{(\bar{x}_1 - \bar{x}_2) - \delta}{\sqrt{(n_1 - 1)s_1^2 + (n_2 - 1)s_2^2}} \sqrt{\frac{n_1 n_2 (n_1 + n_2 - 2)}{n_1 + n_2}}$$

Zweiseitige Hypothese

Wenn die Alternativhypothese eine zweiseitige Hypothese ist, d. h. H_1: μ_1-μ_2 $\neq \delta$, wird der P-Wert für diesen Test wie folgt berechnet:

- Bei Verwendung von z: P-Wert = 2·UTPN(0,1,|z_o|)
- Bei Verwendung von t : P-Wert = 2·UTPT(ν,|t_o|)

Hierbei wird der Freiheitsgrad der t-Verteilung durch $\nu = n_1 + n_2 - 2$ bestimmt. Die Testkriterien lauten

- H_o zurückweisen, wenn P-Wert < α
- H_o nicht zurückweisen, wenn P-Wert > α

Einseitige Hypothese

Wenn die Alternativhypothese eine zweiseitige Hypothese ist, d. h. H_1: μ_1-μ_2 $< \delta$ oder H_1: μ_1-μ_2 $< \delta$, wird der P-Wert für diesen Test wie folgt berechnet:

- Bei Verwendung von z: P-Wert = UTPN(0,1, |z_o|)
- Bei Verwendung von t : P-Wert = UTPT(ν,|t_o|)

Die beim Hypothesentest zu verwendenden Kriterien lauten:

- H_o zurückweisen, wenn P-Wert < α
- H_o nicht zurückweisen, wenn P-Wert > α

Tests mit paarigen Stichproben

Wenn zwei Stichproben der Größe n mit paarigen Datenpunkten vorhanden sind, müssen wir diesen Fall wie eine einzige Stichprobe der Differenzen der paarigen Werte behandeln, statt die Nullhypothese H: μ_1-μ_2 = δ unter Verwendung der Mittelwerte und Standardabweichungen der beiden Stichproben zu testen. Mit anderen Worten, generieren Sie eine neue Zufallsvariable $X = X_1$-X_2, und testen sie H_o: $\mu = \delta$, wobei μ den Mittelwert der Grundgesamtheit für X darstellt. Sie müssen daher $\bar{x}$ und s für die Stichprobe der Werte von x ermitteln. Der Test wird dann mit den bereits beschriebenen Methoden als Test mit einer einzigen Stichprobe fortgesetzt.

Inferenzen in Bezug auf eine einzige Quote

Angenommen wir möchten die Nullhypothese H_0: $p = p_0$ testen, wobei p die Wahrscheinlichkeit eines erfolgreichen Ergebnisses bei einer beliebigen Wiederholung des Bernoulli-Versuchs darstellt. Zum Testen der Hypothese führen wir n Wiederholungen des Experiments durch und ermitteln, dass k erfolgreiche Ergebnisse aufgezeichnet werden. Somit wird durch p' = k/n ein Schätzwert von p angegeben.
Die Varianz der Abweichung wird mit $s_p^2 = p'(1-p')/n = k\cdot(n-k)/n^3$ geschätzt.

Angenommen der Wert $Z = (p-p_0)/s_p$, entspricht der Standardnormalverteilung, d. h. Z ~ N(0,1). Der Wert der zu testenden Kenngröße lautet $z_0 = (p'-p_0)/s_p$.

Statt anhand des P-Wertes zu bestimmen, ob die Hypothese beibehalten wird, verwenden wir den Vergleich zwischen dem kritischen Wert von z0 und dem α oder $\alpha/2$ entsprechenden Wert von z.

<u>Zweiseitiger Test</u>

Bei Verwendung eines zweiseitigen Tests ermitteln wir den Wert von $z_{\alpha/2}$, mit

$$\Pr[Z> z_{\alpha/2}] = 1-\Phi(z_{\alpha/2}) = \alpha/2 \text{ oder } \Phi(z_{\alpha/2}) = 1- \alpha/2,$$

wobei $\Phi(z)$ die Summenverteilungsfunktion der Standardnormalverteilung darstellt (siehe Kapitel 17).

Weisen Sie die Nullhypothese H_0 zurück, wenn $z_0 > z_{\alpha/2}$ oder wenn $z_0 < - z_{\alpha/2}$.

Mit anderen Worten, der Zurückweisungsbereich ist $R = \{ |z_0| > z_{\alpha/2} \}$ und der Beibehaltungsbereich ist $A = \{|z_0| < z_{\alpha/2} \}$.

Einseitiger Test

Bei Verwendung eines einseitigen Tests ermitteln wir den Wert von S mit

$$\Pr[Z> z_\alpha] = 1-\Phi(z_\alpha) = \alpha \text{ oder } \Phi(z_\alpha) = 1- \alpha.$$

Weisen Sie die Nullhypothese H_0 zurück, wenn $z_0 > z_\alpha$ und H_1: $p>p_0$ oder wenn $z_0 < - z_\alpha$ und H_1: $p<p_0$.

Testen der Differenz zweier Quoten

Angenommen wir möchten die Nullhypothese H_0: $p_1-p_2 = p_0$ testen, wobei p für die beiden Grundgesamtheiten 1 und 2 die Wahrscheinlichkeit eines erfolgreichen Ergebnisses einer beliebigen Wiederholung des Bernoulli-Versuchs darstellt. Zum Testen der Hypothese führen wir für Grundgesamtheit 1 n Wiederholungen des Experiments durch und ermitteln, dass k_1 erfolgreiche Ergebnisse aufgezeichnet werden. Außerdem ermitteln wir für n_2 Versuche in Stichprobe 2 k_2 erfolgreiche Ergebnisse. Die Schätzwerte p_1 und p_2 sind somit durch $p_1' = k_1/n_1$ bzw. $p_2' = k_2/n_2$ definiert.

Die Varianzen für die Stichproben werden geschätzt als

$$s_1^2 = p_1'(1-p_1')/n_1 = k_1\cdot(n_1-k_1)/n_1^3 \text{ bzw. } s_2^2 = p_2'(1-p_2')/n_2 = k_2\cdot(n_2-k_2)/n_2^3.$$

Die Varianz der Quotendifferenz wird mit $s_p^2 = s_1^2 + s_2^2$ geschätzt.

Angenommen der Wert $Z = (p_1-p_2-p_0)/s_p$, entspricht der Standardnormalverteilung, d. h. $Z \sim N(0,1)$. Der Wert der zu testenden Kenngröße lautet $z_0 = (p_1'-p_2'-p_0)/s_p$.

Zweiseitiger Test

Bei Verwendung eines zweiseitigen Tests ermitteln wir den Wert von $z_{\alpha/2}$ mit

$$\Pr[Z> z_{\alpha/2}] = 1-\Phi(z_{\alpha/2}) = \alpha/2 \text{ oder } \Phi(z_{\alpha/2}) = 1- \alpha/2,$$

wobei $\Phi(z)$ die Summenverteilungsfunktion der Standardnormalverteilung darstellt.

Weisen Sie die Nullhypothese H_0 zurück, wenn $z_0 >z_{\alpha/2}$ oder wenn $z_0 < - z_{\alpha/2}$.

Mit anderen Worten, der Zurückweisungsbereich ist R = { $|z_0| > z_{\alpha/2}$ } und der Beibehaltungsbereich ist A = {$|z_0| < z_{\alpha/2}$ }.

Einseitiger Test

Bei Verwendung eines einseitigen Tests ermitteln wir den Wert von z_α mit

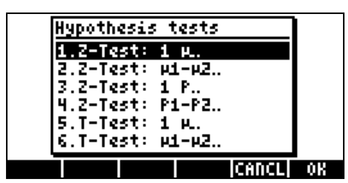
$$\Pr[Z> z_\alpha] = 1-\Phi(z_\alpha) = \alpha \text{ oder } \Phi(z_\alpha) = 1- \alpha.$$

Weisen Sie die Nullhypothese H_0 zurück, wenn $z_0 >z_\alpha$ und H_1: $p_1-p_2 > p_0$ oder wenn $z_0 < - z_\alpha$ und H_1: $p_1-p_2 <p_0$.

Hypothesentest mit vorprogrammierten Funktionen

Der Taschenrechner enthält unter *5. Hypoth. Tests..* Hypothesentestprozeduren, die mit → STAT ▲ ▲ OK aufgerufen werden können.

Wie bei der bereits erläuterten Berechnung von Vertrauensbereichen bietet dieses Programm die folgenden 6 Optionen:



Diese Optionen werden entsprechend den Anwendungen für den Vertrauensbereich erläutert:

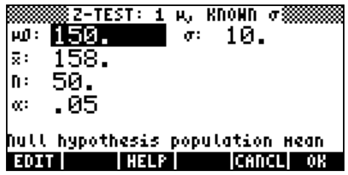
1. Z-Test: 1 μ: Hypothesentest einer einzelnen Stichprobe für den Mittelwert der Grundgesamtheit μ mit bekannter Varianz der Grundgesamtheit oder bei großen Stichproben mit unbekannter Varianz der Grundgesamtheit.
2. Z-Test: $\mu 1 - \mu 2$: Hypothesentest für die Differenz der Mittelwerte der Grundgesamtheit μ_1- μ_2 mit entweder bekannten Varianzen der Grundgesamtheit oder bei großen Stichproben mit unbekannten Varianzen der Grundgesamtheit.
3. Z-Test: 1 p: Hypothesentest einer einzelnen Stichprobe für die Quote p bei großen Stichproben mit unbekannter Varianz der Grundgesamtheit.
4. Z-Test: p1– p2: Hypothesentest für die Differenz zweier Quoten p_1-p_2, bei großen Stichproben mit unbekannten Varianzen der Grundgesamtheit.
5. T-Test: 1 μ: Hypothesentest einer einzelnen Stichprobe für den Mittelwert der Grundgesamtheit μ bei kleinen Stichproben mit unbekannter Varianz der Grundgesamtheit.
6. T-Test: $\mu 1 - \mu 2$: Hypothesentest für die Differenz zweier Mittelwerte der Grundgesamtheit μ_1-μ_2 bei kleinen Stichproben mit unbekannten Varianzen der Grundgesamtheit.

Führen Sie die folgenden Übungen aus:

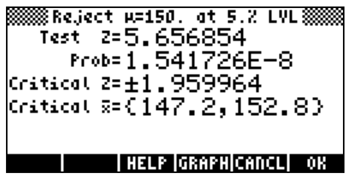
Beispiel 1 – Testen Sie für $\mu_0 = 150$, $\sigma = 10$, $\bar{x} = 158$, n = 50 und $\alpha = 0.05$ die Hypothese H_0: $\mu = \mu_o$ gegen die Alternativhypothese H_1: $\mu \neq \mu_0$.

Drücken Sie ↱ STAT ▲ ▲ OK, um die Hypothesentestfunktion des Taschenrechners aufzurufen. Drücken Sie OK, um Option 1. Z-Test: 1 μ auszuwählen.

Geben Sie die folgenden Daten ein und drücken Sie OK:

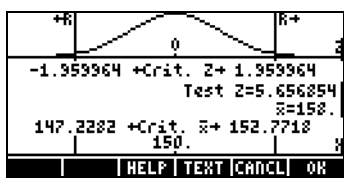


Anschließend werden Sie aufgefordert, die Alternativhypothese auszuwählen. Wählen Sie μ ≠150 aus und drücken Sie OK. Das Ergebnis lautet:



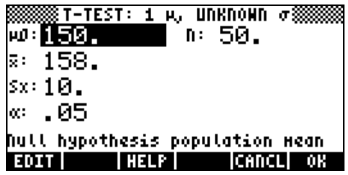
Anschließend weisen wir H_0: $\mu = 150$ gegen H_1: $\mu \neq 150$ zurück. Der Testwert z lautet z_0 = 5,656854. Der P-Wert lautet $1{,}54\times10^{-8}$. Die kritischen Werte $\pm z_{\alpha/2}$ = ±1,959964 entsprechen dem kritischen Bereich x {147,2 152,8}.

Diese Informationen können durch Drücken der Menütaste GRAPH grafisch dargestellt werden:

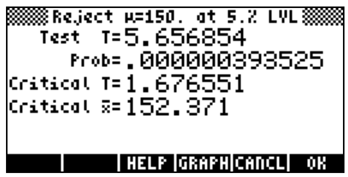


Beispiel 2 – Testen Sie für μ_0 = 150, $\bar{x}$ = 158, s = 10, n = 50 und α = 0,05, die Hypothese H_0: $\mu = \mu_o$ gegen die Alternativhypothese H_1: $\mu > \mu_0$. Die Standardabweichung der Grundgesamtheit σ ist nicht bekannt.

Drücken Sie ‹STAT ▲▲ OK, um die Hypothesentestfunktion des Taschenrechners aufzurufen. Drücken Sie ▲▲ OK, um Option 5. T-Test: 1 μ aufzurufen.
Geben Sie die folgenden Daten ein und drücken Sie OK:

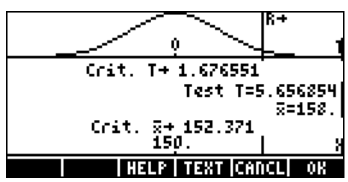


Wählen Sie die Alternativhypothese H_1: $\mu > 150$ aus und drücken Sie OK. Das Ergebnis lautet:



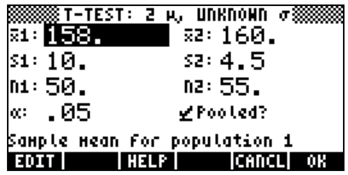
Wir weisen die Nullhypothese H_0: $\mu_0 = 150$ gegen die Alternativhypothese H_1: $\mu > 150$ zurück. Der Testwert lautet $t_0 = 5{,}656854$ mit P-Wert = 0,000000393525. Der kritische Wert von t lautet $t_\alpha = 1{,}676551$ und entspricht dem kritischen Wert $\bar{x} = 152{,}371$.

Drücken Sie GRAPH, um die Ergebnisse wie folgt grafisch darzustellen:

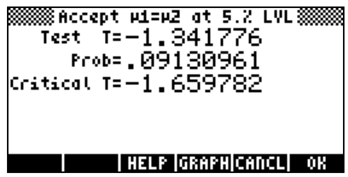


<u>Beispiel 3</u> – Aufgrund der Daten zweier Stichproben gilt $\bar{x}_1 = 158$, $\bar{x}_1 = 160$, $s_1 = 10$, $s_2 = 4{,}5$, n1 = 50 und $n_2 = 55$. Testen Sie für $\alpha = 0{,}05$ und eine „zusammengefasste“ Varianz die Hypothese H_0: $\mu_1 - \mu_2 = 0$ gegen die Alternativhypothese H_1: $\mu_1 - \mu_2 < 0$.

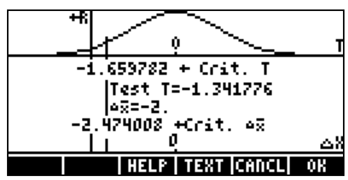
Drücken Sie ↱ STAT ▲▲ OK, um die Hypothesentestfunktion des Taschenrechners aufzurufen. Drücken Sie ▲ OK, um Option 6. T-Test: $\mu 1 - \mu 2$ auszuwählen. Geben Sie die folgenden Daten ein und drücken Sie OK:



Wählen Sie die Alternativhypothese $\mu 1 < \mu 2$ aus und drücken Sie OK. Das Ergebnis lautet



Somit behalten wir die Hypothese $\mu_1-\mu_2 = 0$ oder H_0: $\mu_1=\mu_2$ gegen die Alternativhypothese H_1: $\mu_1-\mu_2 < 0$ oder H_1: $\mu_1=\mu_2$ bei. Der Testwert t lautet t_0 = -1,341776 mit dem P-Wert = 0,09130961 und der kritische Wert t lautet $-t_\alpha$ = -1, 659782. Die Ergebnisse werden wie folgt grafisch dargestellt:



Diese drei Beispiele sollten für das Verständnis der vorprogrammierten Hypothesentestfunktion des Taschenrechners ausreichen.

Inferenzen in Bezug auf eine einzige Varianz

Die zu testende Nullhypothese lautet H_o: $\sigma^2 = \sigma_o^2$ bei einer statistischen Sicherheit von $(1-\alpha)100\%$ oder dem Signifikanzniveau α sowie der Verwendung einer Stichprobengröße n und der Varianz s^2. Die zu verwendende Testkenngröße ist eine chi^2-Testkenngröße, die wie folgt definiert ist:

$$\chi_o^2 = \frac{(n-1)s^2}{\sigma_0^2}$$

Je nach der ausgewählten Alternativhypothese wird der P-Wert wie folgt berechnet:

- H_1: $\sigma^2 < \sigma_o^2$ P-Wert = $P(\chi^2<\chi_o^2)$ = 1-UTPC(ν,χ_o^2)
- H_1: $\sigma^2 > \sigma_o^2$ P-Wert = $P(\chi^2>\chi_o^2)$ = UTPC(ν,χ_o^2)
- H_1: $\sigma^2 \neq \sigma_o^2$ P-Wert =2·min[$P(\chi^2<\chi_o^2)$, $P(\chi^2>\chi_o^2)$] = 2·min[1-UTPC(ν,χ_o^2), UTPC(ν,χ_o^2)]

Hierbei erzeugt die Funktion min[x,y] den Minimalwert von x bzw. y (entsprechend erzeugt max[x,y] den Maximalwert von x bzw. y). UTPC(ν,x) stellt die oberen Wahrscheinlichkeiten des Taschenrechners für den Freiheitsgrad ν = n - 1 dar.

Die Testkriterien sind mit den für den Hypothesentest der Mittelwerte verwendeten Testkriterien identisch, also

- H_o zurückweisen, wenn P-Wert < α
- H_o nicht zurückweisen, wenn P-Wert > α

Beachten Sie, dass dieses Verfahren nur zulässig ist, wenn es sich bei der Grundgesamtheit, aus der die Stichprobe entnommen wurde, um eine Normalgrundgesamtheit handelt.

Besipiel 1 – Gegeben sei σ_o^2 = 25, α = 0.05, n = 25 und s^2 = 20, und die Stichprobe sei einer Normalgrundgesamtheit entnommen. Zum Testen der Hypothese H_o: $\sigma^2 = \sigma_o^2$ gegen H_1: $\sigma^2 < \sigma_o^2$ berechnen wir zunächst

$$\chi_o^2 = \frac{(n-1)s^2}{\sigma_0^2} = \frac{(25-1)\cdot 20}{25} = 189.2$$

Mit dem Freiheitsgrad ν = n - 1 = 25 - 1 = 24 berechnen wir den P-Wert als

P-Wert = $P(\chi^2<19.2)$ = 1-UTPC(24,19.2) = 0,2587…

Da 0,2587… > 0,05, d. h. P-Wert > α, können wir die Nullhypothese H_o: $\sigma^2 = 25 (= \sigma_o^2)$ nicht zurückweisen.

Inferenzen in Bezug auf zwei Varianzen

Die zu testende Nullhypothese lautet H_o: $\sigma_1^2 = \sigma_2^2$ bei einer statistischen Sicherheit von $(1-\alpha)100\%$ oder dem Signifikanzniveau α sowie der Verwendung zweier Stichproben mit den Größen n_1 und n_2 und den Varianzen s_1^2 und s_2^2. Die zu verwendende Testkenngröße ist die Testkenngröße F, die wie folgt definiert ist:

$$F_o = \frac{s_N^2}{s_D^2}$$

Hierbei stellen s_N^2 und s_D^2 Zähler und Nenner der Kenngröße F dar. Die Auswahl von Zähler und Nenner hängt von der getesteten Alternativhypothese ab, wie unten dargestellt. Die entsprechende Verteilung für F weist die Freiheitsgrade $\nu_N = n_N-1$ und $\nu_D = n_D-1$ auf, wobei n_N und n_D die den Varianzen s_N^2 bzw. s_D^2 entsprechenden Stichprobengrößen darstellen.

In der folgenden Tabelle wird die Auswahl von Zähler und Nenner für F_o je nach der ausgewählten Alternativhypothese dargestellt:

Alternativ-hypothese	*Test-kenngröße*	*Freiheits-grad*
H_1: $\sigma_1^2 < \sigma_2^2$ (einseitig)	$F_o = s_2^2/s_1^2$	$\nu_N = n_2-1$, $\nu_D = n_1-1$
H_1: $\sigma_1^2 > \sigma_2^2$ (einseitig)	$F_o = s_1^2/s_2^2$	$\nu_N = n_1-1$, $\nu_D = n_2-1$
H_1: $\sigma_1^2 \neq \sigma_2^2$ (zweiseitig)	$F_o = s_M^2/s_m^2$	$\nu_N = n_M-1, \nu_D = n_m-1$
	$s_M^2=\max(s_1^2,s_2^2)$, $s_m^2=\min(s_1^2,s_2^2)$	

(*) n_M ist der s_M entsprechende Wert von n, und n_m ist der s_m entsprechende Wert von n.

Der P-Wert wird in sämtlichen Fällen wie folgt berechnet: P-Wert = $P(F>F_o)$ = $UTPF(\nu_N, \nu_D,F_o)$

Die Testkriterien lauten:

- H_o zurückweisen, wenn P-Wert < α
- H_o nicht zurückweisen, wenn P-Wert > α

Beispiel 1 – Gegeben seien zwei Normalgrundgesamtheiten entnommene Stichproben, sodass $n_1 = 21$, $n_2 = 31$, $s_1^2 = 0{,}36$ und $s_2^2 = 0{,}25$. Wir testen die Nullhypothese H_o: $\sigma_1^2 = \sigma_2^2$ bei Signifikanzniveau $\alpha = 0{,}05$ gegen die Alternativhypothese H_1: $\sigma_1^2 \neq \sigma_2^2$. Für eine beidseitige Hypothese müssen wir s_M und s_m, wie folgt bestimmen:

$$s_M^2=\max(s_1^2,s_2^2) = \max(0{,}36;0.25) = 0{,}36 = s_1^2$$
$$s_m^2 = \min(s_1^2,s_2^2) = \min(0{,}36;0{,}25) = 0{,}25 = s_2^2$$

Außerdem gilt

$$n_M = n_1 = 21,$$
$$n_m = n_2 = 31,$$
$$\nu_N = n_M - 1= 21-1=20,$$
$$\nu_D = n_m -1 = 31-1 =30.$$

Die Testkenngröße F lautet daher $F_o = s_M^2/s_m^2=0{,}36/0{,}25=1{,}44$.

Der P-Wert lautet P-Wert = $P(F>F_o)$ = $P(F>1.44)$ = $UTPF(\nu_N, \nu_D,F_o)$ = $UTPF(20;30;1{,}44) = 0{,}1788\ldots$

Da $0{,}1788\ldots > 0{,}05$, d. h. P-Wert > α, können wir die Nullhypothese H_o: $\sigma_1^2 = \sigma_2^2$ nicht zurückweisen.

Weitere Anmerkungen zur linearen Regression

In diesem Kapitel werden die weiter oben in diesem Kapitel dargestellten Konzepte der linearen Regression weiter ausgearbeitet und ein Verfahren für den Hypothesentest von Regressionsparametern vorgestellt.

Die Methode der kleinsten Quadrate

x = eine unabhängige nichtzufällige Variable und Y = eine abhängige Zufallsvariable. Die Regressionskurve von Y auf x ist als die Beziehung zwischen x und dem Mittelwert der entsprechenden Verteilung der Werte von Y's definiert.

Die Regressionskurve von Y auf x sei linear, d. h. die Mittelwertverteilung der Werte von Y ist durch A + Bx definiert. Y unterscheidet sich vom Mittelwert (A + B·x) durch den Wert ε, sodass Y = A + B·x + ε, wobei ε eine Zufallsvariable ist.
Zeichnen Sie ein Streuungsdiagramm oder eine Punktwolke, um visuell zu überprüfen, ob die Daten einem linearen Trend entsprechen.

Für n paarige Werte (x_i, y_i) sei y durch
^y = a + b·x definiert, wobei a und b konstant sind.

Der Prognosefehler sei als $e_i = y_i - \hat{y}_i = y_i - (a + b \cdot x_i)$ definiert.

Die Methode der kleinsten Quadrate erfordert, dass wir a und b so auswählen, dass die Summe quadratischer Fehler (SSE, Sum of Squared Errors) minimiert wird:

$$SSE = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} [y_i - (a + bx_i)]^2$$

Die Bedingungen lauten:

$$\frac{\partial}{\partial a}(SSE) = 0 \qquad \frac{\partial}{\partial b}(SSE) = 0$$

Wir erhalten die so genannten Normalgleichungen:

$$\sum_{i=1}^{n} y_i = a \cdot n + b \cdot \sum_{i=1}^{n} x_i$$

$$\sum_{i=1}^{n} x_i \cdot y_i = a \cdot \sum_{i=1}^{n} x_i + b \cdot \sum_{i=1}^{n} x_i^2$$

Hierbei handelt es sich um ein lineares Gleichungssystem mit den Unbekannten a und b, das mit den Taschenrechnerfunktionen für lineare Gleichungen gelöst werden kann. Es ist jedoch nicht erforderlich, sich mit diesen Berechnungen zu beschäftigen, da Sie die weiter oben dargestellte Option **3. Fit Data..** im Menü ↱ STAT verwenden können.

Hinweise:

- a,b sind erwartungstreue Schätzfunktionen von A, B.
- Das Gauß-Markov-Theorem besagt, dass unter allen erwartungstreuen Schätzfunktionen für A und B die Schätzfunktionen mit den kleinsten Quadraten (a,b) am effizientesten sind.

Weitere Gleichungen für lineare Regression

Die Summenkenngrößen, z. B. Σx, Σx² usw., können zum Definieren der folgenden Größen verwendet werden:

$$S_{xx} = \sum_{i=1}^{n}(x_i - \overline{x})^2 = (n-1)\cdot s_x^2 = \sum_{i=1}^{n} x_i^{\,2} - \frac{1}{n}\left(\sum_{i=1}^{n} x_i\right)$$

$$S_y = \sum_{i=1}^{n}(y_i - \overline{y})^2 = (n-1)\cdot s_y^2 = \sum_{i=1}^{n} y_i^{\,2} - \frac{1}{n}\left(\sum_{i=1}^{n} y_i\right)^2$$

$$S_{xy} = \sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})^2 = (n-1)\cdot s_{xy} = \sum_{i=1}^{n} x_i y_i - \frac{1}{n}\left(\sum_{i=1}^{n} x_i\right)\left(\sum_{i=1}^{n} y_i\right)$$

Hieraus folgt, dass die Standardabweichungen von x und y sowie die Kovarianz von x,y durch

$s_x = \sqrt{\frac{S_{xx}}{n-1}}$, $s_y = \sqrt{\frac{S_{yy}}{n-1}}$ bzw. $s_{xy} = \frac{S_{yx}}{n-1}$ definiert sind.

Der Stichprobenkorrelationskoeffizient lautet $r_{xy} = \frac{S_{xy}}{\sqrt{S_{xx}\cdot S_{yy}}}$.

Für $\bar{x}$, $\bar{y}$, S_{xx}, S_{yy} und S_{xy} lautet die Lösung der Normalgleichungen

$$a = \bar{y} - b\bar{x} \quad , \qquad b = \frac{S_{xy}}{S_{xx}} = \frac{s_{xy}}{s_x^2}$$

Prognosefehler

Die Regressionskurve von Y auf x ist durch $Y = A + B \cdot x + \varepsilon$ definiert. Bei einer Menge von n Datenpunkten (x_i, y_i) gilt $Y_i = A + B \cdot x_i + \varepsilon_I$, $(i = 1,2,\ldots,n)$, mit Y_i = unabhängige normalverteilte Zufallsvariablen mit dem Mittelwert $(A + B \cdot x_i)$ und der gemeinsamen Varianz σ^2 sowie ε_i = unabhängige normalverteilte Zufallsvariablen mit dem Mittelwert Null und der gemeinsamen Varianz σ^2.

Es sei y_i = tatsächlicher Datenwert, $\hat{y}_i = a + b \cdot x_i$ = Datenprognose der kleinsten Quadrate. Der Prognosefehler lautet dann: $e_i = y_i - \hat{y}_i = y_i - (a + b \cdot x_i)$.

Ein Schätzwert von σ^2 ist der so genannte Standardfehler eines Schätzwertes

$$s_e^2 = \frac{1}{n-2}\sum_{i=1}^{n}[y_i - (a + bx_i)]^2 = \frac{S_{yy} - (S_{xy})^2 / S_{xx}}{n-2} = \frac{n-1}{n-2} \cdot s_y^2 \cdot (1 - r_{xy}^2)$$

Vertrauensbereiche und Hypothesentest bei linearer Regression

Im Folgenden sind einige Konzepte und Gleichungen für die statistische Inferenz bei linearer Regression aufgeführt:

- Vertrauensgrenzen für Regressionskoeffizienten:
  Für die Steigung (B): $b - (t_{n-2,\alpha/2}) \cdot s_e / \sqrt{S_{xx}} < B < b + (t_{n-2,\alpha/2}) \cdot s_e / \sqrt{S_{xx}}$,
  Für den Abschnitt (A):
  $a - (t_{n-2,\alpha/2}) \cdot s_e \cdot [(1/n) + \bar{x}^2 / S_{xx}]^{1/2} < A <$
  $a + (t_{n-2,\alpha/2}) \cdot s_e \cdot [(1/n) + \bar{x}^2 / S_{xx}]^{1/2}$,
  wobei t der Studentschen t-Verteilung mit dem Freiheitsgrad $\nu = n - 2$ entspricht und n die Anzahl der Datenpunkte in der Stichprobe darstellt.

- Hypothesentest für die Steigung B:

Die Nullhypothese H_0: $B = B_0$, getestet gegen die Alternativhypothese H_1: $B \neq B_0$. Die Testkenngröße lautet $t_0 = (b - B_0)/(s_e/\sqrt{S_{xx}})$, wobei t der Studentschen t-Verteilung mit dem Freiheitsgrad $\nu = n - 2$ entspricht und n die Anzahl der Datenpunkte in der Stichprobe darstellt. Der Test wird wie ein Mittelwert-Hypothesentest ausgeführt, d. h, bei gegebenem Signifikanzniveau α wird der kritische Wert von t , $t_{\alpha/2}$ bestimmt und anschließend H_0 zurückgewiesen, wenn $t_0 > t_{\alpha/2}$ oder wenn $t_0 < - t_{\alpha/2}$.

Wenn Sie den Test für den Wert $B_0 = 0$ ausführen und der Test ergibt, dass die Nullhypothese H_0: $B = 0$ nicht zurückgewiesen werden kann, ist die Gültigkeit der linearen Regression zweifelhaft. Mit anderen Worten, die Aussage $B \neq 0$ wird durch die Stichprobendaten nicht bestätigt. Es handelt sich daher um einen Test der Signifikanz des Regressionsmodells.

- Hypothesentest für die Steigung A:
  Die Nullhypothese H_0: $A = A_0$, getestet gegen die Alternativhypothese H_1: $A \neq A_0$. Die Testkenngröße lautet $t_0 = (a-A_0)/[(1/n)+ \bar{x}^2/S_{xx}]^{1/2}$, wobei t der Studentschen t-Verteilung mit dem Freiheitsgrad $\nu = n - 2$ entspricht und n die Anzahl der Datenpunkte in der Stichprobe darstellt. Der Test wird wie ein Mittelwert-Hypothesentest ausgeführt, d. h, bei gegebenem Signifikanzniveau α wird der kritische Wert von $t = t_{\alpha/2}$ bestimmt und anschließend H_0 zurückgewiesen, wenn $t_0 > t_{\alpha/2}$ oder wenn $t_0 < - t_{\alpha/2}$.

- Vertrauensbereich für den Mittelwert von Y bei $x = x_0$, i.e., $\alpha+\beta x_0$:
  $a+b\cdot x-(t_{n-2,\alpha/2})\cdot s_e\cdot[(1/n)+(x_0- \bar{x})^2/S_{xx}]^{1/2} < \alpha+\beta x_0 <$
  $a+b\cdot x+(t_{n-2, \alpha /2})\cdot s_e\cdot[(1/n)+(x_0- \bar{x})^2/S_{xx}]^{1/2}$.

- Prognosegrenzen: Vertrauensbereich für den Prognosewert $Y_0=Y(x_0)$:
  $a+b\cdot x-(t_{n-2,\alpha/2})\cdot s_e\cdot[1+(1/n)+(x_0- \bar{x})^2/S_{xx}]^{1/2} < Y_0 <$
  $a+b\cdot x+(t_{n-2, \alpha /2})\cdot s_e\cdot[1+(1/n)+(x_0- \bar{x})^2/S_{xx}]^{1/2}$.

Vorgehensweise mit dem Taschenrechner bei Inferenzkenngrößen für lineare Regression

1) Geben Sie (x,y) als Datenspalten in die Statistikmatrix ΣDAT ein.

2) Erzeugen Sie für die entsprechenden Spalten von ΣDAT eine Punktwolke und überprüfen Sie den linearen Verlauf anhand der entsprechenden Anzeige von H-VIEW und V-VIEW.
3) Verwenden Sie für die Datenanpassung als gerade Linie ↱ STAT ▼ ▼ OK, und ermitteln Sie a, b, s_{xy} (Kovarianz) sowie r_{xy} (Korrelation).
4) Ermitteln Sie $\overline{x}$, $\overline{y}$, s_x, s_y mit ↱ STAT ▼ OK,. In Spalte 1 werden die Kenngrößen für x und in Spalte 2 die Kenngrößen für y angezeigt.
5) Berechnen Sie

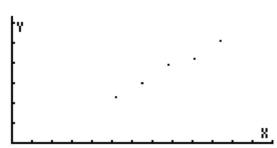
$$S_{xx} = (n-1) \cdot s_x^2 , \quad s_e^2 = \frac{n-1}{n-2} \cdot s_y^2 \cdot (1 - r_{xy}^2)$$

6) Ermitteln Sie für Vertrauensbereiche oder zweiseitige Tests $t_{\alpha/2}$ mit dem Vertrauensbereich (1- α)100 % anhand einer t-Verteilung mit ν = n -2.
7) Ermitteln Sie für ein- oder zweiseitige Tests den Wert von t unter Verwendung der entsprechenden Gleichung für A oder B. Weisen Sie die Nullhypothese zurück, wenn `P-value` < α.
8) Verwenden Sie für Vertrauensbereiche die entsprechenden, oben dargestellten Formeln.

Beispiel 1 – Bestimmen Sie für die folgenden Daten (x,y) den Vertrauensbereich von 95 % für die Steigung B und den Abschnitt A.

x	2,0	2,5	3,0	3,5	4,0
y	5,5	7,2	9,4	10,0	12,2

Geben Sie die Daten (x,y) in die Spalten 1 bzw. 2 von ΣDAT ein. Eine Punktwolke der Daten veranschaulicht einen annähernd linearen Verlauf:



Verwenden Sie die Option `Fit Data` des Menüs ↱ STAT , um folgende Werte zu erhalten:

```
3: '-0,86 + 3,24*X'
2: Correlation: 0,989720229749
1: Covariance: 2,025
```

Diese Ergebnisse bedeuten, dass a = -0,86, b = 3,24, r_{xy} = 0,989720229749 und s_{xy} = 2,025. Der Korrelationskoeffizient ist nahe genug an 1,0, um den linearen Verlauf des Diagramms zu bestätigen.

Über die Option `Single-var…` des Menüs ↱ STAT erhalten wir $\overline{x}$ = 3, s_x = 0,790569415042, $\overline{y}$ = 8,86, s_y = 2,58804945857.

Anschließend berechnen wir für n = 5

$$S_{xx} = (n-1)\cdot s_x^2 = (5-1)\cdot 0.790569415042^2 = 2.5$$

$$s_e^2 = \frac{n-1}{n-2}\cdot s_y^2 \cdot (1-r_{xy}^2) =$$

$$\frac{5-1}{5-2}\cdot 2.5880...^2 \cdot (1-0.9897...^2) = 0.1826...$$

Vertrauensbereiche für die Steigung (B) und den Abschnitt (A):

- Zunächst erhalten wir $t_{n-2,\alpha/2} = t_{3,0.025}$ = 3,18244630528 (Informationen über ein Programm zur Lösung der Gleichung für $t_{\nu,a}$ erhalten Sie in Kapitel 17):
- Anschließend berechnen wir die Größen

  $(t_{n-2,\alpha/2})\cdot s_e/\sqrt{S_{xx}}$ = 3,182…·(0,1826…/2,5)$^{1/2}$ = 0,8602…

  $(t_{n-2,\alpha/2})\cdot s_e\cdot[(1/n)+ \overline{x}^2/S_{xx}]^{1/2}$ =
  3,1824…·√0,1826…·[(1/5)+3²/2,5]$^{1/2}$ = 2,65

- Schließlich lautet der Vertrauensbereich von 95 % für die Steigung B:

  (-0,86-0,860242; -0,86+0,860242) = (-1,72; -0,00024217)

Der Vertrauensbereich von 95 % lautet für den Abschnitt A: (3,24-2,6514; 3,24+2,6514) = (0,58855;5,8914).

Beispiel 2 – Angenommen die in Beispiel 1 verwendeten Daten für y stellen die Dehnung (in hundertstel Zentimeter) eines einer Kraft x (in zehntel Kilogramm) ausgesetzten Metallseiles dar. Wir nehmen für dieses physikalische Phänomen an, dass der Abschnitt A den Wert Null aufweist. Zur Überprüfung dieser Annahme testen wir die Nullhypothese H_0: $A = 0$ gegen die Alternativhypothese H_1: $A \neq 0$ bei dem Signifikanzniveau $\alpha = 0{,}05$.

Die Testkenngröße lautet $t_0 = (a\text{-}0)/[(1/n)+ \bar{x}^2/S_{xx}]^{1/2} = (\text{-}0{,}86)/ [(1/5)+3^2/2{,}5]^{½} = \text{-}0{,}44117$. Der kritische Wert von t für $\nu = n - 2 = 3$ und $\alpha/2 = 0{,}025$ kann mit der in Kapitel 17 entwickelten numerischen Lösung für die Gleichung α = UTPT(γ,t) berechnet werden. In diesem Programm stellt γ den Freiheitsgrad (n-2) und α die Wahrscheinlichkeit für das Überschreiten eines bestimmten Wertes von t dar, d. h. Pr[$t>t_\alpha$] = $1 - \alpha$. Im vorliegenden Beispiel lautet der Wert des Signifikanzniveaus $\alpha = 0{,}05$, g = 3, und $t_{n\text{-}2,\alpha/2} = t_{3,0.025}$. Zudem gilt für $\gamma = 3$ und $\alpha = 0{,}025$, dass $t_{n\text{-}2,\alpha/2} = t_{3,0.025} = 3{,}18244630528$. Da $t_0 > - t_{n\text{-}2,\alpha/2}$, können wir die Nullhypothese H_0: $A = 0$ gegen die Alternativhypothese H_1: $A \neq 0$ bei dem Signifikanzniveau $\alpha = 0{,}05$. nicht zurückweisen.
Dieses Ergebnis bedeutet, dass A = 0 für die lineare Regression geeignet ist. Schließlich wurde für a der Wert –0,86 ermittelt. Dieser Wert ist relativ nahe bei Null.

Beispiel 3 – Testen Sie die Signifikanz für die lineare Regression. Testen Sie die Nullhypothese für die Steigung H_0: $B = 0$ gegen die Alternativhypothese H_1: $B \neq 0$ beim Signifikanzniveau $\alpha = 0{,}05$ für die lineare Anpassung von Beispiel 1.

Die Testkenngröße lautet $t_0 = (b \text{-}B_0)/(s_e/\sqrt{S_{xx}}) = (3{,}24\text{-}0)/(\sqrt{0{,}18266666667/2.5}) = 18{,}95$. Der kritische Wert von t für $\nu = n - 2 = 3$ und $\alpha/2 = 0{,}025$ wurde in Beispiel 2 als $t_{n\text{-}2,\alpha/2} = t_{3,0.025} = 3{,}18244630528$ ermittelt. Da $t_0 > t_{\alpha/2}$, müssen wir die Nullhypothese H_1: $B \neq 0$ beim Signifikanzniveau $\alpha = 0{,}05$ für die lineare Anpassung von Beispiel 1 zurückweisen.

Mehrfache lineare Anpassung

Gegeben sei ein Datensatz der Form

x_1	x_2	x_3	…	x_n	y
x_{11}	x_{21}	x_{31}	…	x_{n1}	y_1
x_{12}	x_{22}	x_{32}	…	x_{n2}	y_2
x_{13}	x_{32}	x_{33}	…	x_{n3}	y_3
.	.	.		.	.
.	.	.	.	.	.
$x_{1,m-1}$	$x_{2,m-1}$	$x_{3,m-1}$	…	$x_{n,m-1}$	y_{m-1}
$x_{1,m}$	$x_{2,m}$	$x_{3,m}$	…	$x_{n,m}$	y_m

Angenommen wir möchten eine Datenanpassung der Form $y = b_0 + b_1 \cdot x_1 + b_2 \cdot x_2 + b_3 \cdot x_3 + \ldots + b_n \cdot x_n$ ermitteln. Sie können die Näherung der kleinsten Quadrate an die Werte der Koeffizienten $\mathbf{b} = [b_0 \quad b_1 \quad b_2 \quad b_3 \ldots b_n]$ durch Erzeugen der Matrix $\mathbf{X}$ erhalten:

$$\begin{bmatrix} 1 & x_{11} & x_{21} & x_{31} & \cdots & x_{n1} \\ 1 & x_{12} & x_{22} & x_{32} & \cdots & x_{n2} \\ 1 & x_{13} & x_{32} & x_{33} & \cdots & x_{n3} \\ . & . & . & . & & . \\ . & . & . & . & . & . \\ 1 & x_{1,m} & x_{2,m} & x_{3,m} & \cdots & x_{n,m} \end{bmatrix}$$

Der Vektor der Koeffizienten wird dann mit $\mathbf{b} = (\mathbf{X}^T \cdot \mathbf{X})^{-1} \cdot \mathbf{X}^T \cdot \mathbf{y}$ ermittelt, wobei $\mathbf{y}$ den Vektor $\mathbf{y} = [y_1\, y_2 \ldots y_m]^T$ darstellt.

Verwenden Sie als *Beispiel* die folgenden Daten, um die mehrfache lineare Anpassung zu ermitteln.

$$y = b_0 + b_1 \cdot x_1 + b_2 \cdot x_2 + b_3 \cdot x_3,$$

x_1	x_2	x_3	y

1,20	3,10	2,00	5,70
2,50	3,10	2,50	8,20
3,50	4,50	2,50	5,00
4,00	4,50	3,00	8,20
6,00	5,00	3,50	9,50

Sie können mit dem Taschenrechner im RPN-Modus wie folgt vorgehen:

Erstellen Sie zunächst im Verzeichnis HOME ein Unterverzeichnis MPFIT (Multiple linear and Polynomial data FITting, Mehrfache lineare Anpassung und Polynomanpassung) und geben Sie das Unterverzeichnis MPFIT ein. Geben Sie im Unterverzeichnis folgendes Programm ein:

```
« → X y « X TRAN X * INV X TRAN * y * » »
```

und speichern Sie es in einer Variablen MTREG (MulTiple REGression).

Geben Sie anschließend die Matrizen **X** und **b** in den Stack ein:

[[1;1,2;3,1;2][1;2,5;3,1;2,5][1;3,5;4,5;2,5][1;4,4;5,3][1;6,5;3,5]]

ENTER ENTER (speichern Sie eine zusätzliche Kopie)

[5,7;8,2;5,0;8,2;9,5] ENTER

Drücken Sie VAR MTREG. Das Ergebnis lautet: [-2,1649…,–0,7144…,-1,7850…,7,0941…], d. h.

$$y = -2,1649–0,7144\cdot x_1 -1,7850\times 10^{-2}\cdot x_2 + 7,0941\cdot x_3.$$

Der Stack des Taschenrechners sollte den Wert von Matrix X und Vektor b enthalten. Die angepassten Werte von y werden durch **y** = **X**·**b** ermittelt. Drücken Sie daher einfach ×, um folgendes Ergebnis zu erhalten: [5,63... 8,25... 5,03... 8,23... 9,45...],

Vergleichen Sie diese angepassten Werte wie in der folgenden Tabelle mit den ursprünglichen Werten:

x_1	x_2	x_3	y	an y angepasst
1,20	3,10	2,00	5,70	5,63
2,50	3,10	2,50	8,20	8,25
3,50	4,50	2,50	5,00	5,03
4,00	4,50	3,00	8,20	8,23
6,00	5,00	3,50	9,50	9,45

Polynomanpassung

Gegeben sei der x-y-Datensatz $\{(x_1,y_1), (x_2,y_2), \ldots, (x_n,y_n)\}$. Angenommen wir möchten ein Polynom der Ordnung p an diesen Datensatz anpassen. Mit anderen Worten, wir möchten eine Datenanpassung der Form $y = b_0 + b_1 \cdot x + b_2 \cdot x^2 + b_3 \cdot x^3 + \ldots + b_p \cdot x^p$ durchführen. Sie können die Näherung der kleinsten Quadrate an die Werte der Koeffizienten $\mathbf{b} = [b_0 \;\; b_1 \;\; b_2 \;\; b_3 \ldots b_p]$ durch Erzeugen der Matrix **X** erhalten.

$$\begin{bmatrix} 1 & x_1 & x_1^2 & x_1^3 & \ldots & x_1^{p-1} & y_1^{\,p} \\ 1 & x_2 & x_2^2 & x_2^3 & \ldots & x_2^{\,p-1} & y_2^{\,p} \\ 1 & x_3 & x_3^2 & x_3^3 & \ldots & x_3^{\,p-1} & y_3^{\,p} \\ . & . & . & . & & . & . \\ . & . & . & . & . & . & . \\ 1 & x_n & x_n^2 & x_n^3 & \ldots & x_n^{\,p-1} & y_n^{\,p} \end{bmatrix}$$

Der Vektor der Koeffizienten wird dann mit $\mathbf{b} = (\mathbf{X}^T \cdot \mathbf{X})^{-1} \cdot \mathbf{X}^T \cdot \mathbf{y}$ ermittelt, wobei $\mathbf{y}$ den Vektor $\mathbf{y} = [y_1 \; y_2 \ldots y_n]^T$ darstellt.
In Kapitel 10 wurde die einem Vektor $\mathbf{x} = [x_1 \; x_2 \ldots x_m]$ entsprechende Vandermonde-Matrix definiert. Die Vandermonde-Matrix ist mit der Matrix **X** für die Polynomanpassung vergleichbar, enthält jedoch lediglich *n* und nicht (*p+1*) Spalten.
Wir können die Funktion VANDERMONDE zum Erstellen der Matrix **X** verwenden, wenn wir die folgenden Regeln beachten:

Wenn $p = n\text{-}1$, ist $\mathbf{X} = \mathbf{V}_n$.
Wenn $p < n\text{-}1$, entfernen Sie die Spalten $p+2, \ldots, n\text{-}1, n$ aus $\mathbf{V}_n$, um $\mathbf{X}$ zu erzeugen.
Wenn $p > n\text{-}1$, fügen Sie die Spalten $n+1, \ldots, p\text{-}1, p+1$ zu $\mathbf{V}_n$ hinzu, um die Matrix $\mathbf{X}$ zu erzeugen.

In Schritt 3 dieser Liste müssen wir berücksichtigen, dass die Spalte i ($i = n+1, n+2, \ldots, p+1$) den Vektor $[x_1^i\ x_2^i\ \ldots\ x_n^i]$ darstellt. Wenn wir für x eine Liste von Datenwerten und keinen Vektor verwenden, d. h. $\mathbf{x} = \{ x_1\ x_2\ \ldots\ x_n \}$, können wir die Folge $\{ x_1^i\ x_2^i\ \ldots\ x_n^i \}$ einfach berechnen. Wir wandeln anschließend diese Liste in einen Vektor um und verwenden das Menü COL, um diese Spalten der Matrix $\mathbf{V}_n$ hinzuzufügen, bis $\mathbf{X}$ fertig gestellt ist.

Nachdem $\mathbf{X}$ erstellt und der Vektor $\mathbf{y}$ verfügbar ist, entspricht die Berechnung des Koeffizientenvektors $\mathbf{b}$ der mehrfachen linearen Anpassung (der vorherigen Matrixanwendung). Somit können wir ein Programm zum Berechnen der Polynomanpassung schreiben, das auf dem bereits für die mehrfache lineare Anpassung verwendeten Programm aufbaut. Wir müssen diesem Programm die oben aufgeführten Schritte 1 bis 3 hinzufügen.

Der *Algorithmus* für dieses Programm kann daher wie folgt geschrieben werden:

Vektoren **x** und **y** derselben Dimension als Listen eingeben. (Hinweis: Da für die Funktion VANDERMONDE eine Liste als Eingabe verwendet wird, empfiehlt es sich, die (x,y)-Daten als Liste einzugeben.) Außerdem den Wert von p eingeben.

- n = Größe von Vektor **x** bestimmen
- Mit der Funktion VANDERMONDE die Vandermonde-Matrix $\mathbf{V}_n$ für die eingegebene Liste **x** generieren
- If $p = n\text{-}1$, then
  $\mathbf{X} = \mathbf{V}_n$,
  Else If $p < n\text{-}1$
  die Spalten $p+2, \ldots, n$ aus $\mathbf{V}_n$ entfernen, um $\mathbf{X}$ zu erstellen
  (FOR-Schleife und COL- verwenden)
  Else

die Spalten *n+1, …, p+1* zu $\mathbf{V}_n$ hinzufügen, um **X** zu erstellen (FOR-Schleife, x^i berechnen, in Vektor konvertieren, COL+ verwenden)

- **y** in Vektor konvertieren
- **b** mit dem Programm MTREG berechnen (siehe obiges Beispiel für mehrfache lineare Anpassung)

Es folgt die *Übertragung des Algorithmus* in ein Programm in der Sprache USER RPL. (Weitere Informationen über das Programmieren erhalten Sie in Kapitel 21.)

```
«                          Programm starten
 → x y p                   Listen x und y sowie p (Ebenen 3,2,1)
                           eingeben
 «                         Unterprogramm 1 starten
 x SIZE → n                Größe der Liste x bestimmen
 «                         Unterprogramm 2 starten
   x VANDERMONDE           x in Stack ablegen, Vn ermitteln
   IF 'p<n-1' THEN         Durch diese IF-Klausel wird Schritt 3 des
                           Algorithmus implementiert
     n                     n in Stack ablegen
     p 2 +                 p+1 berechnen
    FOR j                  Schleife j = n-1, n-2, …, p+1, step = -1
                           starten
        j  COL– DROP       Spalte entfernen und aus Stack löschen
   -1 STEP                 FOR-STEP-Schleife beenden
  ELSE
    IF 'p>n-1' THEN
      n 1 +                n+1 berechnen
      p 1 +                p+1 berechnen
     FOR j                 Schleife mit j = n, n+1, …, p+1 starten
        x  j  ^            xj als Liste berechnen
        OBJ→ →ARRY         Liste in Feld konvertieren
        j COL+             Spalte zu Matrix hinzufügen
     NEXT                  FOR-NEXT-Schleife beenden
   END                     Zweite IF-Klausel beenden
 END                       Erste IF-Klausel beenden. Das Ergebnis ist X
```

y OBJ→ →ARRY	Liste **y** in ein Feld konvertieren
MTREG	Von Programm MTREG verwendetes **X** und **y**
→NUM	In Dezimalformat konvertieren
»	Unterprogramm 2 beenden
»	Unterprogramm 1 beenden
»	Hauptprogramm beenden

Speichern Sie das Programm in einer Variablen POLY (POLYnomanpassung).

Verwenden Sie als *Beispiel* die folgenden Daten, um eine Polynomanpassung mit p = 2, 3, 4, 5, 6 zu erhalten.

x	y
2,30	179,72
3,20	562,30
4,50	1969,11
1,65	65,87
9,32	31220,89
1,18	32,81
6,24	6731,48
3,45	737,41
9,89	39248,46
1,22	33,45

Da wir für die Anpassung von Polynomen unterschiedlicher Ordnung dieselben x-y-Daten verwenden, sollten Sie die Listen der Datenwerte x und y in den Variablen xx bzw. yy speichern. Auf diese Weise müssen wir die Daten nicht bei jeder Anwendung des Programms POLY erneut eingeben. Gehen Sie daher wie folgt vor:

{ 2,3 3,2 4,5 1,65 9,32 1,18 6,24 3,45 9,89 1,22 } ENTER 'xx' STO▸
{179,72 562,30 1969,11 65,87 31220,89 32,81 6731,48 737,41 39248,46 33,45} ENTER 'yy' STO▸

Verwenden Sie zum Anpassen der Daten an die Polynome folgende Eingabe:

XX YY 2 POLY. Ergebnis: [4527,73 -3958,52 742,23]
d. h. $y = 4527{,}73\text{-}39{,}58x\text{+}742{,}23x^2$

XX YY 3 POLY. Ergebnis: [–998,05 1303,21 -505,27 79,23]
d. h. $y = \text{-}998{,}05\text{+}1303{,}21x\text{-}505{,}27x^2\text{+}79{,}23x^3$

XX YY 4 POLY. Ergebnis: [20,92 –2,61 –1,52 6,05 3,51]
d. h. $y = 20{,}92\text{-}2{,}61x\text{-}1{,}52x^2\text{+}6{,}05x^3\text{+}3{,}51x^4$.

XX YY 5 POLY. Ergebnis: [19,08 0,18 –2,94 6,36 3,48 0,00]
d. h. $y = 19{,}08\text{+}0{,}18x\text{-}2{,}94x^2\text{+}6{,}36x^3\text{+}3{,}48x^4\text{+}0{,}0011x^5$

XX YY 6 POLY. Ergebnis: [-16,73 67,17 –48,69 21,11 1,07 0,19 0,00]
d. h. $y = \text{-}16{,}73\text{+}67{,}17x\text{-}48{,}69x^2\text{+}21{,}11x^3\text{+}1{,}07x^4\text{+}0{,}19x^5\text{+}0{,}0058x^6$

Auswählen der besten Anpassung

Wie Sie aus den obigen Ergebnissen ersehen, können Sie ein beliebiges Polynom an einen Satz von Daten anpassen. Nun ergibt sich die Frage, welche Anpassung für die Daten am besten geeignet ist. Für die Auswahl der besten Anpassung können mehrere Kriterien verwendet werden:

- Der Korrelationskoeffizient r. Dieser Wert ist auf den Bereich $-1 < r < 1$ eingeschränkt. Je näher *r* an dem Wert +1 oder –1 ist, desto besser ist die Datenanpassung.
- Die Summe der quadratischen Fehler SSE. Hierbei handelt es sich um die Größe, die durch den Ansatz der kleinsten Quadrate minimiert werden soll.
- Ein Diagramm von Residuen. Hierbei handelt es sich um ein Diagramm der Fehler, die den einzelnen ursprünglichen Datenpunkten entsprechen. Wenn diese Fehler vollständig zufällig sind, darf das Residuendiagramm keinen bestimmten Verlauf aufweisen.

Bevor wir diese Kriterien in einem Programm implementieren, stellen wir folgende <u>Definitionen</u> vor:

Für die Vektoren **x** und **y** der an die Polynomgleichung anzupassenden Daten erstellen wir die Matrix **X** und berechnen mit ihr einen Vektor der Polynomkoeffizienten **b**. Wir können mit **y'** = **X**·**b** einen *Vektor angepasster Daten* **y'** berechnen.

Ein *Fehlervektor* wird mit **e** = **y** – **y**' berechnet.

Die *Summe der quadratischen Fehler* ist gleich dem Quadrat des Betrags des Fehlervektors, d. h. SSE = $|\mathbf{e}|^2 = \mathbf{e} \bullet \mathbf{e} = \Sigma\, e_i^2 = \Sigma\, (y_i - y'_i)^2$.

Zum Berechnen des Korrelationskoeffizienten müssen wir zunächst die *Gesamtquadratsumme* SST (Sum of Squared Totals) ermitteln, die als SST = $\Sigma\, (y_i - \bar{y})^2$ definiert ist, wobei y den *Mittelwert* der ursprünglichen Werte von y darstellt, d. h. $\bar{y} = (\Sigma y_i)/n$.

Für SSE und SST ist der Korrelationskoeffizient definiert durch

$$r = [1-(SSE/SST)]^{1/2}.$$

Im Folgenden wird das neue Programm mit der Berechnung von SSE und r dargestellt (wir empfehlen nochmals, auf der letzten Seite dieses Kapitels nachzulesen, wie die Variablen- und Befehlsnamen in dem Programm erstellt werden):

«	Programm starten
→ x y p	Listen x und y sowie Zahl p eingeben.
«	Unterprogramm 1 starten.
x SIZE → n	Größe der Liste x bestimmen
«	Unterprogramm 2 starten
x VANDERMONDE	x in Stack ablegen, $\mathbf{V}_n$ ermitteln
IF 'p<n-1' THEN	Diese IF-Klausel entspricht Schritt 3 des Algorithmus.
n	n in Stack ablegen.
p 2 +	p+1 berechnen.
FOR j	Schleife j = n-1 bis p+1, step = -1 starten
j COL– DROP	Spalte entfernen und aus Stack löschen.
-1 STEP	FOR-STEP-Schleife beenden.

ELSE	
IF 'p>n-1' THEN	
n 1 +	n+1 berechnen
p 1 +	p+1 berechnen
FOR j	Schleife mit j = n, n+1, …, p+1 starten
x j ^	$\mathbf{x}^j$ als Liste berechnen
OBJ→ →ARRY	Liste in Feld konvertieren
j COL+	Spalte zu Matrix hinzufügen
NEXT	FOR-NEXT-Schleife beenden
END	Zweite IF-Klausel beenden
END	Erste IF-Klausel beenden. Erzeugt **X**
y OBJ→ →ARRY	Liste **y** in ein Feld konvertieren
→ X yv	Matrix und Feld als X und y eingeben
«	Unterprogramm 3 starten
X yv MTREG	Von Programm MTREG verwendetes **X** und **y**
→NUM	Bei Bedarf Umwandlung in Fließkommawerte
→ b	Als b übergebener Ergebnisvektor
«	Unterprogramm 4 starten
b yv	**b** und yv in Stack ablegen
X b *	**X**·**b** berechnen
-	**e** = **y** - **X**·**b** berechnen
ABS SQ DUP	SSE berechnen, Kopie erstellen
y ΣLIST n /	$\bar{y}$ berechnen
n 1 →LIST SWAP CON	Vektor mit *n* Werten von $\bar{y}$ erstellen
yv – ABS SQ	SST berechnen
/	SSE/SST berechnen
NEG 1 + √	r = $[1–SSE/SST]^{1/2}$ berechnen
"r" →TAG	Tagergebnis als „r"
SWAP	Ebene 1 und 2 des Stacks vertauschen
"SSE" →TAG	Tagergebnis als SSE
»	Unterprogramm 4 beenden

- » Unterprogramm 3 beenden
- » Unterprogramm 2 beenden
- » Unterprogramm 1 beenden
- » Hauptprogramm beenden

Speichern Sie dieses Programm unter dem Namen POLYR, um auf die Berechnung des Korrelationskoeffizienten r hinzuweisen.

Wenn das Programm POLYR für Werte zwischen 2 und 6 verwendet wird, wird die folgende Tabelle mit Werten des Korrelationskoeffizienten r und der Summe der quadratischen Fehler SSE erzeugt:

p	*r*	*SSE*
2	0,9971908	10731140,01
3	0,9999768	88619,36
4	0,9999999	7,48
5	0,9999999	8,92
6	0,9999998	432,61

Während der Korrelationskoeffizient für alle Werte von p in der Tabelle sehr nahe an 1,0 ist, variieren die Werte von SSE stark. Der kleinste Wert von SSE entspricht p = 4. Sie können somit die bevorzugte Polynomanpassung für die ursprünglichen x-y-Daten wie folgt auswählen:

$$y = 20,92-2,61x-1,52x^2+6,05x^3+3,51x^4.$$

Kapitel 19
Zahlen mit unterschiedlicher Basis

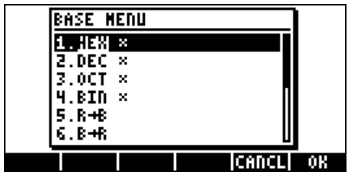
In diesem Kapitel zeigen wir Beispiele für Zahlenberechnungen mit anderer Basis als der Dezimalbasis.

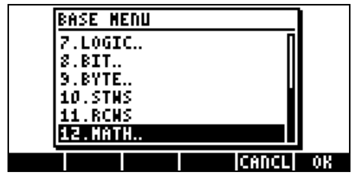
Definitionen

Das Zahlensystem, dass für das tägliche Rechnen verwendet wird, ist als *Dezimal*system bekannt, da es 10 (Latein, deca) Stellen, nämlich 0-9, verwendet, um eine wirkliche Zahl zu schreiben. Anderseits verwenden Computer ein System, das auf zwei möglichen Zuständen basiert oder dem *Binär*system. Diese beiden Zustände werden durch 0 und 1 dargestellt, ON und OFF oder Hochspannung und Niedrigspannung. Computer verwenden auch Zahlensysteme, die auf acht Stellen (0-7), also dem *Oktal*system, und sechzehn Stellen (0-9, A-F) oder *hexadezimal* basieren. Wie im Dezimalsystem bestimmt die relative Position der Stellen den Wert. Im Allgemeinen kann eine Zahl n in Basis b als Zahlenreihe n = $(a_1a_2 \ldots a_n.c_1c_2 \ldots c_m)_b$. geschrieben werden. Der "Punkt" teilt *n* "integere" Stellen von m "dezimalen" Stellen. Der Zahlenwert umgewandelt in unser übliches Dezimalsystem wird wir wie folgt berechnet: $n = a_1 \cdot bn^{-1} + a_2 \cdot b^{n-2} + \ldots + a_n b^0 + c_1 \cdot b^{-1} + c_2 \cdot b^{-2} + \ldots + c_m \cdot b^{-m}$. Zum Beispiel: $(15{,}234)_{10} = 1 \cdot 10^1 + 5 \cdot 10^0 + 2 \cdot 10^{-1} + 3 \cdot 10^{-2} + 4 \cdot 10^{-3}$ und $(101{,}111)_2 = 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 + 1 \cdot 2^{-1} + 1 \cdot 2^{-2} + 1 \cdot 2^{-3}$

Das Menü BASE

Während der Rechner normalerweise mit dem Dezimalsystem bedient wird, können Sie auch Berechnungen mit dem Binär-, Oktal- oder Hexadezimalsystem durchführen. Viele der Funktionen zur Manipulierung der Zahlensysteme außer dem Dezimalsystem sind über das Menü BASE verfügbar und über ↱ BASE (die Taste 3) verfügbar. Wenn das Systemflag 117 auf CHOOSE boxes gesetzt ist, zeigt das Menü BASE die folgenden Einträge:





Ist die Systemmarkierung 117 auf die Menüs SOFT eingestellt, zeigt das Menü BASE das Folgende:

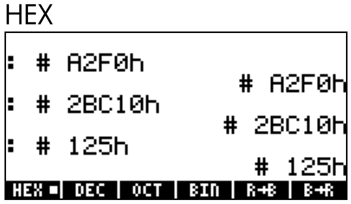
Mit diesem Format wird deutlich, dass die Einträge LOGIC, BIT und BYTE selbst im Menü BASE Untermenüs sind. Diese Menüs werden später in diesem Kapitel besprochen.

Die Funktionen HEX, DEC, OCT und BIN

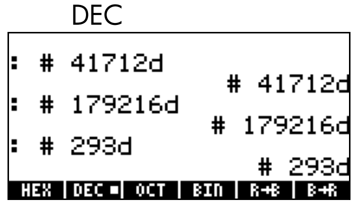
Zahlen in Nicht-Dezimalsystemen wird ein #-Symbol im Rechner vorangestellt. Das Symbol # ist sofort verfügbar als ⬅ # (die Taste 3). Zur Auswahl welches Zahlensystem (derzeitige Basis) für Zahlen verwendet wird, denen ein # vorangestellt wird, wählen Sie eine der folgenden Funktionen im ersten Menü BASE, d.h., HEX(adezimal), DEC(dezimal), OCT(oktal), oder BIN(är). Wenn beispielsweise HEX■ ausgewählt wird, wird jede Zahl, die in den Rechner geschrieben wird und mit einem # beginnt, als Hexadezimalzahl geschrieben. So können Sie Zahlen wie #53, #A5B, etc. in dieses System schreiben. Wenn andere Systeme ausgewählt werden, werden die Zahlen automatisch in die neue aktuelle Basis umgewandelt.

Die folgenden Beispiele zeigen dieselben Zahlen mit vorangestelltem #-Symbol für verschiedene Basen:

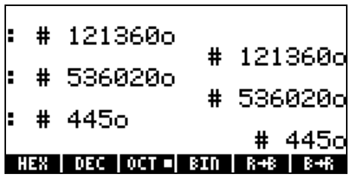
HEX



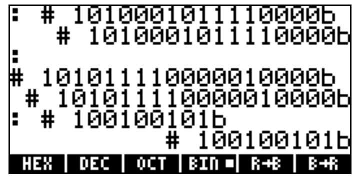
DEC



OCT



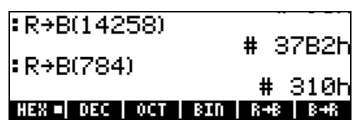
BIN



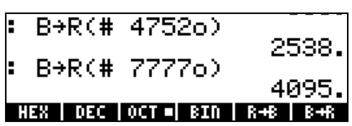
Da das Dezimalsystem (DEC) 10 Stellen besitzt (0,1,2,3,4,5,6,7,8,9), umfasst das Hexadezimalsystem (HEX) 16 Stellen (0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F), das Oktalsystem (OCT) 8 Stellen (0,1,2,3,4,5,6,7) und das Binärsystem (BIN) nur 2 Stellen (0,1).

Umwandlung zwischen Zahlensystemen

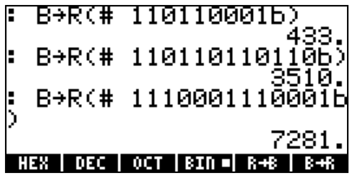
Unabhängig davon welches Zahlensystem gewählt wurde, es wird als Binärsystem bezeichnet, um die Funktionen R→B und B→R verwenden zu können. Wenn beispielsweise HEX■ ausgewählt ist, wandelt die Funktion B→R jede Hexadezimalzahl (mit vorangestelltem #) in eine Dezimalzahl um, während die Funktion R→B in die entgegengesetzte Richtung umwandelt. Probieren Sie die folgenden Beispiele aus, HEX ist die derzeitige Basis:

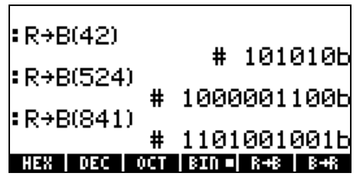


Die folgenden Beispiele zeigen Umwandlungen mit dem Oktalsystem als Basis:



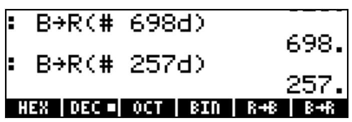
Wir zeigen auch Transformationen mit dem Binärsystem als aktuelle Basis:

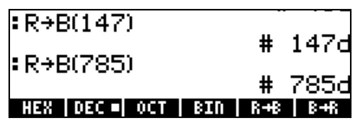




Wir weisen darauf hin, dass Sie immer, wenn Sie eine Nummer mit vorangestelltem # eingeben, als Eintrag die eingegebene Zahl mit vorangestelltem # und nachgestelltem h, o, oder b (hexadezimal, oktal oder binär) erhalten. Der als Suffix verwendete Buchstabe hängt davon ab, welches nichtdezimale Zahlensystem ausgewählt wurde, d.h. HEX, OCT, oder BIN.

Probieren Sie die folgenden Umwandlungen, damit Sie sehen, was passiert, wenn Sie DEC auswählen:





Die einzige Auswirkung, die die Auswahl des DEC (Dezimalsystem) hat, sind die Dezimalzahlen mit vorangestelltem #-Symbol, die mit dem Suffix d versehen werden.

Wortgröße

Die Wortgröße ist die Bitanzahl in einem Binärprojekt. Standardmäßig beträgt die Wortgröße 64 Bits. Die Funktion RCWS (ReCall WordSize) zeigt die aktuelle Wortgröße. Die Funktion STWS (SeT the WordSize) ermöglicht dem Benutzer, die Wortgröße auf eine beliebige Zahl zwischen 0 und 64 einzustellen.

Die Änderung der Wortgröße beeinflusst die Weise, wie integere Rechenoperationen ausgeführt werden. Wenn beispielsweise ein binäres Integer die aktuelle Wortgröße überschreitet, fallen die Bits vorne weg, bevor eine Rechenoperation mit einer solchen Zahl durchgeführt werden kann.

Rechenoperationen mit binären Integerzahlen

Die Rechenoperationen Addition, Subtraktion, Zeichenwechsel, Multiplikation und Division sind für binäre Integerzahlen definiert. Einige Beispiele werden für Addition und Subtraktion bei verschiedenen aktuellen Basen unten gezeigt:

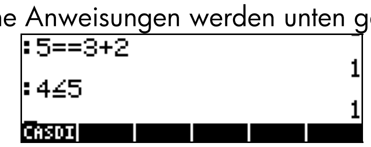
```
#A02h + #12Ah = #B2Ch
#2562d + #298d = #2860d
#5002o + #452o = #5454o
#101000000010b + #100101010b = #101100101100b

#A02h - #12Ah = #8D8h
#2562d - #298d = #2264d
#5002o - #452o = #4330o
#101000000010b - #100101010b = #100011011000b
```

Das Menü LOGIC

Das Menü LOGIC ist über BASE (↱ BASE) erreichbar und bietet die folgenden Funktionen:

Die Funktionen AND, OR, XOR (exklusives OR) und NOT sind logische Funktionen. Bei der Eingabe in diese Funktionen handelt es sich um zwei Werte oder Ausdrücke (einer im Fall von NOT), die als binäre logische Ergebnisse ausgedrückt werden können, d.h. 0 oder 1. Zahlenvergleiche mit den Vergleichsoperatoren =, ≠, >, <, ≤ und ≥, sind logische Anweisungen, die entweder wahr (1) oder falsch (0) sein können. Einige Beispiele für logische Anweisungen werden unten gezeigt:



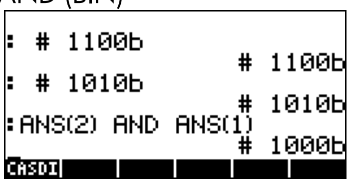
Die Funktionen AND, OR, XOR und NOT können auf die Vergleichs-Anweisungen mit den folgenden Regeln angewendet werden:

1 AND 1 = 1	1 AND 0 = 0	0 AND 1 = 0	0 AND 0 = 0

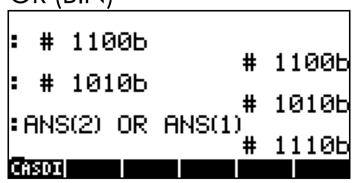
1 OR 1 = 1	1 OR 0 = 1	0 OR 1 = 1	0 OR 0 = 0
1 XOR 1 = 0	1 XOR 0 = 1	0 XOR 1 = 1	0 XOR 0 = 0
NOT(1) = 0	NOT(0) = 1		

Diese Funktionen können zur Bildung logischer Anweisungen zu Programmierzwecken verwendet werden. Im Kontext dieses Kapitels werden sie zur Darstellung der Bit-by-Bit-Ergebnisse entsprechend der oben gezeigten Regeln verwendet. Zum Beispiel:

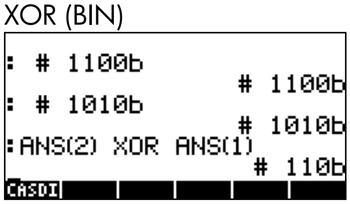
AND (BIN)



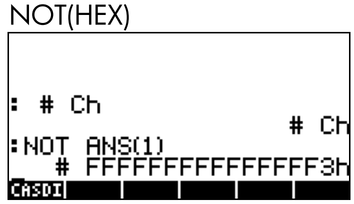
OR (BIN)



XOR (BIN)

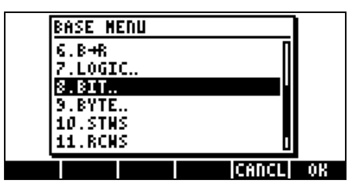


NOT(HEX)



Das Menü BIT

Das Menü BIT ist über BASE (↱ BASE) erreichbar und bietet die folgenden Funktionen:

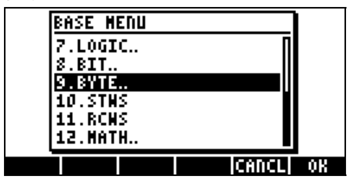


Die Funktionen RL, SL, ASR, SR, RR, sind im Menü BIT enthalten und werden zur Änderung von Bits in einem binären Integer verwendet. Die Definition dieser Funktionen wird unten gezeigt:

RL: Rotate Left one bit (ein Bit links drehen), z.B., #1100b → #1001b
SL: Shift Left one bit (ein Bit nach links schieben), z.B., #1101b → #11010b
ASR: Arithmetic Shift Right one bit (ein Bit arithmetisch rechts schieben), z.B., #1100010b → #110001b
SR: Shift Right one bit (ein Bit nach rechts schieben), z.B., #11011b →#1101b
RR: Rotate Right one bit (ein Bit rechts drehen), z.B., #1101b → #1110b

Das Menü BYTE

Das Menü BIT ist über BASE (BASE) erreichbar und bietet die folgenden Funktionen:



Die Funktionen RLB, SLB, SRB, RRB, sind im Menü BIT, enthalten und werden zur Änderung von Bits in einem binären Integer verwendet. Die Definition dieser Funktionen wird unten gezeigt:

RLB: Rotate Left one byte (ein Byte links drehen), z.B., #1100b → #1001b
SLB: Shift Left one byte (ein Bit nach links schieben), z.B., #1101b → #11010b
SRB: Shift Right one byte (ein Byte nach rechts schieben), z.B., #11011b →#1101b
RRB: Rotate Right one byte (ein Byte rechts drehen), z.B., #1101b → #1110b

Hexadezimalzahlen für Pixelreferenzen

Viele Druckoptionsspezifikationen verwenden Pixelreferenzen als Eingabe, z.B., { #332h #A23h } #Ah 0. 360. ARC, um einen Bogen aus einem Kreis zu zeichnen. Wir verwenden die Funktionen C→PX und PX→C um schnell zwischen Benutzereinheitskoordinaten und Pixelreferenzen umzurechnen. Diese Funktionen finden Sie in der Befehlsreferenz (CAT).

Unten werden einige Beispiele gezeigt:

Kapitel 20
Benutzerdefinierte Menüs und Tastatur

Durch die Anwendung der verschiedenen Taschenrechner-Menüs kennen Sie sich nun mit der Menü-Funktion für verschiedene Anwendungen aus. Sie kennen sich auch mit vielen der Funktionen aus, die mit Hilfe der Tastatur, entweder aufgrund ihrer Hauptfunktion oder durch die Kombination mit der linken Umschalttaste (←), rechten Umschalttaste (→) oder ALPHA-Taste (ALPHA) aufgerufen werden können. In diesem Kapitel geben wir Beispiele aufgabenspezifischer Menüs und der Tasten auf der Tastatur, die für die gewünschte Anwendung hilfreich sein können.

Benutzerdefinierte Menüs

Ein benutzerdefiniertes Menü, ist ein Menü, das vom Benutzer eingerichtet wurde. Die Menüeinstellungen sind in der reservierten Variable CST gespeichert. Somit müssen Sie diese Variable mit den Funktionen zusammenführen, die Sie im Menü anzeigen möchten und mit den für die Softmenü-Tasten gewünschten Befehlen. Um Beispiele eines benutzerdefinierten Menüs geben zu können, müssen wir Flag 117 auf den Menüpunkt SOFT setzen. Setzen Sie das Flag auf SOFT, bevor Sie fortfahren (siehe Kapitel 2 für Anweisungen zur Setzung von Flags).

Menü PRG/MODES/MENU

Befehle, die bei der Anlegung von benutzerdefinierten Menüs hilfreich sind, sind im Menü MENU enthalten, auf das über das Menü PRG (← PRG) zugegriffen werden kann. Das Systemflag 117 auf das Menü SOFT setzen, die Sequenz ← PRG NXT MODES MENU führt zu folgendem Softmenü MENU:

Die folgenden Funktionen sind verfügbar:
MENU: Aktiviert ein Menü durch Eingabe der Nummer
CST: Verweis auf die CST-Variable, z. B. → CST zeigt CST-Inhalte

TMENU: Wird anstatt MENU verwendet, um ein temporäres Menü zu erstellen, ohne die Inhalte von CST zu überschreiben.

RCLMENU: Zeigt die Menü-Nummer des aktuellen Menüs an.

Menü-Nummern (RCLMENU und MENU-Funktionen)

Jedes voreingestellte Menü hat eine zugewiesene Nummer. Angenommen Sie aktivieren das MTH-Menü (MTH).Dann gehen Sie über den Funktionskatalog (CAT) zur Funktion RCLMENU und aktivieren Sie diese. Im ALG-modus ` drücken, nachdem RCLMENU() im Bildschirm angezeigt wird. Das Ergebnis ist die Zahl 3.01. Somit können Sie das Menü MTH aktivieren, indem Sie im ALG-modus MENU(3.01) verwenden, oder 3.01 MENU, im RPN-modus.

Die meisten Menüs können aktiviert werden, ohne die Nummer zu kennen, indem man die Tastatur verwendet. Einige Menüs sind jedoch nicht über die Tastatur abrufbar. Das Softmenü STATS ist z. B. nur über die Funktion MENU zugänglich. Dessen Nummer ist 96.01. Verwenden Sie MENU(96.01) im ALG-modus, oder 96.01 MENU im RPN-modus, um in das Softmenü STAT zu gelangen.

Hinweis: Die Nummer 96.01 in diesem Beispiel stellt das erste (01) Untermenü von Menü 96 dar.

Benutzerdefinierte Menüs (MENU und TMENU-Funktionen)

Angenommen Sie müssen vier Funktionen für eine bestimmte Anwendung aktivieren. Angenommen Sie müssen schnell auf die Funktionen EXP, LN, GAMMA und (ALPHA 2) zugreifen und Sie möchten diese in ein Softmenü verschieben, das für eine Weile aktiv sein wird. Dies könnten Sie durch Einrichtung eines temporären Menüs mit der Funktion TMENU erreichen, oder ein dauerhafteres Menü mit der Funktion MENU. Der Hauptunterschied ist, dass die Funktion MENU die Variable CST erstellt und TMENU nicht. Ist die Variable CST permanent in Ihrem Unterverzeichnis erstellt, dann können Sie das Menü jederzeit aktivieren, indem Sie die Spezifikationen in CST verwenden durch Drücken von CUSTOM . In TMENU gehen die

Spezifikationen verloren, nachdem das temporäre Menü mit einem anderen ersetzt wird.

Z. B. wird ein Menü im RPN-Modus wie folgt eingerichtet:

{EXP LN GAMMA !} `ENTER` TMENU `ENTER`

oder

{EXP LN GAMMA !} `ENTER` MENU `ENTER`

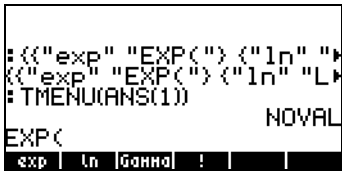
um das folgende Menü zu erstellen:

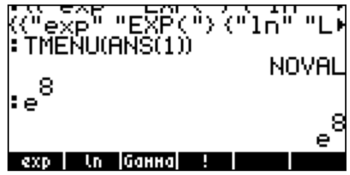
Um eine dieser Funktionen zu aktivieren, brauchen Sie nur ein Funktionsargument (eine Nummer) einzugeben und dann die entsprechende Softmenü-Taste zu drücken.

Im ALG-Modus ist die Liste, die als Argument für eine Funktion TMENU oder MENU eingegeben werden muss, komplizierter:

{{"exp","EXP("),{"ln","LN("),{"Gamma","GAMMA("),{"!","!(")}

Der Grund dafür ist, dass im RPN-modus die Befehlsnamen sowohl Softmenü-Namen als auch Befehle sind. Im ALG-modus haben die Befehlsnamen keine Auswirkung, weil die ALG-funktionen von Klammern und Argumenten gefolgt sein müssen. In der oben angeführten Liste (für den ALG-modus) ist in jedem Untermenü ein Name für die Taste, z. B. "exp" angeführt, gefolgt von der Art, in der die Funktion in den Speicher eingetragen wird, sodass dass Argument der Funktion in die Aufforderung eingegeben werden kann, z. B. "EXP". Wir brauchen uns nicht darum zu kümmern, die Klammer zu schließen, da der Taschenrechner die Klammer beenden wird, bevor er die Funktion ausführt. Die Ausführung der Funktion TMENU im ALG-modus mit der oben angeführten Argumentenliste wird unten beschrieben. Zuerst müssen wir die Liste eingeben, dann errichten wir ein temporäres Menü (siehe Menü Tastennamen) mit Hilfe der Funktion TMENU(ANS(1)). Auf der linken Seite wird auch angezeigt, was passiert, wenn Sie die Softmenü-Taste drücken, z. B. die Aufforderung EXP(. Das Ergebnis nach Eingabe von `8` `ENTER` wird auf der rechten Seite angezeigt.





Eine einfachere Version des Menüs kann wie folgt definiert werden:

MENU({{”EXP(“,“LN(“,“GAMMA(“,”!(“}).

Erweitertes RPN-Menü

Die oben angeführte Liste für den ALG-Modus kann leicht verändert werden, um sie im RPN-Modus anzuwenden. Die veränderte Liste sieht dann wie folgt aus:

{{“exp”,EXP},{“ln”,LN},{“Gamma”,GAMMA},{“!”,!}}

Sie können versuchen, diese Liste mit TMENU oder MENU im RPN-Modus zu verwenden, um sicherzustellen, dass Sie das gleiche Menü erhalten, wie vorher im ALG-Modus.

Menü-Spezifikationen und CST-Variable

Aus den beiden oben angeführten Übungen erkennen wir, dass die allgemeinste Liste für Menüspezifikationen einige Unterlisten enthält, die der Anzahl der in Ihrem benutzerdefinierten Menü anzuzeigenden Elemente entspricht. Jede Unterliste enthält einen Namen für die Menütaste gefolgt von einer Funktion, einem Ausdruck, Namen oder anderem Objekt, das zur Durchführung der gedrückten Menütaste führt. Bei der Spezifizierung der Menüliste im ALG-Modus zu RPN-Modus ist besondere Vorsicht geboten. Im RPN-Modus kann die Auswirkung der Menütaste ein einfacher Taschenrechner-Befehl sein (z. B. EXP, LN, etc., wie oben angeführt), während es im ALG-Modus ein String mit der Befehlaufforderung sein muss, dessen Argument vom Benutzer eingegeben werden muss, bevor ENTER gedrückt und der Befehl ausgeführt wird. Die oben angeführten Beispiele zeigen den Unterschied auf.

Die allgemeine Form der Argumentenliste für Befehle TMENU oder MENU im ALG-Modus lautet:

{“label1”,”function1(“,”ls1(“,”rs1(“}, {“label2”, “function2(“,”ls2(“,”rs2(“},…}

Im RPN-Modus hat die Argumentenliste hingegen folgendes Format:

{“label1”, function1, ls1, rs1}, {“label2”, function2, ls2, rs2},…}

Hierbei stellen Spezifikationen, Funktion1, Funktion 2, usw. die Hauptfunktion der Taste dar, während ls1, ls2, usw. die Tastenfunktion Verschiebung nach links darstellen. Ähnlich dazu stellen rs1, rs2, usw. die Tastenfunktion Verschiebung nach rechts dar. Die Liste wird in Variable CST gespeichert, wenn der Befehl MENU verwendet wird Sie können in jedem Unterverzeichnis eine andere CST-Variable haben und Sie können die aktuellen Inhalte von CST ständig mit denen einer anderen Variable ersetzen, wenn Sie die richtig formatierte Liste für die Erstellung eines anderen benutzerdefinierten Menüs speichern.

Hinweis: Sie können ein 21x8 GROB (siehe Kapitel 22) benutzen, um ein Symbol im Funktionstastenmenü zu erstellen. Als Übung versuchen Sie im RPN-Modus:

{{GROB 21 8 00000EF908FFF900FFF9B3FFF9A2FFF9A3FFF9A0FFF388FF “hp” }}
ENTER MENU

Damit wird das hp Logo auf der Taste F1 dargestellt. Drücken Sie F1 wird der Text 'hp' in der Befehlszeile eingegeben.

Die Tastatur benutzerdefiniert gestalten

Jede Taste auf der Tastatur kann durch zwei Zahlen identifiziert werden, die ihre Reihe und Spalte darstellen. Die Taste VAR (VAR) liegt z. B. in Reihe 3 von Spalte 1 und wird deshalb als Taste 31 bezeichnet. Da jedoch jeder Taste bis zu zehn Funktionen zugewiesen sind, wird jede Funktion durch Dezimalstellen zwischen 0 und 1, gekennzeichnet und zwar laut den folgenden Spezifikationen:

,0 oder 1, Taste allein 0,01 oder 0,11, nicht zutreffend

,2, Taste in Kombination mit [←] ,21, Taste gleichzeitig mit [←]
,3, Taste in Kombination mit [→] ,31, Taste gleichzeitig mit [→]
,4, Taste in Kombination mit [ALPHA] ,41, Taste gleichzeitig mit [ALPHA]
,5, Taste in Kombination mit [ALPHA][←] ,51, [ALPHA]Taste gleichzeitig mit [←]
,6, Taste in Kombination mit [ALPHA][→] ,61, [ALPHA]Taste gleichzeitig mit [→]

Somit wird die VAR-Funktion als Taste 31,0 oder 31,1, bezeichnet und die UPDIR-Funktion als Taste 31,2. Die COPY-Funktion wird Taste 31,3 sein, das große J ist Taste 31.4, und das kleine j ist Taste 31,5. (Taste 31.6 ist nicht definiert). Im Allgemeinen wird eine Taste durch die Reihenfolge XY.Z beschrieben, wobei X = reihennummer, Y = spaltennummer, Z = umschaltung.

Wir können eine bestimmte Taste mit der Taste USER verbinden (linke Umschalttaste mit der Taste [ALPHA] oder [←] USER um eine benutzerdefinierte Tastenfunktion zu erstellen. Im Prinzip kann die ganze Tastatur neu definiert werden, um eine Reihe benutzerdefinierter Funktionen durchzuführen.

Untermenü PRG/MODES/KEY

Befehle, die bei der Anlegung einer benutzerdefinierten Tastatur hilfreich sind, sind im Menü KEYS enthalten, auf das über das Menü PRG ([←] PRG) zugegriffen werden kann. Das Systemflag 117 auf das Menü SOFT setzen, die Sequenz [←] PRG [NXT] MODES KEYS

führt zu folgendem Softmenü KEYS:

Die folgenden Funktionen sind verfügbar:

ASN: Weist einer Taste ein Objekt zu, das durch XY.Z spezifiziert wurde.
STOKEYS: Speichert die benutzerdefinierte Tastenliste.
RCLKEYS: Zeigt die benutzerdefinierte Tastenliste an.
DELKEYS: Macht die Zuweisung für eine oder mehrere Tasten in der benutzerdefinierten Tastenliste rückgängig. Die Argumente sind entweder 0, zum Rückgängigmachen aller benutzerdefinierten

Tasten, oder XY.Z zum Rückgängigmachen der Zuweisung für Taste XY.Z.

Die aktuelle benutzerdefinierte Tastenliste aufrufen
Verwenden Sie den Befehl RCLKEYS um die aktuelle benutzerdefinierte Tastenliste anzuzeigen. Vor jeder benutzerdefinierten Tastenzuweisung, sollte das Ergebnis eine Liste sein, die den Buchstaben S, z. B. {S} enthält.

Ein Objekt einer benutzerdefinierten Taste zuweisen
Angenommen Sie möchten auf den altmodischen PLOT-Befehl zugreifen, der mit dem Taschenrechner der Serie HP 48G eingeführt wurde, aber gegenwärtig nicht direkt von der Tastatur aus verfügbar ist. Die Menü-Nummer für dieses Menü ist 81.01. Sie können dieses Menü wie folgt aktivieren:

ALG-Modus: `MENU(81.01)`
RPN-Modus: `81.01` ENTER `MENU` ENTER

Wenn Sie dieses Menü auf schnelle Weise von der Tastatur aus aktivieren wollen, könnten Sie dieses Menü der Taste GRAPH (F3) zuweisen, dessen Verweisnummer 13.0 ist, z. B. erste Reihe, dritte Spalte, Hauptfunktion. Um ein Objekt einer Taste zuzuweisen, verwenden Sie die Funktion ASN, wie folgt:

ALG-Modus: `ASN(<<MENU(81.01)>>,13.0)`
RPN-Modus: `<< 18.01 MENU >>` ENTER `13.0` ENTER `ASN`

Ein weiteres nützliches Menü ist das ursprüngliche Menü SOLVE (am Ende von Kapitel 6 dieser Anleitung beschrieben), das durch ↱(halten) 7 aktiviert werden kann.

Benutzerdefinierte Tasten anwenden
Um diese benutzerdefinierte Taste zu verwenden, geben sie ↰ USER ein und drücken Sie dann die Taste F3 Beachten Sie, dass der Bildschirm nach Drücken von ↰ USER die Spezifikation `1USR` in der zweiten Display-Zeile

anzeigt. Durch Drücken von ← USER F3 in diesem Beispiel, sollten Sie das Menü PLOT wie folgt wieder herstellen:

Wenn Sie mehr als eine benutzerdefinierte Taste haben und mehr als eine gleichzeitig anwenden möchten, können Sie die Tastatur im USER-Modus durch Eingabe von ← USER ← sperren, bevor Sie die benutzerdefinierten Tasten drücken. Ist die Tastatur im USER-Modus gesperrt, wird die Spezifikation USR in der zweiten Display-Zeile angezeigt. Um die Tastatur wieder freizugeben, drücken Sie nochmals ← USER .

Die Zuweisung einer benutzerdefinierten Taste rückgängig machen

Um die oben durchgeführte Zuweisung zu löschen, verwenden Sie die Funktion DELKEYS wie folgt:

ALG-Modus: `DELKEYS(13.0)`
RPN-Modus: `13.0` ENTER `DELKEYS` ENTER

Mehrere benutzerdefinierte Tasten zuweisen

Der einfachste Weg, um mehrere benutzerdefinierte Tasten zuzuweisen, ist der, eine Liste von Befehlen und Tastenspezifikationen einzugeben. Angenommen wir weisen die drei trigonometrischen Funktionen (SIN, COS, TAN) zu und die drei hyperbolischen Funktionen (SINH, COSH, TANH) und zwar den Tasten von F1 bis F6 , gleichermaßen als benutzerdefinierte Tasten. Verwenden Sie im RPN-Modus:

```
{SIN,11.0,COS,12.0,TAN,13.0,SINH,14.0,COSH,15.0,T
ANH,16.0} ENTER STOKEYS ENTER
```

Verwenden Sie im ALG-Modus:

```
STOKEYS({"SIN(' ,11.0, "COS(", 12.0, "TAN(", 13.0,
"SINH(", 14.0, "COSH(", 15.0, "TANH(", 16.0}) ENTER
```

Verwenden Sie im RPN-Modus z. B. folgenden Tasten:

5 ← USER F1   4 ← USER F2   6 ← USER F3
2 ← USER F4   1 ← USER F5   2 ← USER F6

Um die Zuweisung für alle benutzerdefinierten Tasten rückgängig zu machen, verwenden Sie:
ALG-Modus: `DELKEYS(0)`   RPN-Modus: `0 DELKEYS`

Überprüfen Sie mit Hilfe der Funktion RCLKEYS, ob die benutzerdefinierten Definitionen gelöscht wurden.

Kapitel 21
Programmieren mit UserRPL

Die allgemein verwendete Programmiersprache zur Programmierung des Taschenrechners ist UserRPL. Programmkomponenten können im Zeileneditor, durch Einfügen dieser zwischen den Programm-Containern, « », zusammengeführt werden. Da die meisten Anwender in der Programmierung im RPN-Modus erfahrener sind, werden die meisten Beispiele in diesem Kapitel im RPN-Modus dargestellt. Um die Eingabe der Befehle zu vereinfachen, empfehlen wir zusätzlich das Systemflag 117 auf SOFT-Menüs einzustellen. Die Programme funktionieren natürlich genauso im ALG-Modus, nachdem diese im RPN-Modus getestet und Fehler beseitigt wurden. Wenn Sie es vorziehen, im ALG-Modus zu arbeiten, lernen Sie einfach das Programmieren in RPN, schalten dann den Taschenrechner in den ALG-Modus, um mit den Programmen zu arbeiten. Auf der letzten Seite in diesem Kapitel finden Sie ein einfaches Beispiel der UserRPL-Programmierung im ALG-Modus.

Programmierbeispiel

In den vorangegangenen Kapiteln dieses Handbuches haben wir Ihnen bereits einige Programme zur Verwendung mit verschiedenen Applikationen vorgestellt (beispielsweise wurden in Kapitel 10 CRMC und CRMT vorgestellt, die zum Erstellen einer Matrix anhand einer Anzahl von Listen dienen). In diesem Absatz zeigen wir Ihnen ein einfaches Programm, um Konzepte zur Programmierung des Taschenrechners darzustellen. Das Programm, das wir nachfolgend erstellen, dient zur Definition der Funktion $f(x) = \sinh(x)/(1+x^2)$ und akzeptiert Listen als Argumente (x kann z.B., wie in Kapitel 8 beschrieben, eine Liste von Zahlen sein). In Kapitel 8 wurde bereits erläutert, dass das Plus-Zeichen als Verkettungsoperator für Listen dient und diese nicht Glied für Glied addiert. Eine Glied für Glied Addition in Listen wird mit dem ADD-Operator durchgeführt. Zum Definieren der oben genannten Funktion, verwenden wir das folgende Programm:

```
«'x' STO x SINH 1 x SQ ADD / 'x' PURGE »
```

Zur Eingabe des Programms gehen Sie wie folgt vor:

Tastenfolge:	Erzeugt:	Interpretiert als:
‹‹ ››	«	Starte ein RPL Programm
['] ALPHA x ► STO►	'x' STO	Speichere Ebene 1 in Variable x
ALPHA x	x	Setze x in Ebene 1
MTH HYP SINH	SINH	Berechne sinh aus Ebene 1
1 SPC ALPHA x x^2	1 x SQ	Gebe 1 ein und berechene x^2
MTH MTH LIST ADD	ADD	Berechne $(1+x^2)$,
÷	/	dividiere dann durch
['] ALPHA x ►	'x'	
PRG MEM DIR PURGE	PURGE	Lösche Variable x
ENTER		Programm in Ebene 1

Verwenden Sie diese Tastenfolge: ['] ALPHA G STO► zum Speichern des Programms.

Zum Wiederherstellen des Variablen-Menüs drücken Sie VAR und berechnen dann g(3,5), indem Sie den Wert des Arguments in Ebene 1 eingeben (3 • 5 ENTER) und anschließend g drücken. Das Ergebnis ist 1,2485…, d.h., g(3,5) = 1,2485. Berechnen Sie auch g({1 2 3}), indem Sie die Liste in Ebene 1 der Anzeige eingeben: {} 1 SPC 2 SPC 3 ENTER und g drücken. Ist das CAS-Modul im Modus `EXACT` , wird das Ergebnis {SINH(1)/2 SINH(2)/5 SINH(3)/10} sein. Befindet sich das CAS-Modul im Modus APPROXIMATE, ist das Ergebnis {0,5876.. 0,7253… 1,0017…}.

Globale und lokale Variablen und Unterprogramme

Das Programm g, wie oben definiert, kann angezeigt werden

```
« 'x' STO x SINH 1 x SQ ADD / 'x' PURGE »
```

Beachten Sie, dass das Programm den Variablen-Namen x zur Speicherung der Werte aus den Programmschritten `'x' STO` aus Stack-Ebene 1 verwendet. Die Variable x wird beim Ausführen des Programms, wie alle Variablen, die

Sie vorher gespeichert haben, in ihrem Variablen-Menü gespeichert. Nach Berechnen der Funktion wird die Variable x vom Programm gelöscht, sodass diese nach Beenden des Programms im Variablen-Menü nicht mehr angezeigt wird. Würde die Variable x nach Beenden des Programms nicht gelöscht werden, stünde sie uns auch weiterhin zur Verfügung. Aus diesem Grunde bezeichnet man die Variable x, wie sie in diesem Programm verwendet wird, als *eine globale Variable*. Ein Vorteil liegt darin, dass eine vorher definierte Variable mit dem Namen x von dieser Variablen mit demselben Namen nicht überschrieben, sondern nach Ausführen des Programms aus dem Variablen-Menü entfernt wird. Aus Sicht der Programmierung ist eine globale Variable eine Variable, auf die der Anwender nach Ausführen des Programms zurückgreifen kann. Innerhalb des Programms kann aber auch eine lokale Variable definiert werden, die nur in diesem Programm gilt und auf die nach Ausführen des Programms nicht mehr zugegriffen werden kann. Das vorhergehende Programm könnte folgendermaßen modifiziert werden:

```
« → x « x SINH 1 x SQ ADD / » »
```

Das Pfeilsymbol (→) erhält man aus der Kombination der rechten Shift-Taste ⓡ mit der Taste 0, d.h., ⓡ→. Beachten Sie auch, dass es ein weiteres Paar Programmsymbole (« ») gibt, die anzeigen, dass innerhalb des Hauptprogramms ein Unterprogramm existiert, und zwar `« x SINH 1 x SQ ADD / »`. Das Hauptprogramm startet mit der Kombination `→ x`. Hierdurch wird der Wert in Ebene 1 des Stacks der lokalen Variablen _x zugeordnet. Danach fährt das Programm innerhalb des Unterprogramms fort, x wird in den Stack geschrieben, danach *SINH(x)* ermittelt, anschließend 1 in den Stack geschrieben,, dann x in den Stack, x quadriert und *1* zu x addiert. Abschließend wird der Stack der Ebene 2 (*SINH(x)*) durch den Stack der Ebene 1 ($1+x^2$) dividiert. Danach wird die Programmsteuerung wieder an das Hauptprogramm übergeben. Da sich aber zwischen dem ersten Satz der Symbole (»), die das Programm abschließen, keine weiteren Befehle mehr befinden, wird das Programm beendet. Der letzte Wert im Stack, hier *SINH(x)/* ($1+x^2$), wird als Ergebnis des Programms ausgegeben.

Die der letzten Programmversion entstammende Variable x belegt nie einen Platz unter den Variablen ihres Variablen-Menüs. Sie wird innerhalb des

Speichers des Taschenrechners verarbeitet, und hat keinen Einfluss auf gleichlautende Variablen Ihres Variablen-Menüs. Aus diesem Grunde wird die Variable x, wie hier lokal innerhalb eines Programms verwendet, als *eine lokale Variable* bezeichnet..

Anmerkung: Wollen Sie das Programm modifizieren, setzen Sie den Programmnamen in den Stack (ENTER) und drücken dann . Verwenden Sie anschließend die Pfeiltasten () zum Navigieren innerhalb des Programms. Verwenden Sie die Zurück-/Löschtaste, , um unerwünschte Zeichen zu löschen. Zum Hinzufügen von Programm-Containern (d.h., « »), drücken Sie « ». Da die Symbole immer paarweise erscheinen, müssen Sie diese am Anfang und am Ende eines Unterprogramms aufrufen und dann jeweils eine Komponente mit der Löschtaste löschen, um das erforderliche Programm zu erzeugen, in unserem Fall:

```
« → x « x SINH 1 x SQ ADD / » ».
```

Drücken Sie nach dem Editieren des Programms ENTER. Das modifizierte Programm wird erneut in die Variable gespeichert.

Geltungsbereich für globale Variablen

Jede Variable, die Sie im HOME-Verzeichnis, in anderen Verzeichnissen oder Unterverzeichnissen definieren, wird aus Sicht des Programms als *globale Variable* betrachtet. Der Geltungsbereich einer solchen Variablen, d.h., *die Position im Verzeichnisbaum, wo auf die Variable zugegriffen werden kann*, hängt von der Position der Variablen in diesem Baum ab (siehe Kapitel 2).

Die Regel zur Feststellung des Geltungsbereiches einer Variablen ist wie folgt: Eine Variable ist zugänglich vom Verzeichnis, in dem sie sich befindet oder von jedem Unterverzeichnis dieses Verzeichnisses, sofern sich in dem entsprechenden Unterverzeichnis nicht eine Variable mit gleichem Namen befindet. Aus dieser Regel ergeben sich folgende Schlussfolgerungen:

- Auf eine globale Variable im HOME-Verzeichnis kann von jedem Verzeichnis innerhalb des HOME-Verzeichnisses zugegriffen werden, sofern diese nicht in einem Verzeichnis oder Unterverzeichnis des HOME-Verzeichnisses neu definiert wurde.

- Sobald Sie eine Variable innerhalb eines Verzeichnisses oder Unterverzeichnisses neu definieren, hat diese Definition Vorrang vor allen anderen Definitionen innerhalb eines übergeordneten Verzeichnisses.
- Wird ein Programm gestartet, das auf eine gegebene globale Variable zugreift, verwendet das Programm den Wert der globalen Variablen aus dem Verzeichnis, aus dem das Programm gestartet wurde. Befindet sich im Startverzeichnis des Programms keine Variable mit entsprechendem Namen, sucht das Programm alle übergeordneten Verzeichnisse, einschließlich des HOME-Verzeichnisses durch und verwendet dann den Wert der entsprechenden Variablen, die sich in einem Verzeichnis befindet, das dem Verzeichnis des Programms am nächsten liegt.

Auf ein in einem gegebenen Verzeichnis definiertes Programm kann von diesem und von jedem Unterverzeichnis dieses Verzeichnisses zugegriffen werden.

Alle diese Regeln mögen einem neuen Anwender zunächst verwirrend erscheinen. Diese können aber wie folgt vereinfacht dargestellt werden: *Um Ihre Daten zu organisieren, erstellen Sie Verzeichnisse und Unterverzeichnisse mit sinnvollen Namen, und stellen Sie sicher, dass sich alle benötigten globalen Variablen auch in dem entsprechenden Unterverzeichnis befinden.*

Geltungsbereich für lokale Variablen

Lokale Variablen sind nur in einem Programm oder einem Unterprogramm aktiv. Daher erstreckt sich deren Geltungsbereich nur auf das Programm oder Unterprogramm, in dem diese definiert sind. Ein Beispiel einer lokalen Variablen ist der Index einer FOR-Schleife (weiter unten in diesem Kapitel beschrieben), beispielsweise `« → n x « 1 n FOR j x NEXT n →LIST » »`

Das Menü PRG

In diesem Kapitel wird das Menü PRG (Programmierung) erläutert, wobei Systemflag 117 auf SOFT Menüs eingestellt ist. Mit dieser Einstellung werden die Untermenüs und Befehle im Menü PRG als Beschriftungen der

Funktionstasten dargestellt. Dies vereinfacht das Eingeben von Programmierbefehlen im Zeileneditor.

Um das Menü PRG aufzurufen, verwenden Sie die Tastenfolge ← PRG . Innerhalb des Menüs PRG finden wir folgende Untermenüs (drücken Sie NXT um zur nächsten Untermenü-Seite zu wechseln):

Nachfolgend finden Sie eine kurze Beschreibung dieser Untermenüs und deren Untermenüs:

STACK: Funktionen zur Manipulation von Elementen im RPN-Stack
MEM: Funktionen zur Speichermanipulation
- DIR: Funktionen zur Manipulation von Verzeichnissen
- ARITH: Funktionen zur Manipulation von Indizes, die in Variablen gespeichert sind

BRCH: Kollektion von Untermenüs für Programmverzweigungen und Programmschleifen
- IF: IF-THEN-ELSE-END Anweisung für Programmverzweigungen
- CASE: CASE-THEN-END Anweisung für Programmverzweigungen
- START: START-NEXT-STEP Anweisung für Programmverzweigungen
- FOR: FOR-NEXT-STEP Anweisung für Schleifen
- DO: DO-UNTIL-END Anweisung für Schleifen
- WHILE: WHILE-REPEAT-END Anweisung für Schleifen

TEST: Vergleichs-Operatoren, logische Operatoren, Funktionen zum Testen von Flags
TYPE: Funktionen zum Konvertieren von Objekttypen, zum Aufspalten von Objekten usw.
LIST: Funktionen zum Manipulieren von Listen
- ELEM: Funktionen zum Manipulieren von Elementen einer Liste
- PROC: Funktionen zum Zuordnen von Listen zu Verfahren

GROB: Funktionen zum Manipulieren grafischer Objekte
PICT: Funktionen zum Zeichnen von Bildern in der Grafikanzeige
CHARS: Funktionen zum Manipulieren von Zeichenketten

MODES: Funktionen zum Ändern der Rechenmodi

FMT: Ändern des Zahlen- und Kommaformats

ANGLE: Ändern des Winkelmaßes und Koordinatensystems

FLAG: Setzen und Löschen von Flags, Prüfen des Flag-Status

KEYS: Definieren und Aktivieren anwenderdefinierter Tasten (Kapitel 20)

MENU: Definieren und Aktivieren anwenderdefinierter Menüs (Kapitel 20)

MISC: Ändern verschiedener anderer Modi (Warnton, Uhr usw.)

IN: Funktionen für Programmeingaben

OUT: Funktionen für Programmausgaben

TIME: Zeitgesteuerte Funktionen

ALRM: Manipulieren des Alarms

ERROR: Funktionen zur Fehlerbehandlung

IFERR: IFERR-THEN-ELSE-END Anweisung für Fehlerbehandlung

RUN: Funktionen zum Starten von Programmen und Fehlersuche in diesen

Navigation durch die RPN-Untermenüs

Starten Sie mit der Tastenkombination ← PRG . Drücken Sie dann die entsprechende Funktionstaste (z.B. MEM). Wollen Sie dann auf ein Untermenü des Menüs zugreifen (z.B. DIR innerhalb des Menüs MEM), drücken Sie die entsprechende Taste. Um in der Hierarchie ein Menü nach oben zu wechseln, drücken Sie NXT bis Sie zu dem übergeordneten Menü (z.B. zu MEM im Untermenü DIR) oder zum Menü PRG (PRG) gelangen.

Funktionen der Untermenüs

Nachfolgend finden Sie eine Auflistung der Untermenüs des Menüs PRG und deren jeweilige Funktionen.

STACK	**MEM/DIR**	**BRCH/IF**	**BRCH/WHILE**	**TYPE**
DUP	PURGE	IF	WHILE	OBJ→
SWAP	RCL	THEN	REPEAT	→ARRY
DROP	STO	ELSE	END	→LIST
OVER	PATH	END		→STR
ROT	CRDIR		**TEST**	→TAG

UNROT	PGDIR	**BRCH/CASE**	==	→UNIT
ROLL	VARS	CASE	≠	C→R
ROLLD	TVARS	THEN	<	R→C
PICK	ORDER	END	>	NUM
UNPICK			≤	CHR
PICK3	**MEM/ARITH**	**BRCH/START**	≥	DTAG
DEPTH	STO+	START	AND	EQ→
DUP2	STO-	NEXT	OR	TYPE
DUPN	STOx	STEP	XOR	VTYPE
DROP2	STO/		NOT	
DROPN	INCR	**BRCH/FOR**	SAME	**LIST**
DUPDU	DECR	FOR	TYPE	OBJ→
NIP	SINV	NEXT	SF	→LIST
NDUPN	SNEG	STEP	CF	SUB
	SCONJ		FS?	REPL
MEM		**BRCH/DO**	FC?	
PURGE	**BRCH**	DO	FS?C	
MEM	IFT	UNTIL	FC?C	
BYTES	IFTE	END	LININ	
NEWOB				
ARCHI				
RESTO				

LIST/ELEM	**GROB**	**CHARS**	**MODES/FLAG**	**MODES/MISC**
GET	→GROB	SUB	SF	BEEP
GETI	BLANK	REPL	CF	CLK
PUT	GOR	POS	FS?	SYM
PUTI	GXOR	SIZE	FC?	STK
SIZE	SUB	NUM	FS?C	ARG
POS	REPL	CHR	FS?C	CMD
HEAD	→LCD	OBJ→	FC?C	INFO
TAIL	LCD→	→STR	STOF	
	SIZE	HEAD	RCLF	**IN**
LIST/PROC	ANIMATE	TAIL	RESET	INFORM

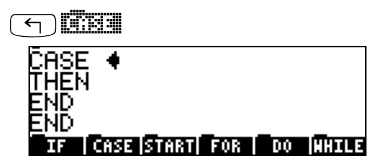
DOLIST		SREPL		NOVAL
DOSUB	**PICT**		**MODES/KEYS**	CHOOSE
NSUB	PICT	**MODES/FMT**	ASN	INPUT
ENDSUB	PDIM	STD	STOKEYS	KEY
STREAM	LINE	FIX	RECLKEYS	WAIT
REVLIST	TLINE	SCI	DELKEYS	PROMPT
SORT	BOX	ENG		
SEQ	ARC	FM,	**MODES/MENU**	**OUT**
	PIXON	ML	MENU	PVIEW
	PIXOF		CST	TEXT
	PIX?	**MODES/ANGLE**	TMENU	CLLCD
	PVIEW	DEG	RCLMENU	DISP
	PX→C	RAD		FREEZE
	C→PX	GRAD		MSGBOX
		RECT		BEEP
		CYLIN		
		SPHERE		

TIME	**ERROR**	**RUN**
DATE	DOERR	DBUG
→DATE	ERRN	SST
TIME	ERRM	SST↓
→TIME	ERR0	NEXT
TICKS	LASTARG	HALT
		KILL
TIME/ALRM	**ERROR/IFERR**	OFF
ACK	IFERR	
ACKALARM	THEN	
STOALARM	ELSE	
RCLALARM	END	
DELALARM		
FINDALARM		

Kürzel innerhalb des Menüs PRG

Viele der oben für das Menü PRG aufgeführten Funktionen stehen auch auf andere Weise zur Verfügung:

- Vergleichsoperatoren (≠, ≤, <, ≥, >) sind über die Tastatur verfügbar.
- Viele Funktionen und Einstellungen im Untermenü MODES können über die Eingabefunktionen der MODE-Taste aktiviert werden.
- Auf Funktionen des Untermenüs TIME kann über die Tastenfolge ↱ TIME zugegriffen werden.
- Die Funktionen STO und RCL (im Untermenü MEM/DIR) sind über die Tasten STO▸ und ↰ RCL verfügbar.
- Die Funktionen RCL und PURGE (im Untermenü MEM/DIR) sind über das Menü TOOL (TOOL) verfügbar.
- Wenn Sie im Untermenü BRCH vor Drücken einer beliebigen Untermenü-Taste die linke (↰) oder die rechte Shift-Taste (↱) drücken, werden Anweisungen zur ausgewählten Untermenü-Taste erstellt. Dies funktioniert nur, wenn der Taschenrechner im RPN-Modus steht. Nachfolgend einige Beispiele:



↱ START

↱ FOR

↰ DO

↰ WHILE

Beachten Sie, dass das Einfügezeichen (◆) nach dem Schlüsselwort jeder Anweisung steht, sodass Sie mit der Eingabe gleich an der richtigen Stelle beginnen können.

Tastenfolge für häufig verwendete Befehle

Im Folgenden finden Sie Tastenfolgen, mit denen Sie häufig vorkommende Befehle zur numerischen Programmierung im Menü PRG aufrufen können. Die Befehle werden vorerst nach Menüs aufgeführt:

STACK

DUP	↰ PRG STACK DUP
SWAP	↰ PRG STACK SWAP
DROP	↰ PRG STACK DROP

MEM DIR

PURGE	↰ PRG MEM DIR PURGE
ORDER	↰ PRG MEM DIR ORDER

BRCH IF

IF	↰ PRG BRCH IF IF
THEN	↰ PRG BRCH IF THEN
ELSE	↰ PRG BRCH IF ELSE
END	↰ PRG BRCH IF END

BRCH CASE

CASE	← PRG BRCH CASE CASE
THEN	← PRG BRCH CASE THEN
END	← PRG BRCH CASE END

BRCH START

START	← PRG BRCH START START
NEXT	← PRG BRCH START NEXT
STEP	← PRG BRCH START STEP

BRCH FOR

FOR	← PRG BRCH FOR FOR
NEXT	← PRG BRCH FOR NEXT
STEP	← PRG BRCH FOR STEP

BRCH DO

DO	← PRG BRCH DO DO
UNTIL	← PRG BRCH DO UNTIL
END	← PRG BRCH DO END

BRCH WHILE

WHILE	← PRG BRCH WHILE WHILE
REPEAT	← PRG BRCH WHILE REPEA
END	← PRG BRCH WHILE END

TEST

==	← PRG TEST ≠
AND	← PRG TEST NXT AND
OR	← PRG TEST NXT OR
XOR	← PRG TEST NXT XOR
NOT	← PRG TEST NXT NOT
SAME	← PRG TEST NXT SAME
SF	← PRG TEST NXT NXT SF
CF	← PRG TEST NXT NXT CF
FS?	← PRG TEST NXT NXT FS?
FC?	← PRG TEST NXT NXT FC?
FS?C	← PRG TEST NXT NXT FS?C
FC?C	← PRG TEST NXT NXT FC?C

TYPE

OBJ→	← PRG TYPE OBJ→
→ARRY	← PRG TYPE →ARRY
→LIST	← PRG TYPE →LIST
→STR	← PRG TYPE →STR
→TAG	← PRG TYPE →TAG
NUM	← PRG TYPE NXT NUM
CHR	← PRG TYPE NXT CHR
TYPE	← PRG TYPE NXT TYPE

LIST ELEM

GET	← PRG LIST ELEM GET
GETI	← PRG LIST ELEM GETI
PUT	← PRG LIST ELEM PUT
PUTI	← PRG LIST ELEM PUTI
SIZE	← PRG LIST ELEM SIZE
HEAD	← PRG LIST ELEM NXT HEAD
TAIL	← PRG LIST ELEM NXT TAIL

LIST PROC

REVLIST	← PRG LIST PROC REVLI
SORT	← PRG LIST PROC NXT SORT
SEQ	← PRG LIST PROC NXT SEQ

MODES ANGLE

DEG	← PRG NXT MODES ANGLE DEG
RAD	← PRG NXT MODES ANGLE RAD

MODES MENU

CST	← PRG NXT MODES MENU CST
MENU	← PRG NXT MODES MENU MENU
BEEP	← PRG NXT MODES MISC BEEP

IN

INFORM	← PRG NXT IN INFOR
INPUT	← PRG NXT IN INPUT

MSGBOX ← PRG NXT OUT MSGBO
PVIEW ← PRG NXT OUT PVIEW

RUN

DBUG ← PRG NXT NXT RUN DBG
SST ← PRG NXT NXT RUN SST
SST↓ ← PRG NXT NXT RUN SST↓
HALT ← PRG NXT NXT RUN HALT
KILL ← PRG NXT NXT RUN KILL

Programme zum Generieren von Zahlenlisten

Beachten Sie, dass außer den Funktionen des Menüs PRG weitere Funktionen, die Sie beim Programmieren verwenden können, zur Verfügung stehen. Im Grunde können fast alle Funktionen des Taschenrechners in ein Programm integriert werden. So können Sie beispielsweise Funktionen des Menüs MTH verwenden. Speziell für Operationen mit Listen können Sie die Listenfunktionen, wie SORT, ΣLIST usw., aus dem Menü MTH/LIST verwenden.

Als zusätzliche Programmierübung, aber auch, um die oben aufgeführten Tastenfolgen zu üben, werden nachfolgend nur drei Programme zum Erstellen oder Manipulieren von Listen gezeigt. Die Programmnamen und Listen lauten wie folgt:

LISC:

```
« → n x « 1 n FOR j x NEXT n →LIST » »
```

CRLST:

```
« → st en df « st en FOR  j j df STEP en st - df / FLOOR 1 +
→LIST » »
```

CLIST:

```
« REVLIST DUP DUP SIZE 'n' STO ΣLIST SWAP TAIL DUP SIZE 1 - 1
SWAP FOR j DUP ΣLIST SWAP TAIL NEXT 1 GET n →LIST REVLIST 'n'
PURGE »
```

Die Arbeitsweise der Programme ist wie folgt:

(1) *LISC:* Erzeugt eine Liste mit n Elementen, die alle gleich einer Konstante c sind.
Ablauf: Geben Sie nacheneinander n, dann c ein, drücken Sie anschließend LISC
Beispiel: 5 ENTER 6,5 ENTER LISC erzeugt folgende Liste: {6,5 6,5 6,5 6,5 6,5}

(2) *CRLST*: Erzeugt eine Liste mit Zahlen von n_1 bis n_2 mit einem Wertzuwachs von Δn, d.h.,{n_1, $n_1+\Delta n$, $n1+2\cdot\Delta n$, … $n_1+N\cdot\Delta n$ }, wobei N = floor((n_2-n_1)/Δn)+1.
Ablauf: geben Sie nacheinander $n_{1,}$ dann n_2, dann Δn ein und drücken Sie anschließend CRLST
Beispiel: .5 ENTER 3,5 ENTER ,5 ENTER CRLST erzeugt: {0,5 1 1,5 2 2,5 3 3,5}

(3) *CLIST*: Erzeugt eine Liste mit der kumulierten Summe der Elemente, d.h., wenn die Originalliste {x_1 x_2 x_3 … x_N} enthält, erzeugt CLIST die folgende Liste:

$$\{x_1, x_1 + x_2, x_1 + x_2 + x_3, ..., \sum_{i=1}^{N} x_i\}$$

Ablauf: Legen Sie die Originalliste in Ebene 1 ab, drücken Sie CLIST.
Beispiel: {1 2 3 4 5} ENTER CLIST erzeugt {1 3 6 10 15}.

Beispiele zum sequentiellen Programmieren

Generell bezeichnet man ein Programm als jede Sequenz von Befehlen des Taschenrechners, die von den Programm-Containern « und » eingeschlossen werden. Auch Unterprogramme können Teil eines Programms werden. Die Beispiele, die vorher in diesem Handbuch aufgeführt wurden (z.B. in Kapitel 3 und 8), können in zwei Kategorien unterteilt werden: (a) Programme, die durch Definition einer Funktion erzeugt wurden und (b) Programme, die eine Abfolge von Stack-Operationen ausführen. Diese beiden Arten von Programmen werden nachfolgend beschrieben. Generell bestehen diese Programme aus Eingabe→Verarbeitung→Ausgabe. Daher werden diese Programme als sequentielle Programme bezeichnet.

Über Definition einer Funktion generierte Programme

Hierbei handelt es sich um Programme, die mithilfe der Funktion DEFINE ([←] DEF), die mit einem Argument, das wie folgt aussieht, erstellt wurden:

'function_name(x_1, x_2, …) = Ausdruck mit den Variablen x_1, x_2, …'

Das Programm wird in der Variablen mit dem Namen `function_name` gespeichert. Wird das Programm mit der Tastenfolge [→] function_name wieder in den Stack geladen, sieht das Programm wie folgt aus:

« → x_1, x_2, … Ausdruck mit Variablen x_1, x_2, …'».

Um die eingegebenen Variablen x_1, x_2, … im RPN-Modus zu untersuchen, geben Sie einige Variablen in richtiger Reihenfolge (d.h., x_1 zuerst, dann x_2, x_3, usw.) ein und drücken dann die Funktionstaste function_name. Der Taschenrechner gibt das Ergebnis der Funktion function_name(x_1, x_2, …) aus.

<u>Beispiel</u>: *Manning-Gleichung für weite rechteckige Kanäle.*
Als Beispiel nehmen Sie die Manning-Gleichung, die Durchfluss (Durchfluss je Einheit der Breite) q in einem weiten, rechteckigen, offenen Kanal berechnet:

$$q = \frac{C_u}{n} y_0^{5/3} \sqrt{S_0}$$

wobei C_u eine Konstante, die von den verwendeten Maßeinheiten abhängt [C_u = 1.0 gilt für das International System (S.I. – metrisches System) und C_u = 1.486 gilt für das britische Maßsystem (E.S.)], n den Manning-Koeffizienten für Rauhigkeit, der von der Kanaloberfläche und anderen Faktoren abhängt, y_0 die Flusstiefe und S_0 das dimensionslose Gefälle des Kanals, darstellt.

Hinweis: Werte des Manning-Koeffizienten n können Tabellen als dimensionslose Werte entnommen werden. Sie liegen im Allgemeinen zwischen 0,001 und 0,5. Auch der Wert von C_u ist dimensionslos. Der Wert für y_0 muss jedoch die korrekte Dimension aufweisen, d.h., m für S.I. und Fuß für E.S. Das Ergebnis für q wird dann in der richtigen Maßeinheit ausgegeben, d.h., m^2/s in S.I. und ft^2/s in E.S. Die Manning-Gleichung ist daher nicht *dimensionskonsistent*.

Angenommen wir wollen die Funktion q(Cu, n, y0, S0) zur Berechnung des Durchhflusses q für diesen Fall erstellen . Verwenden Sie den Ausdruck:

'q(Cu,n,y0,S0)=Cu/n*y0^(5./3.)*√S0',

als Argument der Funktion DEFINE. Beachten Sie, dass der Exponent 5./3. in der Gleichung das Verhältnis von reellen Zahlen durch Dezimalkomma darstellt. Falls erforderlich drücken Sie VAR, um die Variablenliste zu laden. An dieser Stelle befindet sich unter den Funktionstasten eine Variable mit der Bezeichnung q. Um den Inhalt von q anzuzeigen, drücken Sie ↱ q. Das Programm, das durch die Definition der Funktion `q(Cu,n,y0,S0)` erstellt wurde, wird wie folgt angezeigt:

« → Cu n y0 S0 'Cu/n*y0^(5/3)*√S0' ».

Dies wird interpretiert als "geben Sie Cu, n, Y0, S0 nacheinander ein und berechne dann den Ausdruck". Um beispielsweise q für Cu = 1,0, n = 0,012, y0 = 2 m und S0 = 0,0001 im RPN-Modus zu berechnen, geben Sie ein:

1 ENTER 0,012 ENTER 2 ENTER 0,0001 ENTER q

Das Ergebnis ist 2,6456684 (oder q = 2,6456684 m^2/s).

Anstatt diese durch ENTER einzugeben, können die einzelnen Eingabedaten auch durch Leerzeichen getrennt in eine einzelne Zeile des Stacks eingegeben werden.

Programme zum Simulieren einer Sequenz (Folge) von Stack-Operationen

In diesem Fall gehen wir davon aus, dass sich die Glieder, die zur Folge der Operationen gehören, im Stack befinden. Der erste Schirtt im Programm ist das Öffnen eines Programm-Containers über ↱ « ». Anschließend werden die einzelnen Operationen eingegeben. Nachdem Sie alle Operationen eingegeben haben, drücken Sie ENTER, um das Programm abzuschließen. Wenn Sie das Programm nur ein einziges Mal verwenden wollen, können Sie an dieser Stelle EVAL drücken, um das Programm mit den verfügbaren Eingabedaten durchzuführen. Soll das Programm immer wieder verwendet werden, muss es unter einem Variablennamen gespeichert werden.

Die einfachste Art ein solches Programm zu beschreiben, ist anhand eines Beispiels:

Beispiel: Staudruck für einen rechteckigen Kanal
Angenommen wir wollen den Staudruck h_v in einem rechteckigen Kanal der Breite b mit einer Flusstiefe y und einer Durchflussmenge Q berechnen. Die spezifische Energie wird als $h_v = Q^2/(2g(by)^2)$ berechnet, wobei g der Erdbeschleunigung (g = 9,806 m/s^2 in S.I.-Einheiten oder g = 32,2 ft/s^2 in E.S.-Einheiten) entspricht. Wollen wir h_v für Q = 23 cfs (Kubikfuß pro Sekunde = ft^3/s), b = 3 ft und y = 2 ft berechnen, geben wir ein: $h_v = 23^2/(2\cdot 32.2\cdot (3\cdot 2)^2)$. Im RPN-Modus können wir interaktiv berechnen:

2 ENTER 3 × ← x^2 3 2 . 2 ×
2 × 2 3 ← x^2 ▶ ÷

Das Ergebnis ist 0,228174 oder h_v = 0,228174.

Um diese Berechnung als Programm aufzubauen, müssen sich die Eingabedaten (Q, g, b, y) in der Reihenfolge, in der sie bei der Berechnung verwendet werden, im Stack befinden. In Hinblick auf die Variablen Q, g, b und y wird die soeben durchgeführte Kalkulation wie folgt eingegeben (bitte nicht eintippen):

y ENTER b × ← x^2 g × 2 × Q ← x^2 ▶ ÷

Wie Sie sehen wird y zuerst verwendet, dann folgen nacheinander b, g und Q (in dieser Reihenfolge). Daher müssen wir die Variablen für diese Berechnung in umgekehrter Reihenfolge eintippen, d.h. (bitte nicht eintippen):
Q ENTER g ENTER b ENTER y ENTER

Für die einzelnen zu berücksichtigenden Werte verwenden wir:

23 ENTER 32,2 ENTER 3 ENTER 2 ENTER

Das Programm selbst enthält nur die Tastenfolgen (oder Anweisungen), die durch Entfernen der Werte aus vorangegangener Berechnung ersetzt wurden d.h. Q, g, b und y werden entfernt (bitte nicht eintippen):

y ENTER b × ← x^2 g × 2 × Q ← x^2 ▶ ÷

übrig bleiben nur noch die nachfolgend gezeigten Operationen (bitte nicht eintippen):

ENTER × ← × 2 × ← x^2 ▶ ÷

Hinweis: Verwenden Sie bei der Eingabe des Programms nicht die Taste ▶, sondern die Tastenfolge: ← PRG STACK SWAP.

Anders als bei der interaktiven Berechnung müssen wir in dem Programm die Stack-Ebenen 1 und 2 austauschen. Um das Programm zu erstellen, verwenden Sie die nachstehenden Tastenfolgen:

→ « »	Öffnen der Programmsymbole
×	Multiplizieren von y mit b
← x^2	Quadrieren von (b·y)
×	$(b·y)^2$ mit g multiplizieren
2 ×	Eingeben einer 2 und Multiplizieren mit g· $(b·y)^2$
← PRG STACK SWAP	Austauschen von Q mit 2·g· $(b·y)^2$
← x^2	Quadrieren von Q
← PRG STACK SWAP	Austauschen von 2·g· $(b·y)^2$ mit Q^2
÷	Dividieren von Q^2 durch 2·g· $(b·y)^2$
ENTER	Programm eingeben

Das Programm sieht dann wie folgt aus:

```
« * SQ * 2 * SWAP SQ SWAP / »
```

Hinweis: SQ ist die Funktion, die sich aus der Tastenfolge ← x² ergibt.

Erstellen wir nun eine Kopie des Programms und speichern dieses unter dem Variablen-Namen hv:

' ALPHA ← H ALPHA ← V STO▶

Im Funktionstastenmenü sollte jetzt eine neue Variable hv vorhanden sein. (Drücken Sie VAR, um die Variablenliste aufzurufen.) Das Programm, das sich noch im Stack befindet, kann mit der Funktion EVAL untersucht werden. Das Ergebnis sollte wie zuvor 0,228174... betragen. Das Programm ist für spätere Verwendung unter dem Namen hv verfügbar. Verwenden Sie beispielsweise für Q = 0,5 m^3/s, g = 9,806 m/s^2, b = 1,5 m und y = 0,5 m:

0,5 SPC 9,806 SPC 1,5 SPC 0,5 hv

Hinweis: SPC wird hier als Alternative zu ENTER für die Dateneingabe verwendet.

Das Resultat ist nun 2,26618623518E-2, d.h., hv = $2{,}26618623518\times10^{-2}$ m.

Hinweis: Da die in hv programmierte Gleichung dimensionskonsistent ist, können bei der Eingabe auch Einheiten verwendet werden.

Wie bereits früher erwähnt, handelt es sich bei den beiden Programmen in diesem Absatz um *sequentielle Programme*. Das bedeutet, das Programm folgt einem einzelnen Pfad, INPUT→ OPERATION →OUTPUT. Eine Verzweigung des Programms über die Befehle des Menüs ← PRG BRCH ist möglich. Weitere Details hierzu finden Sie weiter unten.

Interaktive Eingabe in Programmen

Bei den vorausgegangenen Programmbeispielen ist es für den Anwender nicht immer klar, in welcher Reihenfolge die Variablen vor der Programmausführung im Stack angeordnet sein müssen. Bei dem Programm q, geschrieben als

```
« →Cu n y0 S0 'Cu/n*y0^(5/3)*√S0' »,
```

ist es immer möglich, die Programmdefinitionen neu in den Stack (↱) zu laden, die Reihenfolge in der die Variablen eingegeben werden müssen, anzuzeigen; hier →Cu n y0 S0. Bei dem Programm bietet die Definition

« * SQ * 2 * SWAP SQ SWAP / »

keinen Hinweis auf die Eingabereihenfolge der Daten. Es sei denn, Sie sind mit RPN und der UserRPL Sprache gut vertraut.

Ein Weg, das Resultat eines Programms wie eine Formel zu überprüfen liegt darin, symbolische Variablen anstelle von numerischen Ergebnissen in den Stack einzugeben und dann das Programm mit diesen Werten arbeiten zu lassen. Hierzu muss das CAS-Modul (Calculator Algebraic System) des Taschenrechners auf `symbolic` und `exact` eingestellt sein. Dies geschieht mithilfe von MODE CAS. Vergewissern Sie sich, das die Optionen `_Numeric` und `_Approx` deaktiviert sind. Um zur Normalanzeige zurückzukehren, drücken Sie OK OK. Zum Anzeigen des Variablen-Menüs drücken Sie VAR.

Diesen Ansatz verenden wir, um festzustellen, welche Formel hinter der Verwendung des Programms verborgen ist, wie folgt: Wir wissen, dass für das Programm 4 Eingaben erforderlich sind. Daher verwenden wir die symbolischen Variablen S4, S3, S2 und S1:

ALPHA S 4 ENTER ALPHA S 3 ENTER ALPHA S 2 ENTER ALPHA S 1 ENTER

und anschließend . Die daraus resultierende Formel könnte in in etwa wie folgt aussehen

`‘SQ(S4)/(S3*SQ(S2*S1)*2)’,`

wenn Ihre Anzeige nicht auf “Textbook”-Stil eingestellt ist, oder wie

$$\frac{SQ(S4)}{S3\cdot SQ(S2\cdot S1)\cdot 2}$$

wenn der “Textbook”-Stil ausgewählt wurde. Da wir wissen, dass die Funktion SQ( ) für x^2 steht, interpretieren wird das letzte Resultat als

$$\frac{S4^2}{2 \cdot S3 \cdot (S2 \cdot S1)^2},$$

welches die Positionen der verschiedenen Eingabeebenen des Stacks in der Formel anzeigt. Wenn wir dieses Resultat mit der von uns programmierten Originalformel vergleichen, d.h.,

$$h_v = \frac{Q^2}{2g(by)^2},$$

erkennen wir, dass wir y in Stack-Ebene 1 (S1), b in Stack-Ebene 2 (S2), g in Stack-Ebene 3 (S3) und Q in Stack-Ebene 4 (S4) eingeben müssen.

Prompt mit einem Eingabestring

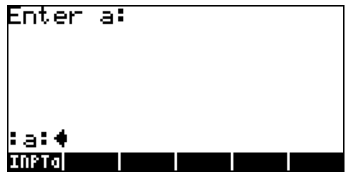
Diese beiden Ansätze sind nicht sehr effizient beim Bestimmen der Reihenfolge für die Eingabedaten. Sie sind dem Anwender zumindest durch die Anzeige der Variablen-Namen, die er eingeben muss, nützlich. Die einfachste von der UserRPL-Sprache zur Verfügung gestellte Methode ist das Verwenden von Zeichenketten und der Funktion INPUT (← PRG NXT IN INPUT) zum Laden der Eingabedaten.

Das folgende Programm fordert den Anwender auf, einen Wert für die Variable a einzugeben und legt diesen dann in die Stack-Ebene 1 ab.

```
« “Enter a: “ {“↵:a: “ {2 0} V } INPUT OBJ→ »
```

Das Programm enthält die Symbole :: (tag) und ↵(return), verfügbar über die Tastenfolgen ← :: und → ↵, zusammen mit der Taste •. Das Symbol Tag (::) dient dazu, Zeichenketten für die Ein- und Ausgabe zu markieren. Das Symbol Return (↵) entspricht dem Drücken der Eingabetaste auf einem Computer. Die Zeichenketten zwischen den Anführungszeichen (“ “) werden direkt über die alphanumerische Tastatur eingegeben.

Speichern Sie das Programm in eine Variable mit der Bezeichnung INPTa (für INPuT a).
Starten Sie das Programm durch Drücken der Funktionstaste INPTa.



Als Resultat wird der Anwender aufgefordert einen Wert für a einzugeben. Der Cursor wird direkt rechts von dem Prompt :a: gesetzt. Geben Sie beispielsweise 35 ein und drücken ENTER. Als Resultat erscheint der Eingabestring `:a:35` in Stack-Ebene 1.

Funktion mit Eingabestring

Wenn Sie den oben erwähnten Code beispielsweise zum Berechnen der Funktion f(a) = 2*a^2+3 eingeben wollen, können Sie das Programm folgendermaßen modifizieren:

```
« “Enter a: “ {“↵:a: “ {2 0} V }
INPUT OBJ→ → a « ‘2*a^2+3‘ » »
```

Speichern Sie dieses neue Programm unter dem Namen ‘FUNCa’ (FUNCtion von a):

Starten Sie das Programm durch Drücken von FUNCa. Sobald Sie dazu aufgefordert werden, geben Sie z.B. 2 ein und drücken dann ENTER. Das Ergebnis ist der algebraische Begriff $2a^2+3$. Dies ist aber falsch. Der Taschenrechner bietet Ihnen Funktionen zum Überprüfen Ihres Programms und Auffinden logischer Fehler während der Programmausführung.

Fehlersuche im Programm

Um die Fehlerursache herauszufinden, verwenden wir die Funktion DBUG wie folgt:

' FUNCa ENTER	Kopiert den Programmnamen in Stack-Ebene 1
← PRG NXT NXT RUN DBG	Startet die Fehlersuche (den Debugger)
SST↓	Fehlersuche Schritt für Schritt, Ergebnis: "Enter a:"
SST↓	Ergebnis: {" ↵ a:" {2 0} V}
SST↓	Ergebnis: der Anwender wird aufgefordert, einen Wert für a einzugeben
2 ENTER	Geben Sie den Wert 2 für a ein. Ergebnis: "↵:a:2"
SST↓	Ergebnis: a:2
SST↓	Ergebnis: Stack leeren, Ausführung von →a
SST↓	Ergebnis: Stack leeren, ins Unterprogramm springen «
SST↓	Ergebnis: '2*a^2+3'
SST↓	Ergebnis: '2*a^2+3', Unterprogramm verlassen »
SST↓	Ergebnis: '2*a^2+3', Hauptprogramm verlassen »

Ein erneutes Drücken der Funktionstaste SST↓ führt zu keinem weiteren Ergebnis, da wir das gesamte Programm Schritt für Schritt durchlaufen sind. Diese Fehlersuche brachte keinerlei Anhaltspunkte, warum das Programm nicht den Wert von $2a^2+3$ für a = 2 berechnet. Um zu sehen, welchen Wert a im Unterprogramm hat, müssen wird den Debugger neu starten und den Wert von a im Unterprogramm berechnen. Versuchen Sie folgendes:

VAR	Lädt das Variablenmenü
' FUNCa ENTER	Kopiert den Programmnamen in die Stack-Ebene 1
← PRG NXT NXT RUN DBG	Startet die Fehlersuche (den Debugger)
SST↓	Fehlersuche Schritt für Schritt, Ergebnis: "Enter a:"
SST↓	Ergebnis: {" ↵a:" {2 0} V}

SST↓	Ergebnis: Anwender wird aufgefordert, für a einen Wert einzugeben
2 ENTER	Geben Sie den Wert 2 für a ein. Ergebnis: "↵a:2"
SST↓	Ergebnis: a:2
SST↓	Ergebnis: Stack leeren, Ausführung von →a
SST↓	Ergebnis: Stack leeren, ins Unterprogramm springen «

An dieser Stelle befinden wir uns im Unterprogramm « '2*a^2+3' », das die lokale Variable a verwendet. Um den Wert von a anzuzeigen, verwenden Sie:

ALPHA ← A EVAL	Tatsächlich wird für die lokale Variable a = 2 angezeigt.

Brechen wir hier den Debugger ab, da wir das Ergebnis ja bereits kennen. Um den Debugger abzubrechen, drücken Sie KILL. Es erscheint die Meldung <!> Interrupted zur Bestätigung des abgebrochenen Prozesses. Um zur Normalanzeige des Taschenrechners zurückzukehren drücken Sie ON.

Anmerkung: Immer wenn Sie im Debugg-Modus SST↓ drücken, erscheint oben links in der Anzeige der auszuführende Programmschritt. Eine Funktionstaste SST ist auch als Teil des Menüs PRG im Untermenü RUN verfügbar. Diese Funktion kann dazu verwendet werden, sofort ein Unterprogramm, das von einem Hauptprogramm aus aufgerufen wird, auszuführen. Beispiele zur Anwendung von SST folgen später.

Programm korrigieren

Die einzig mögliche Erklärung dafür, dass das Programm kein numerisches Ergebnis ausgibt, scheint auf das Fehlen des Befehls →NUM nach dem algebraischen Ausdruck '2*a^2+3' zurückzuführen zu sein. Wir editieren das

Programm und fügen die fehlende Funktion EVAL ein. Das bearbeitete Programm sieht dann wie folgt aus:

```
« "Enter a: " {"↵a: " {2 0} V } INPUT
   OBJ→ → a « '2*a^2+3' →NUM » »
```

Speichern Sie das Programm wieder unter FUNCa und starten Sie es mit a = 2. Dieses Mal ist das Ergebnis 11, d.h., $2*2^2+3 = 11$.

Eingabestring für zwei oder drei Eingabewerte

In diesem Abschnitt werden wir innerhalb des HOME-Verzeichnisses ein Unterverzeichnis, das Beispiele für Eingabestrings mit ein, zwei oder drei Werten enthält, erstellen. Dies sind allgemeine Eingabestrings, die in zukünftige Programme integriert werden können, unter Berücksichtigung, dass sich die Variablennamen entsprechend den Anforderungen der einzelnen Programme ändern können.

Erstellen wir als erstes ein Unterverzeichnis mit der Bezeichnung PTRICKS (Programmier-TRICKS), in das wir nützliche Programmkomponenten ablegen, die wir in späteren komplexeren Programmen verwenden können. Zum Erstellen des Verzeichnisses wechseln Sie zunächst in das HOME-Verzeichnis. Zum Anlegen des nachfolgenden Unterverzeichnisses PTRICKS verwenden Sie im HOME-Verzeichnis die nachstehende Tastenfolge:

' ALPHA ALPHA P T R I C K S ENTER	Geben Sie den Verzeichnisnamen 'PTRICKS' ein
← PRG MEM DIR CRDIR	Erstellen Sie das Verzeichnis
VAR	Laden Sie die Variablenliste

Ein Programm kann mehr als 3 Eingabewerte benötigen. Wenn wir Eingabestrings verwenden, wollen wir die Anzahl der Werte auf 5 beschränken, da wir normalerweise nur 7 Stack-Ebenen sehen können. Wenn wir die Stack-Ebene 7 zum Eingeben eines Namens für den Eingabestring verwenden und die Ebene 6 frei halten, um die Anzeige einfach lesen zu können, bleiben zur Definition von Eingabevariablen nur noch die Stack-Ebenen 1 bis 5 übrig.

Programm mit Eingabestring für zwei Eingabewerte

Ein Programm für zwei Eingabewerte, beispielsweise a und b, sieht wir folgt aus:

```
« "Enter a and b: " {"↵:a:↵:b: " {2 0} V } INPUT OBJ→ »
```

Dieses Programm kann durch Modifizieren des Inhalts von INPTa leicht erstellt werden. Speichern Sie dieses Programm in die Variable INPT2.

Anwendung: Berechnen einer Funktion mit zwei Variablen

Betrachten Sie das Gesetz über Ideale Gase pV = nRT, wobei p = Gasdruck (Pa), V = Gasvolumen(m^3), n = Mol (gmol), R = universelle Gaskonstante = 8,31451_J/(gmol*K) und T = absolute Temperatur (K) darstellen.

Wir können den Druck p als Funktion von zwei Variablen V und T als p(V,T) = nRT/V für eine gegebene Gasmasse definieren, da n auch konstant bleibt. Gehen wir davon aus, dass n = 0,2 gmol ist, ist die Funktion die wir programmieren wollen,

$$p(V,T) = 8.31451 \cdot 0.2 \cdot \frac{T}{V} = (1.662902\_\frac{J}{K}) \cdot \frac{T}{V}$$

Die Funktion kann durch Eingeben des folgenden Programms festgelegt werden:

```
« → V T '(1.662902_J/K)*(T/V)' »
```

und dieses dann in die Variable **p** speichern.

Als nächstes muss der Eingabestring hinzugefügt werden, der den Anwender auffordert, die Werte für V und T einzugeben. Hierzu modifizieren Sie das Programm in **p** wie folgt:

```
« "Enter V and T: " {"↵ :V:↵ :T: " {2 0} V }
 INPUT OBJ→ → V T '(1.662902_J/K)*(T/V)' »
```

Speichern Sie das modifizierte Programm wieder unter [p]. Drücken Sie [p] , um das Programm zu starten. Geben Sie für V = 0.01_m^3 und für T = 300_K ein und drücken dann [ENTER]. Als Ergebnis erscheint 49887.06_J/m^3. Die Einheit J/m^3 entspricht der Einheit Pascals (Pa), die im S.I.-System bevorzugte Maßeinheit für Druck.

> **Hinweis**: Da wir absichtlich bei der Definition der Funktion Einheiten verwendet haben, müssen auch die Werte zum Erzielen eines korrekten Ergebnisses mit Einheiten eingegeben werden.

Programm mit Eingabestring für drei Eingabewerte

Ein Programm für drei Eingabewerte, beispielsweise a, b und c, sieht wir folgt aus:

```
« "Enter a, b and c:" {"↵ :a:↵ :b:↵ :c:" {2 0} V }
                   INPUT  OBJ→ »
```

Dieses Programm kann durch Modifizieren des obigen Inhalts von INPT2 leicht erstellt werden. Speichern Sie das Programm unter der Bezeichnung INPT3. Mit diesem Programm ist unsere Sammlung von Programmen, die dem Anwender ermöglichen, einen, zwei oder drei Werte einzugeben, vervollständigt. Speichern Sie diese Programme als Referenz und kopieren und modifizieren Sie diese für neue Programme, die Sie erstellen.

Anwendung: Berechnen einer Funktion mit drei Variablen

Nehmen wir an, dass wir das Programm für das Gesetz über Ideale Gase so modifizieren wollen, dass auch n für Mol als zusätzliche Variable eingeschlossen wird. Das bedeutet, dass wir die nachfolgende Funktion wie folgt definieren

$$p(V,T,n) = (8.31451\_\frac{J}{K})\frac{n \cdot T}{V},$$

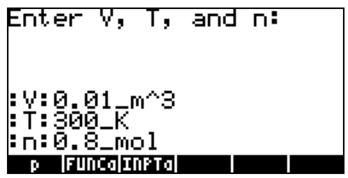
und modifizieren wollen, dass drei verschiedene Variablen eingegeben werden können. Die Vorgehensweise ähnelt der, die wir schon bei der

Definition der Funktion p(V.T) angewendet haben. Das Programm sieht dann wie folgt aus:

```
« "Enter V, T, and n:" {" ↵ :V:↵ :T:↵ :n:" {2 0} V }
INPUT OBJ→ →V T n  '(8.31451_J/(K*mol))*(n*T/V)' »
```

Speichern Sie dieses Programm in der Variablen p. Um das Programm zu starten, drücken Sie p.

Geben Sie für V = 0.01_m^3, für T = 300_K und für n = 0,8_mol ein. Bevor Sie ENTER drücken, sieht der Stack folgendermaßen aus:



Drücken Sie ENTER. Das Resultat ist 199548,24_J/m^3, oder 199548,24_Pa = 199,55 kPa.

Eingabe über Eingabemasken

Die Funktion INFORM (← PRG NXT IN INFOR.) kann zum Erstellen detaillierter Eingabemasken für ein Programm verwendet werden. Die Funktion INFORM benötigt 5 Argumente in nachstehender Reigenfolge:

1. Einen Titel: Eine Zeichenkette zur Beschreibung der Eingabemaske
2. Felddefinitionen: Eine Liste mit einer oder mehreren Felddefinitionen $\{s_1\ s_2\ \ldots\ s_n\}$, wobei jede Felddefinition, s_i, ein von zwei Formaten aufweisen kann:
    a. Eine einfache Feldbezeichnung: Eine Zeichenkette
    b. Eine Liste mit einer oder mehreren Felddefinitionen {"label" "helpInfo" $type_0$ $type_1$ … $type_n$}. "label" ist eine Feldbezeichnung. "helpInfo" ist eine Zeichenkette zur detaillierten Beschreibung des Feldes und die "Type"-

Spezifikation ist eine Liste von Variablentypen, die in dem Feld erlaubt sind (siehe Objekttypen im Kapitel 24).

3. Information zum Feldformat: Eine einzelne Zahl *col* oder eine Liste {*col tabs*}. In dieser Spezifikation ist *col* die Anzahl der Spalten in der Eingabebox und *tabs* (optional) legt die Anzahl der Tab-Stops der Labels und die Felder der Maske fest. Die Liste kann eine leere Liste sein. Vorgabewert sind *col* = 1 und *tabs* = 3.
4. Liste mit Reset-Werten: Eine Liste mit Werten, welche bei Auswahl der Option RESET die verschiedenen Felder während der Arbeit mit einer Eingabemaske zurücksetzt.
5. Liste mit Anfangs-Werten: Eine Liste, welche die Anfangs-Werte der Felder enthält.

Die Listen unter 4 und 5 können leer sein. Außerdem können Sie den Befehl NOVAL (← PRG NXT IN NOVAL) verwenden, wenn für diese Optionen keine Werte auszuwählen sind.

Ist die Funktion INFORM aktiviert und Sie haben die Option CANCEL eingegeben, erhalten Sie eine Null oder eine Liste mit den in die Felder eingegebenen Werte in der spezifizierten Reihenfolge sowie die Zahl 1, im RPN-Stack:

```
2:   {v1 v2 … vn}
1:              1
```

Wenn also der Wert in der Stack-Ebene gleich 0 ist, wurde keine Eingabe vorgenommen, während eine 1 anzeigt, dass die Eingabewerte in der Stack-Ebene 2 zur Verfügung stehen.

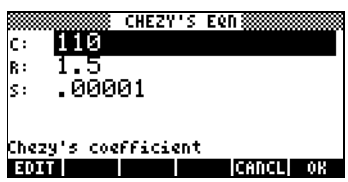
Beispiel 1 - Als Beispiel betrachten Sie das folgende Programm INFP1 (INput Form Program 1), um den Durchfluss Q in einem offenen Kanal nach der Chezy-Formel zu berechnen: $Q = C\cdot(R\cdot S)^{1/2}$, wobei C der Chezy-Koeffizient, eine Funktion aus der Rauhigkeit der Kanaloberfläche (typische Werte 80 - 150) ist, R ist der hydraulische Radius des Kanals (eine Länge) und S die Neigung des Kanalbetts (eine dimensionslose Zahl von 0,01 bis 0,000001). Das folgende Programm definiert eine Eingabemaske über die Funktion INFORM:

```
« " CHEZY'S EQN" { { "C:" "Chezy's coefficient" 0} { "R:"
"Hydraulic radius" 0 } { "S:" "Channel bed slope" 0} } { }
{ 120 1 .0001} { 110 1.5 .00001 }  INFORM »
```

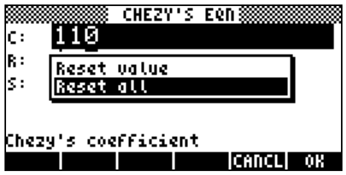
Im Programm sind die oben angeführten 5 Komponenten wie folgt:

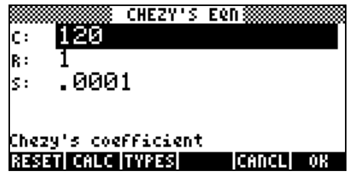
1. Titel: `" CHEZY'S EQN"`
2. Felddefinitionen: Hier haben wir drei mit den Labels "C:", "R:", "S:", die Infostrings "Chezy coefficient", "Hydraulic radius", "Channel bed slope", und die Beschränkung der Eingabe auf den Datentyp 0 (reelle Zahlen) für alle drei Felder:
   ```
   { { "C:" "Chezy's coefficient" 0}
   { "R:" "Hydraulic radius" 0 }
   { "S:" "Channel bed slope" 0} }
   ```
3. Information zum Feldformat: `{ }` (eine leere Liste, es werden die Vorgabewerte verwendet)
4. Liste mit Reset-Werten: `{ 120 1 .0001}`
5. Liste mit Anfangs-Werten: `{ 110 1.5 .00001}`

Speichern Sie dieses Programm in die Variable INFP1. Drücken Sie INFP1, zum Starten des Programms. Die Eingabemaske mit den geladenen Anfangswerten sieht wie folgt aus:

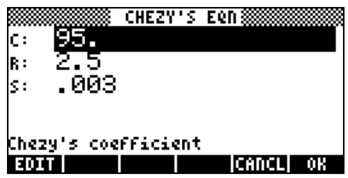


Um zu verdeutlichen, was mit diesen Werten beim Zurücksetzen geschieht, drücken Sie NXT RESET (wählen Sie *Reset all*, um die Werte der Felder zurückzusetzen):





Geben Sie nun verschiedene Werte für die drei Felder ein, sagen wir C = 95, R = 2,5 und S = 0,003. Drücken Sie nach jedem neuen Wert OK. Nach dieser Änderung sieht die Eingabemaske folgendermaßen aus:



Um nun die Werte in das Programm zu übertragen, drücken Sie noch einmal OK. Hierdurch wird die Funktion INFORM aktiviert und im Stack erscheint:

Soweit die Demonstration der Verwendung von INFORM. Um zu sehen, wie diese Eingabewerte in einer Berechnung verwendet werden können, modifizieren wir das Programm wie folgt:

```
« " CHEZY'S EQN" { { "C:" "Chezy's coefficient" 0} { "R:"
"Hydraulic radius" 0 } { "S:" "Channel bed slope" 0} } { }
{ 120 1 .0001} { 110 1.5 .00001 }  INFORM IF THEN OBJ→ DROP →
C R S 'C*(R*S)' →NUM "Q" →TAG ELSE "Operation cancelled"
MSGBOX END »
```

Die oben gezeigten Programmschritte zeigen nach dem Befehl INFORM eine bedingte Verzweigung mithilfe von IF-THEN-ELSE-END (eine detaillierte Beschreibung finden Sie an anderer Stelle in diesem Kapitel). Das Programm kann abhängig vom Wert in der Stack-Ebene 1 zu einer von zwei

Möglichkeiten springen. Ist der Wert 1, springt das Programm zu den Befehlen:

```
OBJ→ DROP → C R S 'C*√(R*S)' →NUM "Q" →TAG
```

Diese Befehle errechnen Q und setzen einen Tag auf Q. Befindet sich aber der Wert 0 in Stack-Ebene 1 (was der Fall ist, wenn CANCEL beim Verwenden der Eingabemaske gedrückt wird) , springt das Programm zu den Befehlen:

```
"Operation cancelled" MSGBOX
```

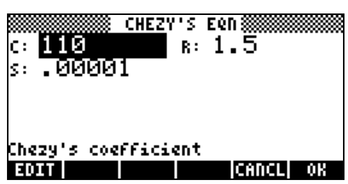
Diese Befehle erzeugen die Meldung, dass die Operation abgebrochen wurde.

Hinweis: Die Funktion MSGBOX gehört zu einer Sammlung von Ausgabefunktionen im Menü PRG/OUT. Die Befehle Commands IF, THEN, ELSE, END finden Sie im Menü PRG/BRCH/IF. Die Funktionen OBJ→, →TAG befinden sich im Menü PRG/TYPE. DROP steht unter PRG/STACK zur Verfügung. Die Funktionen → und →NUM finden Sie auf der Tastatur.

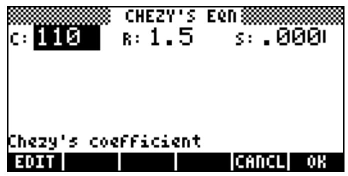
Beispiel 2 – Um die Verwendung der 3. Funktion (Informationen zum Feldformat) der Argumente, die Funktion INFORM zu illustrieren, ändern Sie die leere Liste des Programms INFP1 in { 2 1 }. Dies bedeutet, dass nur 2 Spalten anstelle der Standardvorgabe von 3 und nur 1 Tab-Stop verwendet werden. Speichern Sie dieses neue Programm in die Variable INFP2:

```
« " CHEZY'S EQN" { { "C:" "Chezy's coefficient" 0} { "R:"
"Hydraulic radius" 0 } { "S:" "Channel bed slope" 0} } { 2 1 }
{ 120 1 .0001} { 110 1.5 .00001 }  INFORM IF THEN OBJ→ DROP →
C R S 'C*(R*S)' →NUM "Q" →TAG ELSE "Operation cancelled"
MSGBOX END »
```

Starten Sie das Programm INFP2, erhalten Sie die folgende Eingabemaske:



Beispiel 3 - Ändern Sie das Feldformat in { 3 0 } und speichern Sie das Programm als INFP3. Starten Sie das Programm und schauen Sie sich die neue Eingabemaske an:



Erstellen einer Auswahlbox

Die Funktion CHOOSE (← PRG NXT IN CHOOS) bietet dem Anwender die Möglichkeit, eine Auswahlbox in ein Programm zu integrieren. Diese Funktion benötigt drei Argumente:

1. Ein Prompt (eine Zeichenkette, die die Auswahlbox beschreibt)
2. Eine Liste der Auswahldefinitionen {c_1 c_2 … c_n}. Eine Auswahldefinition c_i kann eine der beiden unten aufgeführten Formate haben:
   a. Ein Objekt, z.B. eine Zahl, ein algebraischer Ausdruck, das in der Auswahlbox angezeigt wird und als Auswahlergebnis übernommen wird.
   b. Eine Liste {object_displayed object_result}, die so aufgebaut ist, dass object_displayed in der Auswahlbox angezeigt wird und object_result als Ausgbe für das Ergebnis übernommen wird.
3. Eine Zahl, die die Position der Standardauswahl in einer Auswahlliste anzeigt. Ist dieser Wert 0, ist keine Standardauswahl hervorgehoben.

Die Aktivierung der CHOOSE-Funktion gibt entweder Null, wenn CANCEL gewählt wurde, oder wenn eine Auwahl getroffen wurde, zurück (z.B. v) sowie die Zahl 1, d.h. im RPN-Stack:

```
2:            v
1:            1
```

<u>Beispiel 1</u> – Die Manning-Gleichung zur Kalkulation der Durchfluss-geschwindigkeit in einem offenen Kanal verwendet einen von den

verwendeten Einheiten abhängigen Koeffizienten C_u. Verwenden Sie das S.I. (Systeme International), dann ist C_u = 1,0. Verwenden Sie jedoch das E.S. (English System), dann ist C_u = 1,486. Das folgende Programm verwendet eine Auswahlbox zum Festlegen des zu verwendenden Maßsystems von C_u. Speichern Sie es in die Variable CHP1 (CHoose-Programm 1):

```
« "Units coefficient" { { "S.I. units" 1}
{ "E.S. units" 1.486}  } 1 CHOOSE »
```

Starten Sie das Programm (drücken Sie CHP1), so erscheint die folgende Auswahlbox:

Abhängig von, ob Sie `S.I. Maßeinheiten` oder `E.S. Maßeinheiten` vorwählen, legt Funktion CHOOSE, entweder einen Wert von 1 oder einen Wert von 1,486 in Stapelniveau 2 und 1 in Niveau 1. Brechen Sie die Auswahlbox ab, gibt CHOICE eine Null (0) zurück.

Die durch die Funktion CHOOSE zurückgegeben Werte können von anderen Programmbefehlen, wie im modifizierten Programm CHP2 gezeigt, verwendet werden.

```
« "Units coefficient" { { "S.I. units" 1} { "E.S. units"
1.486}  } 1 CHOOSE IF THEN "Cu" →TAG ELSE "Operation
cancelled" MSGBOX END »
```

Die auf die Funktion CHOOSE folgenden IF-THEN-ELSE-END-Anweisungen besagen, dass in diesem Prgramm eine Entscheidung anhand des in der Stack-Ebene 1 abgelegten Wertes getroffen wird. Ist der Wert in der Stack-Ebene 1 gleich 1, erzeugen die Anweisungen `"Cu" →TAG` ein gekennzeichnetes (tagged) Ergebnis auf dem Bildschirm. Ist der Wert 0 erscheint in der Anzeige die Meldung `"Operation cancelled" MSGBOX`, was bedeutet, dass die Operation abgebrochen wurde.

Identifizieren der Ausgabe von Programmen

Der einfachste Weg numerische Ausgaben von Programmen zu identifizieren ist, diese Ergebnisse zu kennzeichnen. Bei einem "Tag" (Kennzeichnung) handelt es sich einfach um einen String, der an eine Zahl oder an ein Objekt angehängt wird. Der String bekommt einen Namen, der dem Objekt entspricht. Als wir beispielsweise weiter oben in den Programmen INPTa (oder INPT1) und INPT2 nach Fehlern gesucht haben, erhielten wir als Ergebnis numerische Ausgaben wie `:a:35.`

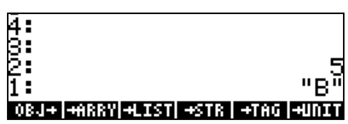
Markieren eines numerischen Ergebnisses

Um ein numerisches Ergebnis zu kennzeichnen, müssen Sie Zahl und Kennzeichnungs-String in Stack-Ebene 2 ablegen. Verwenden Sie dann die Funktion →TAG (⬅ PRG TYPE →TAG). Um beispielsweise das Ergebnis `B:5.` zu erhalten, geben Sie Folgendes ein:

5 ENTER ➡ "" ALPHA B ⬅ PRG TYPE →TAG

Aufspalten eines gekennzeichneten Ergebnisses in eine Zahl und einen Tag

Zum Aufspalten eines gekennzeichneten Ergebnisses in die Zahl und den dazugehörigen Tag verwenden Sie einfach die Funktion OBJ→ (⬅ PRG TYPE OBJ→). Durch →OBJ bleibt der numerische Wert in der Stack-Ebene 2 und der Tag (Kennzeichnung) wird in die Stack-Ebene 1 abgelegt. Wollen Sie nur den numerischen Wert weiter verwenden, können Sie den Tag mithilfe der Rücktaste ⬅ löschen. So produziert beispielsweise die Aufspaltung der gekennzeichneten Größe B:5 (siehe oben) folgendes Ergebnis:



"Extrahieren" einer gekennzeichneten Größe

Mit "Extrahieren" ist das Herausziehen des Objekts aus einer gekennzeichneten Größe gemeint. Die Funktion wird mit der nachstehenden

Tastenfolge aufgerufen: ← PRG TYPE NXT DTAG. So gibt DTAG beispielsweise bei einer gekennzeichneten Größe `a:2` den Wert 2 zurück.

Hinweis: Bei mathematischen Operationen mit gekennzeichneten Größen extrahiert der Taschenrechner die numerischen Werte automatisch. So zeigt beispielsweise die linke Abbildung zwei gekennzeichnete Größen vor und die rechte nach Drücken von × im RPN-Modus:

Beispiele für gekennzeichnete Ausgaben

Beispiel 1 – mit FUNCa gekennzeichnete Ausgabe

Wir wollen die vorher definierte Funktion FUNCa so ändern, dass damit gekennzeichnete Ausgaben erstellt werden. Holen Sie mit → FUNCa den Inhalt der FUNCa in den Stack. Die Originalfunktion sieht folgendermaßen aus:

```
« "Enter a: " {"↵:a: " {2 0} V } INPUT OBJ→ → a «
            '2*a^2+3' →NUM »»
```

Modifizieren Sie sie wie folgt:

```
« "Enter a: " {"↵:a: " {2 0} V } INPUT OBJ→ → a «
        '2*a^2+3' →NUM "F" →TAG »»
```

Speichern Sie das modifizierte Programm wieder mithilfe von ← FUNCa unter FUNCa. Starten Sie das Programm durch Drücken von FUNCa. Geben Sie, wenn Sie hierzu aufgefordert werden, den Wert 2 ein, und drücken ENTER. Das Resultat ist nun ein gekennzeichnetes Ergebnis `F:11` .

Beispiel 2 - Gekennzeichnete Ein- und Ausgabe mit FUNCa

In diesem Beispiel modifizieren wir FUNCa so dass nicht nur das berechnete Ergebnis sondern auch eine Kopie der Eingabe gekennzeichnet ausgegeben wird.

Rufen Sie mit ⓡ FUNCa den Inhalt von FUNCa in den Stack.

```
« "Enter a: " {"↵:a: " {2 0} V } INPUT OBJ→ → a «
          '2*a^2+3' →NUM "F" →TAG » »
```

Modifizieren Sie sie wie folgt:

```
« "Enter a: " {"↵:a: " {2 0} V } INPUT OBJ→ → a «
          '2*a^2+3' EVAL "F" →TAG a SWAP» »
```

(Beachten Sie, dass die Funktion SWAP über die Tastenfolge ⓛ PRG STACK SWAP) aufgerufen wird. Speichern Sie das modifizierte Programm wieder mithilfe von ⓛ FUNCa unter FUNCa. Starten Sie das Programm durch Drücken von FUNCa. Geben Sie, wenn Sie hierzu aufgefordert werden, den Wert 2 ein, und drücken ENTER. Als Ergebnis erhalten Sie die gekennzeichneten Zahlen `a:2.` in Stack-Ebene 2 und `F:11.` in Stack-Ebene 1.

Hinweis: Da wir einen Eingabestring für die Eingabe der Daten verwenden, speichert die lokale Variable a einen gekennzeichneten Wert (in obigem Beispiel `:a:2`). Daher müssen wir die Eingabe nicht markieren. Wir müssen nur in das Unterprogramm ein `a` vor die SWAP-Funktion einfügen und die gekennzeichnete Eingabe wird in den Stack geschrieben. Es soll noch mal darauf hingewiesen werden, dass bei der Kalkulation einer Funktion die Markierung der Eingabe automatisch weggelassen und nur deren numerischer Wert bei der Berechnung verwendet wird.

Um sich die Funktion FUNCa Schritt für Schritt anzuschauen, können Sie die Funktion DBUG wie folgt verwenden:

' FUNCa ENTER	Kopiert den Programmnamen in Stack-Ebene 1
ⓛ PRG NXT NXT RUN DBG	Startet die Fehlersuche (den Debugger)
SST↓	Fehlersuche Schritt für Schritt, Ergebnis: "Enter a:"
SST↓	Ergebnis: {" ↵a:" {2 0} V}
SST↓	Ergebnis: Anwender wird aufgefordert, für a

	einen Wert einzugeben
2 ENTER	Geben Sie 2 für a ein. Ergebnis: "↵a:2"
SST↓	Ergebnis: a:2
SST↓	Ergebnis: Stack leeren, Ausführung von →a
SST↓	Ergebnis: Stack leeren, ins Unterprogramm springen «
SST↓	Ergebnis: '2*a^2+3'
SST↓	Ergebnis: Stack leeren, ins Unterprogramm springen
SST↓	Ergebnis: 11.,
SST↓	Ergebnis: "F"
SST↓	Ergebnis: F: 11.
SST↓	Ergebnis: a:2.
SST↓	Ergebnis: Austausch von Ebene 1 und 2
SST↓	Unterprogramm verlassen »
SST↓	Hauptprogramm verlassen »

Beispiel 3 - Gekennzeichnete Ein- und Ausgabe mit Funktion p(V,T)
In diesem Beispiel modifizieren wir das Programm p so, dass Eingabe, Ausgabe und Ergebnis gekennzeichnet werden. Rufen Sie mit ↱ p den Inhalt des Programms in den Stack.

```
« "Enter V, T, and n:" {"↵ :V:↵ :T:↵ :n:" {2 0} V }
INPUT OBJ→ →V T n  '(8.31451_J/(K*mol))*(n*T/V)' »
```

Modifizieren Sie sie wie folgt:

```
« "Enter V, T and n: " {"↵ :V:↵ :T:↵ :n:"  {2 0} V }
            INPUT OBJ→ →V T n « V T n
  '(8.31451_J/(K*mol))*(n*T/V)' EVAL "p" →TAG » »
```

Hinweis: Beachten Sie, dass wir die Berechnung und Markierung der Funktion p(V,T,n) in ein Unterprogramm gelegt [die Befehlssequenz, die sich innerhalb der inneren Programmsymbole « » befindet] und den Aufruf der Variablen V, T und n vorangestellt haben. Dies ist erforderlich, da ohne die

Trennung der beiden Variablenlisten (V T N « V T n) durch Programmsymbole das Programm davon ausgeht, dass der Eingabebefehl

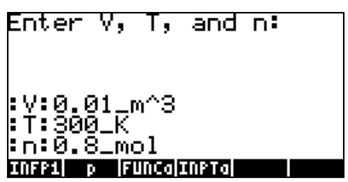
→V T N V T n

sechs Eingabewerte erfordert, obwohl nur drei vorhanden sind. Es würde zu einer Fehlermeldung kommen und die Programmausführung würde abgebrochen werden.

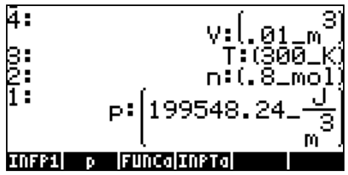
Um das Unterprogramm in die modifizierte Version des Programms einzufügen, müssen wir sowohl am Anfang als auch am Ende des Unterprogramms « » verwenden. Da die Programmsymbole beim Aufruf von « » immer paarweise auftreten, müssen wir das Abschlusssymbol am Anfang (») und das Anfangssymbol am Ende («) des Unterprogramms löschen.

Um irgendein Zeichen während der Bearbeitung eines Programms zu löschen, setzen Sie den Cursor rechts hinter das Zeichen und löschen Sie dies mit der Rückschritt-Taste .

Speichern Sie das Programm mit der Tastenfolge in die Variable p. Starten Sie das Programm durch Drücken der Taste . Geben Sie, wenn Sie dazu aufgefordert werden, für V = 0,01_m^3, für T = 300_K und für n = 0,8_mol ein. Bevor Sie die Taste ENTER drücken, sieht der Stack folgendermaßen aus:



Nach Ausführung des Programms erscheint der Stack wie folgt:



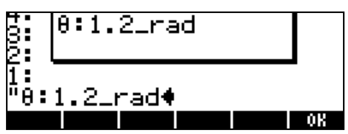
Zusammenfassung: Die Verwendung von Markierungen, um Ein- und Ausgabevariable zu identifizieren, zieht sich wie ein Faden durch alle drei Beispiele. Verwenden wir einen Eingabe-String für den Eingabewert, werden diese vorgekennzeichnet und können einfach wieder zur Ausgabe in den Stack geladen werden. Der Befehl →TAG erlaubt es uns, die Ausgabe eines Programms zu identifizieren.

Verwenden von Meldefenstern

Eine vornehmere Art, die Ausgabe eines Programms anzuzeigen, ist über Meldefenster. Der Befehl für ein Meldefenster wird durch die Tastenfolge ← PRG NXT OUT MSGBOX aufgerufen. Um den Ausgabe-String in einem Fenster anzuzeigen, muss sich dieser in Stack-Ebene 1 befinden. Um die Funktion des Befehls MSGBOX kennen zu lernen, versuchen Sie nachfolgende Übung:

→ "" ALPHA → T ALPHA ← :: 1 . 2
→ _ ALPHA ← R ALPHA ← A ALPHA ← D
← PRG NXT OUT MSGBOX

Als Ergebnis sehen Sie folgendes Meldefenster:



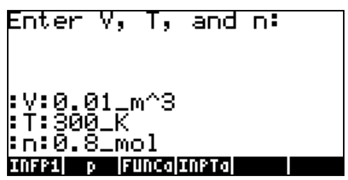
Drücken Sie OK zum Löschen des Meldefensters.

Sie können ein Meldefenster zur Anzeige einer gekennzeichneten Ausgabe eines Programms verwenden, indem Sie diese Ausgabe in einen String für MSGBOX umwandeln. Um ein gekennzeichnetes Ergebnis, eine Formel oder einen nicht gekennzeichneten Wert in einen String umzuwandeln, verwenden Sie die Funktion →STR, verfügbar über die Tastenfolge ← PRG TYPE →STR.

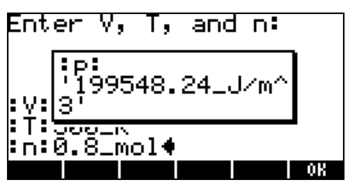
Verwenden eines Meldefensters zur Programmausgabe
Die Funktion p aus dem letzten Beispiel kann folgendermaßen geändert werden:

```
« "Enter V, T and n: " {"↵ :V:↵ :T:↵ :n: " {2 0} V }
INPUT OBJ→ →V T n « V T n
'(8.34451_J/(K*mol))*(n*T/V)' EVAL "p" →TAG →STR MSGBOX »
»
```

Speichern Sie das Programm mit der Tastenfolge ← p in die Variable p. Starten Sie das Programm durch Drücken der Taste p. Geben Sie, wenn Sie dazu aufgefordert werden, für V = 0.01_m^3, für T = 300_K und für n = 0,8_mol ein.
Wie in der älteren Version von p sieht der Stack vor Drücken der Taste ENTER folgendermaßen aus:



Bei der ersten Programmausgabe handelt es sich um ein Meldefenster mit folgendem String:



Drücken Sie OK zum Löschen des Meldefensters. Im Stack steht nun Folgendes:

Ein- und Ausgabe in einem Meldefenster anzeigen

Wir können das Programm so modifizieren, dass sowohl Ein- als auch Ausgabe im Meldefenster angezeigt werden. Das dahingehend abgeänderte Programm [F3] sieht folgendermaßen aus:

```
« "Enter V, T and n: " {"↵ :V:↵ :T:↵ :n: " {2 0} V } INPUT
OBJ→ →V T n « V →STR "↵ " + T →STR "↵ " + n →STR "↵ " +
'(8.31451_J/(K*mol))*(n*T/V)' EVAL "p" →TAG →STR + + +
MSGBOX » »
```

Beachten Sie, dass Sie im Unterprogramm nach den Variablennamen V, T und n folgenden Code eingeben:

`→STR "↵ " +`

Um diesen Code das erste Mal einzugeben, verwenden Sie die Tastenfolge:

[←] PRG [TYPE] [→STR] [→] _" [→] ← [▶] [+]

Da die Funktionen des Menüs TYPE über die Funktionstasten weiter verfügbar sind, geben Sie im Unterprogramm für die zweite und dritte Code-Folge (`→STR "↵ " +`) (also nach den Variablen T und n) die nachstehende Tastenfolge ein:

[→STR] [→] _" [→] ← [▶] [+]

Sie werden feststellen, dass nach der Tastenfolge [→] ← im Stack eine neue Zeile erzeugt wird.

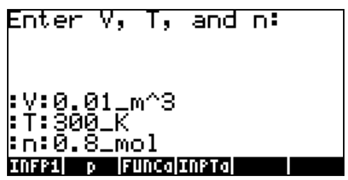
Die letzte Modifikation, die noch vorgenommen werden muss, ist das Eingeben von drei Pluszeichen nach dem Funktionsaufruf am Ende des Unterprogramms.

Hinweis: Das Pluszeichen (+) wird in diesem Programm zur *Verknüpfung* von Strings verwendet. Verknüpfung bedeutet einfach die Zusammenführung einzelner Strings.

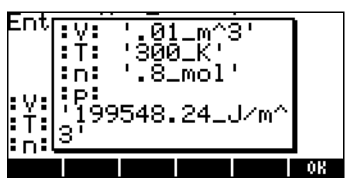
Um die Funktionsweise des Programms anzusehen,

- Speichern Sie das Programm mit der Tastenfolge [←] [p] in die Variable p.
- Starten Sie das Programm durch Drücken der Taste [p].
- Geben Sie, wenn Sie dazu aufgefordert werden, für V = 0,01_m^3, für T = 300_K und für n = 0,8_mol ein.

Wie in der älteren Version von [p] sieht der Stack vor Drücken der Taste [ENTER] für Ihre Eingabe folgendermaßen aus:



Bei der ersten Programmausgabe handelt es sich um ein Meldefenster mit folgendem String:



Drücken Sie [OK] zum Löschen des Meldefensters.

Integrieren von Einheiten in ein Programm

Wie Sie bei den verschiedenen Versionen des Programms [p] gesehen haben, kann das Anhängen von Einheiten an die Eingabewerte ziemlich mühsam sein. Sie können es so einrichten, dass das Programm selbst diese

Einheiten an die Ein- und Ausgabewerte anhängt. Veranschaulichen wir diese Option, indem wir unser Programm ein weiteres Mal wie folgt abändern.

Laden Sie den Inhalt des Programms mit der Tastenfolge wieder in den Stack und modifizieren es wie folgt:

Hinweis: Zur besseren Übersicht und zum leichterem Verständnis haben wir das Programm willkürlich in verschiedene Zeilen aufgeteilt. Das Programm muss nicht unbedingt so im Stack angezeigt werden. Die Befehlsfolge ist jedoch auf jeden Fall richtig. Beachten Sie auch, dass das Zeichen ↵ nicht im Stack angezeigt wird, sondern eine neue Zeile erzeugt.

```
« "Enter V,T,n [S.I.]: " {"↵ :V:↵ :T:↵ :n: " {2 0} V }
INPUT OBJ→ → V T n
« V '1_m^3' *  T '1_K' *  n '1_mol' *  → V T n
« V "V" →TAG →STR "↵ " + T "T" →TAG →STR "↵ " +  n
"n" →TAG →STR "↵ " +
'(8.31451_J/(K*mol))*(n*T/V)' EVAL "p" →TAG →STR + + +
MSGBOX » » »
```

Diese neue Programmversion enthält eine zusätzliche Unterprogrammebene (also eine dritte Ebene der Programmsymbole « » und einige Schritte, die Listen verwenden, beispielsweise

`V '1_m^3' * { } + T '1_K' * + n '1_mol' * + EVAL → V T n`

Dieser Programmcode wird wie folgt *interpretiert*: (Wir verwenden als Eingabewerte `:V:0.01`, `:T:300`, und `:n:0.8`):

1. `V` : Der Wert von V als gekennzeichnete Eingabe (z.B. V:0,01) wird in den Stack gestellt.

2. `'1_m^3'` : Die S.I.-Einheiten von V werden dann in Stack-Ebene 1 gestellt und die gekennzeichnete Eingabe für V in Stack-Ebene 2 verschoben.

3. `*` : Durch Multikplikation der Inhalte aus den Stack-Ebenen 1 und 2 erzeugen wir eine Zahl mit

Einheiten (z.B., 0.01_m^3), der Tag geht jedoch verloren.

4. `T '1_K' *` : Wert von T mit S.I.-Einheiten berechnen

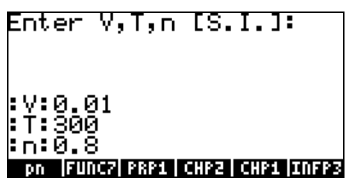
5. `n '1_mol' *` : Wert von n mit S.I.-Einheiten berechnen

6. `→V T n` : Die Werte von V, T und n in den Stack-Ebenen 3, 2 und 1 werden an die nächste Unterprogrammebene weitergegeben.

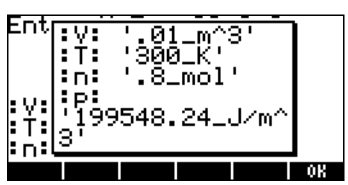
Um zu sehen, wie diese Programmversion arbeitet, führen Sie folgende Schritte durch:

- Speichern Sie das Programm mit der Tastenfolge [↰][p] in die Variable p.
- Starten Sie das Programm durch Drücken der Taste [p].
- Geben Sie, wenn Sie dazu aufgefordert werden, für V = 0,01, für T = 300 und für n = 0,8 ein (Einheiten sind nun nicht mehr erforderlich).

Vor Drücken der Taste ENTER sieht der Stack folgendermaßen aus:



Drücken Sie die Taste ENTER zum Ausführen des Programms. Bei der ersten Programmausgabe handelt es sich um ein Meldefenster mit folgendem String:

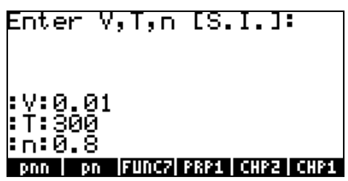


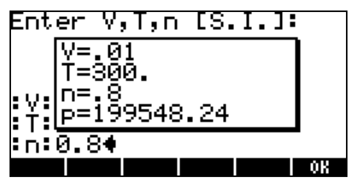
Zum Löschen des Meldefensters drücken Sie OK.

Ausgabe im Meldefenster ohne Einheiten
Ändern wir das Programm p ein weiteres Mal, um es ganz ohne Einheiten zu verwenden. Das Programm ohne Einheiten, sieht dann wie folgt aus:

```
« "Enter V,T,n [S.I.]: " {"↵ :V:↵ :T:↵ :n: " {2 0} V }
INPUT OBJ→ →V T n
« V DTAG  T DTAG  n DTAG →  V T n
« "V=" V →STR + "↵ "+ "T=" T →STR + "↵ "+ "n="  n →STR +
"↵ "+
'8.34451*n*T/V' EVAL →STR "p=" SWAP + + + + MSGBOX » » »
```

Wird das Programm mit den Werten V = 0,01, T = 300 und n = 0,08 gestartet, erscheint im Meldefenster folgende Ausgabe:





Zum Löschen des Meldefensters drücken Sie die Taste OK.

Relationale und logische Operatoren

Bis jetzt haben wir hauptsächlich mit sequentiellen Programmen gearbeitet. Die UserRPL-Sprache bietet Befehle, die im Programmablauf Verzweigungen und Schleifen ermöglichen. Viele davon treffen Entscheidungen basierend darauf, ob eine logische Aussage wahr oder unwahr ist. In diesem Absatz stellen wir relationale und logische Operatoren vor, mit deren Hilfe solche logischen Aussagen aufgebaut werden können.

Relationale Operatoren

Relationale Operatoren vergleichen die relative Position von zwei Objekten. Wenn Sie beispielsweise nur mit reellen Zahlen arbeiten, werden mit diesen

Operatoren die relativen Positionen von zwei oder mehreren reellen Zahlen ermittelt. Abhängig von den tatsächlich verwendeten Zahlen können solche Aussagen wahr (im Taschenrechner durch die Zahl 1 dargestellt) oder falsch (im Taschenrechner durch die Zahl 0 dargestellt) sein.

Zum Programmieren stehen folgende relationalen Operatoren zur Verfügung:

Operator	**Bedeutung**	**Beispiel**
==	„ist gleich"	'x==2'
≠	„ist nicht gleich"	'3 ≠ 2'
<	„ist kleiner als"	'm<n'
>	„ist größer als"	'10>a'
≥	„ist größer als oder gleich"	'p ≥ q'
≤	„ist kleiner als oder gleich"	'7≤12'

Alle Operatoren, mit Ausnahme von == (der durch die Tastenfolge [↱] = [↱] = erzeugt wird), stehen auf der Tastatur zur Verfügung. Sie finden sie auch in [↰] PRG TEST.

Zwei Zahlen, Variablen oder algebraische Ausdrücke verknüpft mit einem relationalen Operator bilden einen logischen Ausdruck, der wahr (1.), falsch (0.) oder nicht abgefragt sein kann. Um festzustellen, ob ein logischer Ausdruck wahr oder unwahr ist, stellen Sie ihn in die Stack-Ebene 1 und drücken ([EVAL]). Beispiele:

'2<10' [EVAL], Ergebnis: 1. (wahr)

'2<10' [EVAL], Ergebnis: 0. (unwahr)

In dem nächsten Beispiel gehen wir davon aus, dass die Variable m nicht initialisiert wurde (ihr wurde noch kein numerischer Wert zugewiesen);

'2==m' [EVAL], Ergebnis: '2==m'

Die Tatsache, dass das Ergebnis nach dem Vergleich dem Originalausdruck entspricht, bedeutet, dass keine eindeutige Aussage getroffen werden kann.

Logische Operatoren

Logische Operatoren sind Ausdrücke, die in einfache logische Aussagen integriert werden oder diese modifizieren können. Die verfügbaren logischen

Operatoren können über die nachstehende Tastenfolge aufgerufen werden: ← PRG TEST NXT .

Folgende Operatoren stehen zur Verfügung: AND, OR, XOR (exklusiv oder), NOT und SAME. Abhängig vom Wert der betroffenen logischen Aussage erzeugen die Operatoren Ergebnisse die entweder wahr oder unwahr sind. Der Operator NOT (Negation) gilt für einzelne logische Aussagen. Alle anderen gelten immer für zwei logische Aussagen.

Eine Auflistung aller möglichen Kombinationen für eine oder zwei Aussagen zusammen mit dem Ergebnis durch Anwendung eines bestimmten logischen Operators führt zu einer Wahrheitstabelle für den Operator. Nachfolgend finden Sie Wahrheitstabellen für alle im Taschenrechner verfügbaren logischen Operatoren:

p	NOT p
1	0
0	1

p	q	p AND q
1	1	1
1	0	0
0	1	0
0	0	0

p	q	p OR q
1	1	1
1	0	1
0	1	1
0	0	0

p	q	p XOR q

1	1	0
1	0	1
0	1	1
0	0	0

Auch der logische Operator SAME ist im Taschenrechner integriert. Hierbei handelt es sich nicht um einen logischen Standardoperator. Dieser ermittelt, ob zwei Objekte identisch sind. Ist dies der Fall, wird 1 (wahr) zurückgegeben, ist dies nicht der Fall, wird 0 (unwahr) zurückgegeben. Geben Sie beispielsweise folgende Übung im RPN-Modus ein, wird als Wert 0 zurückgegeben:

‘SQ(2)’ ENTER 4 ENTER SAME

Beachten Sie, dass bei der Verwendung von SAME der Begriff "identisch" sehr genau ausgelegt wird. Aus diesem Grund ist SQ(2) nicht identisch mit 4, obwohl der numerische Wert beider Ausdrücke 4 ergibt.

Programmverzweigung

Verzweigung eines Programmablaufes bedeutet, dass das Programm zwischen zwei oder mehreren Abläufen entscheidet. Die UserRPL-Sprache stellt zu diesem Zweck verschiedene Befehle zur Verfügung. Über nachfolgende Tastenfolge können Sie das Menü mit diesen Befehlen erreichen:

← PRG BRCH

In diesem Menü finden Sie Untermenüs für die Befehlsanweisungen.

Die Anweisungen IF…THEN..ELSE…END und CASE…THEN…END werden als Verzweigungsbefehle bezeichnet. Die weiteren Anweisungen wie START, FOR, DO und WHILE dienen zum Einfügen von Wiederholungen in ein Programm und werden als Schleifenbefehle bezeichnet. Die zuletzt genannten Anweisungen werden in einem späteren Absatz näher beschrieben.

Verzweigung mit IF

In diesem Absatz finden Sie Beispiele zu den Anweisungen IF…THEN…END und IF…THEN…ELSE…END.

Die Anweisung IF…THEN…END

IF…THEN…END ist die einfachste Form des IF-Befehls. Die allgemeine Syntax des Befehls lautet wie folgt:

IF *logical_statement* THEN *program_statements* END.

Diese Anweisung arbeitet wie folgt:

1. Ermitteln der logischen Aussage (logical_statement)
2. Ist die logische Aussage wahr, Ausführung der Programmschritte (program_statements) und Fortsetzung des Programmablaufs nach dem Befehl END.
3. Ist die logische Aussage unwahr, Überspringen der Programmschritte und Fortsetzung des Programmablaufs nach dem Befehl END.

Um IF, THEN, ELSE und END einzugeben, verwenden Sie:

← PRG BRCH IF

Die Funktionen IF THEN ELSE END stehen in diesem Menü zur Verfügung, müssen aber vom Anwender einzeln eingegeben werden. Alternativ dazu können Sie die IF…THEN…END Anweisung mit der nachstehenden Tastenfolge direkt in den Stack eingeben:

← PRG BRCH ← IF

Hierdurch wird im Stack folgende Eingabe abgelegt:

Der Cursor ⇐ befindet sich vor der IF-Anweisung und der Anwender muss nun seine logische Aussage eingeben, die beim Ausführen des Programms dann aktiviert wird.
<u>Beispiel</u>: Geben Sie das folgende Programm ein

```
« → x « IF 'x<3' THEN 'x^2' EVAL END "Done" MSGBOX » »
```

und speichern es unter dem Namen 'f1'. Drücken Sie VAR, um sich zu vergewissern, dass die Variable f1 tatsächlich in Ihrem Variablen-Menü zur Verfügung steht. Prüfen Sie die folgenden Ergebnisse:

0 f1	Ergebnis: 0	1.2 f1	Ergebnis: 1.44
3.5 f1	Ergebnis: nichts passiert	10 f1	Ergebnis: nichts passiert

Diese Ergebnisse bestätigen, dass die IF…THEN…END-Anweisung korrekt arbeitet. Das Programm berechnet die Funktion $f_1(x) = x^2$, wenn $x < 3$ (anderenfalls erfolgt keine Ausgabe).

Die Anweisung IF…THEN…ELSE…END

Die Befehlanweisung IF…THEN…ELSE…END bietet, abhängig vom Wahrheitswert der logischen Aussage, zwei verschiedene Programmverläufe. Die allgemeine Form der Anweisung sieht wie folgt aus:

```
IF logical_statement THEN program_statements_if_true
      ELSE program_statements_if_false END.
```

Diese Anweisung arbeitet wie folgt:

1. Ermitteln der logischen Aussage
2. Ist die logische Aussage wahr, Ausführung der Programmschritte (program_statements_if_true) und Fortsetzen des Programmablaufs nach dem Befehl END.
3. Ist die logische Aussage unwahr, Ausführung der Programmschritte (program_statements_if_false) und Fortsetzung des Programmablaufs nach dem Befehl END.

Alternativ können Sie die Anweisung IF…THEN…ELSE…END mit der nachstehenden Tastenfolge direkt in den Stack eingeben:

← PRG BRCH → IF

Hierdurch wird im Stack folgende Eingabe abgelegt:

Beispiel: Geben Sie das folgende Programm ein:

```
« → x « IF 'x<3' THEN 'x^2' ELSE '1-x' END EVAL "Done" MSGBOX
» »
```

und speichern es unter dem Namen 'f2'. Drücken Sie VAR, um sich zu vergewissern, dass die Variable f2 tatsächlich in Ihrem Variablen-Menü zur Verfügung steht. Prüfen Sie die folgenden Ergebnisse:

0 f2 Ergebnis: 0	1.2 f2 Ergebnis: 1.44
3.5 f2 Ergebnis: -2.5	10 f2 Ergebnis: -9

Diese Ergebnisse bestätigen, dass die IF…THEN…ELSE…END-Anweisung korrekt arbeitet. Das Programm berechnet folgende Funktion:

$$f_2(x) = \begin{cases} x^2, if\ x < 3 \\ 1 - x, otherwise \end{cases}$$

Hinweis: Für diesen speziellen Fall könnte als Alternative auch die Funktion IFTE in folgender Form verwendet werden: 'f2(x) = IFTE(x<3,x^2,1-x)'

Verschachtelte IF…THEN…ELSE…END Anweisungen

In den meisten Programmiersprachen, die über eine IF…THEN…ELSE…END Anweisung verfügen, sieht das allgemeine Format folgendermaßen aus:

`IF` *`logical_statement`* `THEN`

```
      program_statements_if_true
ELSE
      program_statements_if_false
END
```

Wenn Sie ein Programm mit einer IF-Anweisung für den Taschenrechner erstellen, können Sie zunächst den Code, wie oben dargestellt, von Hand notieren. Für Programm f2 könnten Sie beispielsweise wie folgt schreiben:

```
IF x<3 THEN
      x^2
ELSE
      1-x
END
```

Diese Anweisung funktioniert einwandfrei, wenn Ihre Funktion nur zwei Verzweigungen hat. Bei Funktionen mit drei oder mehr Verzweigungen müssen Sie die IF…THEN…ELSE…END Anweisungen ineinander verschachteln. Betrachten Sie beispielsweise folgende Funktion:

$$f_3(x) = \begin{cases} x^2, if\ x < 3 \\ 1 - x,\ \ if\ 3 \le x < 5 \\ \sin(x), if\ 5 \le x < 3\pi \\ \exp(x), if\ 3\pi \le x < 15 \\ -2, elsewhere \end{cases}$$

Dies wäre ein möglicher Weg, diese Funktion mit IF… THEN … ELSE … END Anweisung zu berechnen:

```
IF x<3 THEN
      x^2
ELSE
      IF x<5 THEN
            1-x
```

```
     ELSE
          IF x<3π          THEN
               sin(x)
          ELSE
               IF x<15 THEN
                    exp(x)
               ELSE
                    -2
               END
          END
     END
END
```

Eine komplexe IF-Anweisung wie diese wird <u>verschachtelte</u> IF … THEN … ELSE … END Anweisung genannt.

Eine Möglichkeit f3(x), mit einer verschachtelten IF-Anweisung zu untersuchen, wäre das folgende Programm:

```
« → x « IF ‘x<3‘ THEN ‘x^2‘ ELSE IF ‘x<5‘ THEN ‘1-x‘ ELSE IF
‘x<3*π‘ THEN ‘SIN(x)‘ ELSE IF ‘x<15‘ THEN ‘EXP(x)‘ ELSE –2
END END END END EVAL » »
```

Speichern Sie das Programm in die Variable f3 und versuchen Sie Folgendes:

1,5 f3	Ergebnis:	2,25 (d.h., x^2)
2,5 f3	Ergebnis:	6,25 (d.h., x^2)
4,2 f3	Ergebnis:	-3,2 (d.h., 1-x)
5,6 f3	Ergebnis:	-0,631266… (d.h., sin(x), mit x in Radian)
12 f3	Ergebnis:	162754,791419 (d.h., exp(x))
23 f3	Ergebnis:	-2. (d.h., -2)

Die Anweisung CASE

Die Anweisung CASE wird zum Programmieren von Verzweigungen mit verschachtelten IF-Anweisungen wie weiter oben gezeigt, verwendet. Die allgemeine Form der Anweisung sieht wie folgt aus:

```
CASE
Logical_statement₁ THEN program_statements₁ END
Logical_statement₂ THEN program_statements₂ END
.
.
.
Logical_statement THEN program_statements END
Default_program_statements (optional)
END
```

Bei der Auswertung dieser Anweisung testet das Programm jede einzelne logische Aussage (*logical_statements),* bis es eine findet, die wahr ist. Das Programm führt dann die entsprechenden Programmanweisungen (*program_statements)* aus und fährt mit dem Programm nach dem Befehl END fort.

Die Anweisungen CASE, THEN und END finden Sie mit der Tastenfolge ← PRG BRCH CASE .

Wenn Sie sich im Menü BRCH befinden, d.h., (← PRG BRCH), können Sie die folgenden Tastenkürzel verwenden, um Ihre CASE-Anweisung einzugeben (Die Position des Cursors wird durch das Symbol ← angezeigt):

- ← CASE: Startet die Case-Anweisung mit den Eingabeaufforderungen: CASE ← THEN END END
- → CASE: Beendet eine CASE-Zeile, indem es THEN ← END anfügt.

Beispiel – Programm $f_3(x)$ mit CASE-Anweisungen
Die Funktion wird durch die folgenden 5 Ausdrücke definiert:

$$f_3(x) = \begin{cases} x^2, if\ x < 3 \\ 1-x,\ if\ 3 \le x < 5 \\ \sin(x), if\ 5 \le x < 3\pi \\ \exp(x), if\ 3\pi \le x < 15 \\ -2, elsewhere \end{cases}$$

Mit CASE-Anweisungen in der UserRPL-Sprache können wir diese Funktion als folgt codieren:

```
« → x « CASE 'x<3' THEN 'x^2' END 'x<5' THEN '1-x' END
'x<3*π' THEN 'SIN(x)' END 'x<15' THEN 'EXP(x)' END –2 END
EVAL » »
```

Speichern Sie das Programm unter F3C. Versuchen Sie dann das folgende Beispiel:

1,5	F3C	Ergebnis:	2,25 (d.h., x^2)
2,5	F3C	Ergebnis:	6,25 (d.h., x^2)
4,2	F3C	Ergebnis:	-3,2 (d.h., 1-x)
5,6	F3C	Ergebnis:	-0,631266… (d.h., sin(x), mit x in Radian)
12	F3C	Ergebnis:	162754,791419 (d.h., exp(x))
23	F3C	Ergebnis:	-2. (d.h., -2)

Wie Sie sehen, erzeugt f3c genau die gleichen Ergebnisse wie f3. Der einzige Unterschied liegt in den Anweisungen für die Verzweigungen. Bei der Funktion $f_3(x)$, die 5 verschiedene Ausdrücke für die Definition verwendet, ist die CASE-Anweisung vielleicht einfacher zu codieren als eine Anzahl verschachtelter IF … THEN … ELSE … END Anweisungen.

Programmschleifen

Bei Programmschleifen handelt es sich um Anweisungen, über die das Programm eine bestimmte Anzahl von Aussagen wiederholt. Angenommen Sie möchten die Gesamtsumme der Quadrate der Integer-Zahlen von 0 bis n berechnen, d.h.

$$S = \sum_{k=0}^{n} k^2$$

Um diese Summe zu berechnen, müssen Sie im Gleichungs-Editor nur ‍→ Σ eingeben und dann die Grenzwerte und Ausdrücke für die Summenbildung laden (Beispiele hierzu finden Sie in den Kapiteln 2 und 13). Um den Einsatz von Programmschleifen zu veranschaulichen, werden wir diese Summenbildung mit unserem eigenen UserRPL-Code berechnen. UserRPL bietet vier verschiedene Anweisungen zur Programmierung von Programmschleifen, START, FOR, DO und WHILE. Die Anweisungen START und FOR verwenden einen Index oder Zähler, um festzulegen, wie oft die Schleife ausgeführt werden soll. Die Anweisungen DO und WHILE verwenden eine logische Aussage, um zu entscheiden, wann eine Schleife verlassen wird. Eine genauere Erläuterung zum Verwenden der Schleifenbefehle finden Sie im nächsten Absatz.

Die Anweisung START

START verwendet zwei Werte eines Index, um eine Anzahl an Anweisungen zu wiederholen. Es gibt zwei verschiedene START-Anweisungen: START…NEXT und START … STEP. START…NEXT wird verwendet, wenn der Index jeweils um 1 erhöht wird. START…STEP wird eingesetzt, wenn das Inkrement des Index vom Anwender festgelegt wird.

Die Befehle der START-Anweisung rufen Sie wie folgt auf:

← PRG BRCH START START

Innerhalb des Menüs BRCH (← PRG BRCH) stehen folgende Tastenfolgen zum Erstellen einer START-Anweisung zur Verfügung (das Symbol zeigt die Cursor-Position):

- ← START : Startet die START…NEXT Anweisung: START ◆ NEXT
- → START : Startet die START…STEP Anweisung: START ◆ STEP

Die Anweisung START…NEXT

Die allgemeine Form der Anweisung sieht wie folgt aus:

```
start_value end_value START program_statements NEXT
```

Da in diesem Fall das Inkrement 1 ist, sollten Sie sicherstellen, dass `start_value < end_value` damit die Schleife beendet werden kann. Anderenfalls kommt es zu einer sogenannten Endlosschleife.

Beispiel – Berechnen der oben definierten Summenbildung S

Die START…NEXT-Anweisung hat einen Index, auf dessen Wert der Anwender nicht zugreifen kann. Da für die Kalkulation aber der Index selbst erforderlich ist (in diesem Fall k), erstellen wir einen eigenen Index k, der innerhalb der Schleife, jedes Mal wenn das Programm die Schleife durchläuft, erhöht wird. Mit dem nachfolgenden Programm ließe sich eine mögliche S berechnen:

```
« 0. DUP →n S k « 0. n START k SQ S + 1. ‘k‘ STO+ ‘S‘ STO
NEXT S “S” TAG » »
```

Geben Sie das Programm in Ihrer Taschenrechner ein und speichern Sie dieses unter S3.

Nachfolgend eine kurze Erläuterung zur Arbeitsweise des Programms:

1. Das Programm benötigt als Eingabe eine Integer-Zahl. Vor Ausführen des Programms befindet sich diese Zahl (n) in Stack-Ebene 1. Das Programm wird gestartet.
2. Eine Null wird eingegeben und n in die Stack-Ebene 2 geschoben.
3. Der Befehl DUP, der über die Tastenfolge ALPHA ALPHA D U P ALPHA eingegeben werden kann, kopiert den Inhalt der Stack-Ebene 1, verschiebt alle Stack-Ebenen nach oben und schreibt die gerade erstellte Kopie in Stack-Ebene 1. Nachdem DUP ausgeführt wurde, befindet sich n in Stack-Ebene 3 und die Ebenen 1 und 2 enthalten Nullen.

4. Der Programmcode → `n S k` speichert die Werte von n, 0 und 0 in die lokalen Variablen n, S, k. Man sagt, dass die Variablen n, S und k <u>initialisiert</u> wurden (S und k auf 0 und n auf den Wert, den der Anwender ausgewählt hat).
5. Der Programmcode `0. n START` beschreibt eine START-Schleife deren Index die Werte 0, 1, 2, …, n annehmen wird.
6. Die Summe S wird durch den folgenden Programmcode um k^2 erhöht: `k SQ S +`
7. Der Index k wird durch den folgenden Programmcode um 1 erhöht: `1. k +`
8. An dieser Stelle stehen die aktuellen Werte für S und k in den Stack-Ebenen 2 und 1. Der Programmcode 'k' STO speichert den Wert der Stack-Ebene 1 in die lokale Variable k. Danach steht der aktuelle Wert von S in Stack-Ebene 1.
9. Der Programmcode 'S' STO speichert den Wert der Stack-Ebene 1 in die lokale Variable k. Danach ist der Stack leer.
10. Der Programmteil NEXT erhöht den Index um 1 und setzt das Programm an den Beginn der Schleife (Schritt 6).
11. Die Schleife wird wiederholt, bis der Index den maximalen Wert n erreicht hat.
12. Der letzte Teil des Programms ruft den letzten Wert von S (die Summenbildung) auf, kennzeichnet diesen und schreibt ihn in Stack-Ebene 1, wo dieser dann als Ausgabe angezeigt wird.

Um das Programm Schritt für Schritt zu betrachten, können Sie den Debugger wie folgt verwenden (n = 2). SL1 bedeutet Stack-Ebene 1:

VAR 2 ['] S1 ENTER	Schreiben Sie eine 2 in Ebene 2 und den Programmnamen ‚S1' in Ebene 1
← PRG NXT NXT RUN DBG	Starten der Fehlersuche (des Debuggers) SL1 = 2.
SST↓	SL1 = 0., SL2 = 2.
SST↓	SL1 = 0., SL2 = 0., SL3 = 2. (DUP)
SST↓	Leeren des Stacks (-> n S k)
SST↓	Leeren des Stacks (« - Starten des Unterprogramms)

SST↓	SL1 = 0., (Startwert des Schleifenindex)
SST↓	SL1 = 2.(n), SL2 = 0 (Endwert des Schleifenindex)
SST↓	Leeren des Stacks (START – Beginn der Schleife)

--- Erster Durchlauf der Schleife für k = 0

SST↓	SL1 = 0. (k)
SST↓	SL1 = 0. (SQ(k) = k^2)
SST↓	SL1 = 0.(S), SL2 = 0. (k^2)
SST↓	SL1 = 0. (S + k^2)
SST↓	SL1 = 1., SL2 = 0. (S + k^2)
SST↓	SL1 = 0.(k), SL2 = 1., SL3 = 0. (S + k^2)
SST↓	SL1 = 1.(k+1), SL2 = 0. (S + k^2)
SST↓	SL1 = ‘k’, SL2 = 1., SL3 = 0. (S + k^2)
SST↓	SL1 = 0. (S + k^2) [Speichert den Wert von SL2 = 1, in SL1 = ‚k’]
SST↓	SL1 = ‘S’, SL2 = 0. (S + k^2)
SST↓	Leeren des Stacks [Speichert Wert von SL2 = 0, in SL1 = ‚S’]
SST↓	Leeren des Stacks (NEXT – Ende der Schleife)

--- Zweiter Durchlauf der Schleife für k = 1

SST↓	SL1 = 1. (k)
SST↓	SL1 = 1. (SQ(k) = k^2)
SST↓	SL1 = 0.(S), SL2 = 1. (k^2)
SST↓	SL1 = 1. (S + k^2)
SST↓	SL1 = 1., SL2 = 1. (S + k^2)
SST↓	SL1 = 1.(k), SL2 = 1., SL3 = 1. (S + k^2)
SST↓	SL1 = 2.(k+1), SL2 = 1. (S + k^2)
SST↓	SL1 = ‘k’, SL2 = 2., SL3 = 1. (S + k^2)
SST↓	SL1 = 1. (S + k^2) [Speichert den Wert von SL2 = 2, in SL1 = ‚k’]
SST↓	SL1 = ‘S’, SL2 = 1. (S + k^2)
SST↓	Leeren des Stacks [Speichert den Wert von

	SL2 = 1, in SL1 = ‚S’]
SST↓	Leeren des Stacks (NEXT – Ende der Schleife)

-- Dritter Durchlauf der Schleife für k = 2

SST↓	SL1 = 2. (k)
SST↓	SL1 = 4. (SQ(k) = k^2)
SST↓	SL1 = 1.(S), SL2 = 4. (k^2)
SST↓	SL1 = 5. (S + k^2)
SST↓	SL1 = 1., SL2 = 5. (S + k^2)
SST↓	SL1 = 2.(k), SL2 = 1., SL3 = 5. (S + k^2)
SST↓	SL1 = 3.(k+1), SL2 = 5. (S + k^2)
SST↓	SL1 = ‘k’, SL2 = 3., SL3 = 5. (S + k^2)
SST↓	SL1 = 5. (S + k^2) [Speichert den Wert von SL2 = 3, in SL1 = ‚k’]
SST↓	SL1 = ‘S’, SL2 = 5. (S + k^2)
SST↓	Leeren des Stacks [Speichert den Wert von SL2 = 0, in SL1 = ‚S’]
SST↓	Leeren des Stacks (NEXT – Ende der Schleife)

-- bei n = 2 ist der Schleifenindex verbraucht und das Programm fährt mit den Anweisungen nach NEXT fort

SST↓	SL1 = 5 (S wird in den Stack geladen)
SST↓	SL1 = “S”, SL2 = 5 (“S” wird in den Stack geschrieben)
SST↓	SL1 = S:5 (markieren des Ausgabewerts)
SST↓	SL1 = S:5 (Unterprogramm verlassen ≫)
SST↓	SL1 = S:5 (Hauptprogramm verlassen ≫)

Die Schritt-für-Schritt-Auflistung ist hier beendet. Das Ergebnis des Programms S1 mit n = 2 beträgt S:5.

Prüfen Sie auch die folgenden Ergebnisse: VAR

3	S1	Ergebnis: `S:14`	4	S1	Ergebnis: `S:30`

5 S1	Ergebnis: `S:55`	8 S1	Ergebnis: `S:204`
10 S1	Ergebnis: `S:385`	20 S1	Ergebnis: `S:2870`
30 S1	Ergebnis: `S:9455`	100 S1	Ergebnis: `S:338350`

Die Anweisung START…STEP

Die allgemeine Form der Anweisung lautet wie folgt:

```
start_value end_value START program_statements increment
NEXT
```

start_value (Startwert), end_value (Endwert) und `increment` (Inkrement) des Schleifenindex können positive oder negative Werte haben. Bei einem `increment > 0` wird die Schleife so lange ausgeführt, wie der Index kleiner oder gleich `end_value` ist. Bei einem `increment < 0` wird die Schleife so lange ausgeführt, wie der Index größer oder gleich `end_value` ist.

Beispiel – Erstellen einer Werteliste

Nehmen Sie an, dass Sie eine Werteliste für x von x = 0.5 bis x = 6.5 in Schritten von 0.5 erstellen wollen. Dafür können Sie das folgende Programm verwenden:

```
« → xs xe dx « xs DUP xe START DUP dx + dx STEP DROP xe
xs – dx / ABS 1 + →LIST » »
```

und es in die Variable GLIST speichern.

In diesem Programm sind xs = Startwert der Schleife, xe = Endwert der Schleife und dx = Inkrement der Schleife. Das Programm schreibt Werte für xs, xs+dx, xs+2·dx, xs+3·dx, … in den Stack. Dann errechnet es mit dem folgenden Programmcode die Anzahl der erstellten Elemente:

```
xe xs – dx / ABS 1. +
```

Am Ende erstellt das Programm eine Liste mit den Elementen aus dem Stack.

- Prüfen Sie, ob der Programmaufruf 0.5 ENTER 2.5 ENTER 0.5 ENTER GLIST folgende Liste {0,5 1. 1,5 2. 2,5} erstellt.

- Um die Schritt-für-Schritt-Ausführung anzusehen, verwenden Sie das Programm DBUG für eine kurze Liste, beispielsweise:

VAR 1 SPC 1,5 SPC 0,5 ENTER	Parameter 1 1,5 0,5 eingeben
['] DBUG ENTER	Programmnamen in Ebene 1
← PRG NXT NXT RUN DBG	Start der Fehlersuche (des Debuggers)

Verwenden Sie SST↓ , um in das Programm zu springen und zu beobachten, wie die einzelnen Anweisungen abgearbeitet werden.

Die Anweisung FOR

Wie bei START-Anweisung, gibt es auch bei FOR zwei Varianten: Die FOR…NEXT-Anweisung, bei einer Erhöhung des Index um 1 und die FOR…STEP-Anweisung mit einer Erhöhung des Index, um eine Zahl, die vom Anwender selbst auszuwählen ist. Anders als bei der START-Anweisung müssen wir bei der Anweisung FOR für den Schleifenindex einen Namen vergeben (z.B. j, k, n). Wir müssen uns aber nicht wie bei START um die Erhöhung des Index selbst kümmern. Der entsprechende Wert des Index kann für weitere Berechnungen verwendet werden.

Die Befehle der FOR-Anweisung rufen Sie wie folgt auf:

← PRG BRCH FOR

Innerhalb des Menüs BRCH (← PRG BRCH) stehen folgende Tastenfolgen zur Verfügung, um eine FOR-Anweisung zu erstellen (das Symbol ⇦ zeigt die Cursor-Position):

- ← FOR: Startet die FOR…NEXT- Anweisung: FOR ⇦ NEXT

- → FOR: Startet die FOR…STEP- Anweisung: FOR ⇦ STEP

Die Anweisung FOR…NEXT

Die allgemeine Form der Anweisung sieht wie folgt aus:

```
start_value end_value FOR loop_index program_statements
NEXT
```

Um eine Endlosschleife zu verhindern, stellen Sie sicher, dass `start_value < end_value` ist.

Beispiel – Berechnung der Summenbildung S mit einer FOR…NEXT Anweisung
Das folgende Programm berechnet die Summenbildung

$$S = \sum_{k=0}^{n} k^2$$

Verwenden Sie beispielsweise eine FOR…NEXT-Schleife:

```
« 0 →n S « 0 n FOR k k SQ S + 'S' STO NEXT S "S" TAG » »
```

Speichern Sie dieses neue Programm in die Variable . Versuchen Sie folgendes Beispiel: VAR

3	Ergebnis: `S:14`	4	Ergebnis: `S:30`
5	Ergebnis: `S:55`	8	Ergebnis: `S:204`
10	Ergebnis: `S:385`	20	Ergebnis: `S:2870`
30	Ergebnis: `S:9455`	100	Ergebnis: `S:338350`

Sicher werden Sie schon bemerkt haben, dass dieses Programm sehr viel einfacher ist als das unter gespeicherte. Der Index k muss innerhalb des Programms weder initialisiert noch inkrementiert werden. Das Programm selbst übernimmt diese Aufgaben.

Die Anweisung FOR…STEP

Die allgemeine Form der Anweisung sieht wie folgt aus:

```
start_value end_value FOR loop_index program_statements
increment STEP
```

start_value (Startwert), end_value (Endwert) und `increment` (Inkrement) des Schleifenindex können positive oder negative Werte sein. Bei einem `increment > 0` wird die Schleife so lange ausgeführt, wie der Index kleiner oder gleich `end_value` ist. Bei einem `increment < 0` wird die Schleife so lange ausgeführt, wie der Index größer oder gleich `end_value` ist. Die Programmanweisungen werden mindestens ein Mal ausgeführt (z.B. `1 0 START 1 1 STEP` gibt 1 zurück).

Beispiel – Erstellen einer Werteliste mit einer FOR…STEP-Anweisung
Geben Sie das folgende Programm ein

```
« → xs xe dx « xe xs – dx / ABS 1. + → n « xs xe FOR x
x dx STEP n →LIST » » »
```

und speichern Sie dies in die Variable GLIST .

- Prüfen Sie, ob der Programmaufruf 0,5 ENTER 2,5 ENTER 0,5 ENTER GLIST folgende Liste {0,5 1. 1,5 2. 2,5} erstellt.
- Um die Ausführung Schritt für Schritt anzusehen, verwenden Sie das Programm DBUG für eine kurze Liste, beispielsweise:

VAR 1 SPC 1.5 SPC 0.5 ENTER	Parameter 1 1.5 0.5 eingeben
['] GLIST ENTER	Programmnamen in Ebene 1
← PRG NXT NXT RUN DBG	Start der Fehlersuche (de Debuggers)

Verwenden Sie SST↓ , um in das Programm zu springen und zu beobachten, wie die einzelnen Anweisungen abgearbeitet werden.

Die Anweisung DO

Die allgemeine Form der Anweisung sieht wie folgt aus:

```
DO program_statements UNTIL logical_statement END
```

Die Anweisung DO startet eine unendliche Schleife und führt die program_statements aus, bis logical_statement (logische Aussage) FALSE (0) zurückgibt. `logical_statement` muss den Wert eines Index enthalten, dessen Wert durch `program_statements` (Programm-Aussage) verändert wird.

Beispiel 1 – Dieses Programm erzeugt in der oberen linken Ecke der Anzeige einen Zähler, der in einer Endlosschleife 1 aufaddiert, bis durch Drücken auf eine Taste der Zähler gestoppt wird: `« 0 DO DUP 1 DISP 1 + UNTIL KEY END DROP »`

Der Befehl KEY erkennt TRUE, sobald eine Taste gedrückt wird.

Beispiel 2 – Berechnen der Summenbildung S mit einer DO…UNTIL…END-Anweisung
Das folgende Programm berechnet die Summenbildung

$$S = \sum_{k=0}^{n} k^2$$

Verwenden Sie eine DO…UNTIL…END-Schleife:

```
« 0. →n S « DO n SQ S + ‘S‘ STO n 1 – ‘n‘ STO UNTIL
‘n<0‘ END S “S” TAG » »
```

Speichern Sie das neue Programm in die Variable [illegible]. Versuchen Sie folgendes Beispiel: VAR

3 [illegible]	Ergebnis: `S:14`	4 [illegible]	Ergebnis: `S:30`
5 [illegible]	Ergebnis: `S:55`	8 [illegible]	Ergebnis: `S:204`
10 [illegible]	Ergebnis: `S:385`	20 [illegible]	Ergebnis: `S:2870`
30 [illegible]	Ergebnis: `S:9455`	100 [illegible]	Ergebnis: `S:338350`

Beispiel 3 – Erzeugen einer Liste mit einer DO…UNTIL…END-Anweisung
Geben Sie das folgende Programm ein

```
« → xs xe dx « xe xs – dx / ABS 1. + xs → n x « xs DO
‘x+dx’ EVAL DUP ‘x’ STO UNTIL ‘x≥xe’ END n →LIST » » »
```

und speichern es in die Variable [GLIS2] .

- Prüfen Sie, ob der Programmaufruf 0,5 [ENTER] 2,5 [ENTER] 0,5 [ENTER] [GLIS2] folgende Liste {0,5 1. 1,5 2. 2,5} erstellt.
- Um die Ausführung Schritt für Schritt zu betrachten, verwenden Sie das Programm DBUG für eine kurze Liste, beispielsweise:

[VAR] 1 [SPC] 1,5 [SPC] 0,5 [ENTER]	Parameter 1 1,5 0,5 eingeben
[‘] [GLIS2] [ENTER]	Programmnamen in Ebene 1
[←] PRG [NXT] [NXT] [RUN] [DBG]	Start der Fehlersuche (des Debuggers)

Verwenden Sie [SST↓] , um in das Programm zu springen und zu beobachten, wie die einzelnen Anweisungen abgearbeitet werden.

Die Anweisung WHILE

Die allgemeine Form de Anweisung sieht wie folgt aus:

```
WHILE logical_statement REPEAT program_statements END
```

Der Befehl WHILE wiederholt `program_statements` so lange, wie `logical_statement` wahr ist (nicht Null). Ist dies nicht mehr der Fall, fährt das Programm mit den Befehlen direkt nach END fort. `program_statements` müssen einen Index enthalten, der geändert wird, bevor `logical_statement` zu Beginn der nächsten Wiederholung überprüft wird. Anders als beim Befehl DO wird die Schleife, wenn die Überprüfung von logical_statement beim ersten Mal unwahr zurück gibt, nie ausgeführt.

Beispiel 1 – Berechnung der Summenbildung S mit einer WHILE…REPEAT…END-Anweisung
Das folgende Programm berechnet die Summenbildung

$$S = \sum_{k=0}^{n} k^2$$

Verwenden einer WHILE…REPEAT…END-Schleife:

```
« 0. →n S « WHILE 'n≥0' REPEAT n SQ S + 'S' STO n 1 –
'n' STO END S "S" TAG » »
```

Speichern Sie das neue Programm in die Variable S4. Versuchen Sie folgendes Beispiel: VAR

3	S4	Ergebnis: S:14	4	S4	Ergebnis: S:30
5	S4	Ergebnis: S:55	8	S4	Ergebnis: S:204
10	S4	Ergebnis: S:385	20	S4	Ergebnis: S:2870
30	S4	Ergebnis: S:9455	100	S4	Ergebnis: S:338350

<u>Beispiel 2</u> – Erzeugen einer Liste mit einer WHILE…REPEAT…END-Anweisung Geben Sie das folgende Programm ein:

```
« → xs xe dx « xe xs – dx / ABS 1. + xs → n x « xs
WHILE 'x<xe' REPEAT 'x+dx' EVAL DUP 'x' STO END n →LIST » »
»
```

und speichern es in die Variable GLIS4 .

- Prüfen Sie, ob der Programmaufruf 0,5 ENTER 2,5 ENTER 0,5 ENTER GLIS4 folgende Liste {0,5 1. 1,5 2. 2,5} erstellt.
- Um die Ausführung Schritt für Schritt zu betrachten, verwenden Sie das Programm DBUG für eine kurze Liste, beispielsweise:

VAR 1 SPC 1,5 SPC 0,5 ENTER	Parameter 1 1,5 0,5 eingeben
['] GLIS4 ENTER	Programmnamen in Ebene 1
← PRG NXT NXT RUN DBG	Start der Fehlersuche (des Debuggers)

Verwenden Sie SST↓, um in das Programm zu springen und zu beobachten, wie die einzelnen Anweisungen abgearbeitet werden.

Fehler und Fehler auffangen

Mithilfe der Funktionen des Untermenüs PRG/ERROR können Sie Fehler im Taschenrechner und Fehler in Programmen auffangen. Das Menü PRG/ERROR kann mit der Tastenfolge ← PRG NXT NXT ERROR aufgerufen werden. Es enthält die folgenden Funktionen und Untermenüs:

DOERR

Diese Funktion simuliert einen benutzerdefinierten Fehler, wobei sich der Taschenrechner so verhält, als wäre dieser Fehler tatsächlich aufgetreten. Der Funktion kann als Argument eine Integer-Zahl, eine binäre Integer-Zahl, eine Fehlermeldung oder eine Null (0) zugeordnet werden. Geben Sie beispielsweise im RPN-Modus 5 ENTER DOERR ein, erscheint die folgende Fehlermeldung: *Error: Memory Clear (Fehler: Speicher leer).*

Wenn Sie #11h ENTER DOERR eingeben, erscheint die folgende Meldung: *Error: Undefined FPTR Name (Fehler: nicht definierter FPTR Name)*

Wenn Sie "TRY AGAIN" ENTER DOERR eingeben, bekommen Sie die Meldung: *TRY AGAIN (VERSUCHEN SIE ES ERNEUT)*

Schließlich erzeugt 0 ENTER DOERR die folgende Meldung: *Interrupted (unterbrochen)*

ERRN

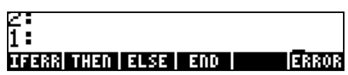
Diese Funktion gibt die Kennziffer des letzten Fehlers zurück. Wenn Sie beispielsweise 0 1/x ON ERRN versuchen, erscheint die Nummer #305h. Diese binäre Integer-Zahl steht für den Fehler *Infinite Result (unendliches Ergebnis)*

ERRM
Diese Funktion gibt eine Zeichenkette mit der Meldung des letzten Fehlers zurück. Z.B. Approx im Modus, wenn Sie versuchen 0 1/x ON ERRM, Sie erhalten die folgende Zeichenkette: *"Infinite Result" (unendliches Ergebnis)*

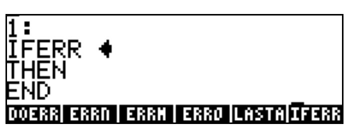
ERR0
Mit dieser Funktion wird die letzte Fehlernummer gelöscht, sodass die Funktion ERRN, Approx im Modus, #0h zurückgibt, wenn Sie beispielsweise 0 1/x ON ERR0 ERRN versuchen, erhalten Sie # 0h. Versuchen Sie hingegen 0 1/x ON ERR0 ERRM einzugeben, erhalten Sie eine leere Zeichenkette " ".

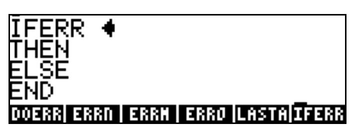
LASTARG
Diese Funktion gibt Kopien der Argumente des zuletzt ausgeführten Befehls oder der zuletzt ausgeführten Funktion zurück. Wenn Sie im RPN-Modus 3 ÷ 2 ENTER eingeben und dann LASTARG (LASTA) drücken, werden im Stack die Werte 3 und 2 aufgelistet. Geben Sie im RPN-Modus 5 TAN ENTER ein, gibt LASTARG eine 5 aus.

Untermenü IFERR
Das Untermenü IFERR bietet die folgenden Funktionen:



Dies sind die Anweisungen der IFERR … THEN … END oder IFERR … THEN … ELSE … END-Anweisungen. Mit beiden logischen Anweisungen können Fehler, die bei der Programmausführung auftreten, abgefangen werden. Wenn Sie im Untermenü ERROR ← IFERR oder → IFERR eingeben, werden die IFERR-Befehle in den Stack geschrieben und können dort vervollständigt werden:





Die allgemeine Form der beiden Anweisungen sieht wie folgt aus:

IF Abfangbedingung THEN Fehlerbedingung END

IF Abfangbedingung THEN Fehlerbedingung ELSE Normalbedingung END

Die Arbeitsweise dieser logischen Anweisungen entspricht denen, die Sie bei IF … THEN … END und IF … THEN … ELSE … END bereits kennen gelernt haben. Wird während der Ausführung der Abfangbedingungen ein Fehler entdeckt, wird die Fehlerbedingung ausgeführt. In allen anderen Fällen werden die Normalbedingungen ausgeführt.

Betrachten Sie beispielsweise das folgende Programm (ERR), das als Eingabe zwei Matrizen A und b verwendet und dabei die Abfangbedingung auf Fehler prüft: A b / (RPN-Modus, also A/b). Tritt ein Fehler auf, ruft das Programm die Funktion LSQ (Least SQuares, siehe Kapitel 11) auf, um das Gleichungssystem zu lösen:

```
« → A b « IFERR A b / THEN LSQ END » »
```

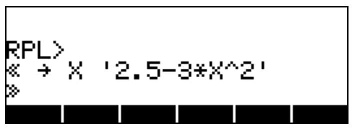
Testen Sie das Programm mit den Argumenten A = [[2, 3, 5] , [1, 2, 1]] und b = [[5] , [6]]. Eine einfache Division dieser Argumente verursacht einen Fehler: */Error: Invalid Dimension.*

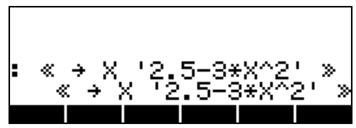
Mit der Fehlerabfrage des Programms ERR jedoch ergibt sich mit den gleichen Argumenten: [0,262295…, 0,442622…].

Programmieren in UserRPL im algebraischen Modus

Obwohl alle Programme, die wir bisher vorgestellt haben, im RPN-Modus erstellt und auch ausgeführt wurden, können Sie, wenn Sie sich im algebraischen Modus befinden, Programme auch mithilfe der Funktion RPL> in UserRPL eingeben. Die Funktion wird über den Befehlskatalog aufgerufen. Als Beispiel, versuchen Sie dasselbe Programm im ALG-Modus und speichern Sie dies in der Variablen P2: `« → X '2.5-3*X^2' »`

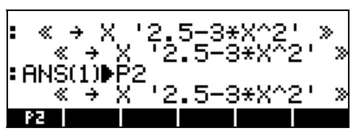
Starten Sie zuerst die RPL> Funktion aus dem Befehlskatalog (→ CAT). Allen im ALG-Modus gestarteten Funktionen werden mit einem Klammerpaar an deren Namen angehängt, dargestellt. Die Funktion RPL> ist hier keine Ausnahme. Bevor Sie aber ein Programm in den Display eingeben, müssen Sie diese Klammern entfernen. Zum Löschen der Klammern aus der RPL>() Anweisung verwenden Sie die Pfeiltasten (◀ ▶) sowie die Löschtaste (⬅). An dieser Stelle können Sie das RPL-Programm eingeben. Nachfolgende Abbildung zeigt den RPL> Befehl zusammen mit dem Programm vor und nach Drücken der Taste ENTER.





Zum Speichern des Programms verwenden Sie den Befehl STO.

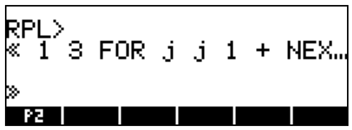
← ANS STO▶ ALPHA P 2 ENTER

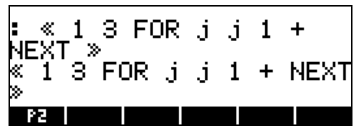


Eine Auswertung des Programms P2 für das Argument X = 5 wird in der nachfolgenden Abbildung dargestellt:

Während Sie im ALG-Modus das Programm ohne die RPL> Funktion erstellen, können einige RPL-Anweisungen beim Drücken der Taste ENTER eine Fehlermeldung erzeugen , so z.B.:

Verwenden Sie dagegen RPL, gibt es beim Laden des Programms im ALG-Modus keine Probleme:





Kapitel 22
Programme zum Manipulieren von Grafiken

Dieses Kapitel enthält einige Beispiele, mit denen die Funktionen des Taschenrechners für interaktive bzw. programmgesteuerte Manipulierung von Grafiken erläutert werden. Wie in Kapitel 21, empfiehlt sich auch hier die Verwendung des RPN-Modus, sowie das Setzen des Systemflags 117 auf SOFT-Menüeinträge.

In Kapitel 12 wurden mehrere Grafikanwendungen des Taschenrechners vorgeführt. Die Beispiele in Kapitel 12 haben die interaktive Erzeugung von Grafiken mithilfe der vorprogrammierten Eingabemasken des Taschenrechners behandelt. Grafiken können auch in Ihren Programmen eingesetzt werden, z.B. für die Erläuterung der numerischen Ergebnisse. Für die Lösung solcher Aufgaben werden zuerst die Funktionen im Menü PLOT beschrieben.

Das Menü PLOT

Befehle zum Einstellen und Erstellen von Plots finden Sie im Menü PLOT. Ins Menü PLOT gelangen Sie mit nachfolgender Tastenfolge: 8 1 . 0 / ← PRG NXT MODES MENU MENU.

Das so erzeugte Menü erlaubt dem Anwender Zugang zu einer Vielzahl von Grafik-Funktionen. Definieren wir die Taste F3 (GRAPH) für einen einfachen Zugang zum beschriebenen Menü, für die nachfolgenden Beispiele:

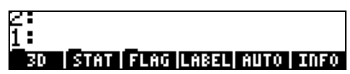
Benutzerdefinierte Taste für das Menü PLOT

Verwenden Sie die nachfolgende Tastenfolge, um zu überprüfen, ob auf dem Taschenrechner bereits benutzerdefinierte Tasten gespeichert wurden. ← PRG NXT MODES KEYS RCLK.

Falls keine benutzerdefinierten Tasten gespeichert sind, wird eine Liste mit einem S zurückgegeben (z.B. {S}). Dies bedeutet, dass auf Ihrem Taschenrechner nur die standardmäßige Tastendefinition gespeichert ist.
Um eine Taste als benutzerdefinierte Taste anzugeben, müssen Sie zu dieser Liste eine Anweisung bzw. ein Programm, gefolgt von der Angabe der Taste hinzufügen (mehr dazu finden Sie in Kapitel 20). Geben Sie die nachfolgende Liste `{ S << 81.01 MENU >> 13.0 }` in den Stack ein und verwenden Sie die Funktion STOREKEYS (← PRG NXT MODES KEYS STOK), um die Taste F3 als Zugriffstaste für das Menü PLOT zu definieren. Mit ← PRG NXT MODES KEYS RCLK können Sie überprüfen, ob die Liste im Taschenrechner gespeichert wurde.

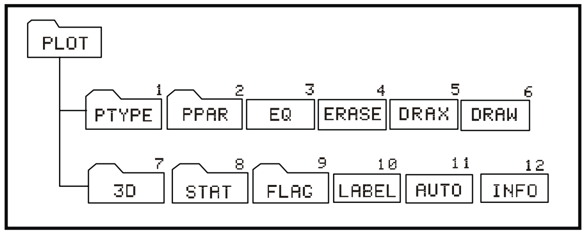
Hinweis: Bei der Beschreibung des Menüs PLOT, sowie seiner Funktionen oder Untermenüs, werden keine Beispiele aufgeführt. In diesem Abschnitt werden wir uns mehr auf die Inhalte des Menüs PLOT in Bezug auf verschiedene Grafikarten konzentrieren.

Um eine benutzerdefinierte Taste zu aktivieren, müssen Sie vor Drücken der gewünschten Taste oder Tastenfolge, erst ← USER drücken (genau wie bei der Taste ALPHA). Um das Menü PLOT mit der oben angegebenen Tastendefinition zu starten, drücken Sie ← USER F3. Das folgende Menü erscheint (drücken Sie NXT, um zum zweiten Menü zu gelangen).



Beschreibung des Menüs PLOT

Die folgende Abbildung zeigt die Einträge des Menüs PLOT. Die Zahlen oberhalb der Menüs und Funktionen dienen weiterhin als Referenz in der nachfolgenden Beschreibung dieser Objekte.



Die Funktionstasten 3D, STAT, FLAG, PTYPE und PPAR haben auch eigene Untermenüs, die zu einem späteren Zeitpunkt ausführlicher beschrieben werden. An dieser Stelle werden nur die Funktionen beschrieben, die im Menü Nummer 81.02 direkt mit den Funktionstasten aufgerufen werden können. Diese Funktionen lauten:

<u>LABEL (10)</u>
Mit der Funktion LABEL werden die Achsen eines Plots beschriftet, einschließlich der Variablennamen bzw. der unterer Grenzwerte und oberer Grenzwerte der Achsen. Die Namen der Variablen werden aus den Informationen in der Variable PPAR zusammengestellt.

<u>AUTO (11)</u>
Die Funktion AUTO (AUTO Maßstab) berechnet den Anzeigebereich für die y-Achse oder für beide Achsen (x und y) in zweidimensionalen Plots, je nach Plottyp, der in PPAR definiert wurde. Bei dreidimensionalen Grafiken hat die Funktion AUTO keine Auswirkungen. Bei zweidimensionalen Plots werden von der Funktion AUTO die folgenden Aktionen ausgeführt:

- FUNCTION : Je nach Plotbereich von x, nimmt die Funktion eine Stichprobe in EQ; anschließend wird der untere Grenzwert und obere Grenzwert von y bestimmt.
- CONIC : Der Maßstab der y-Achse wird mit dem der x-Achse gleich gesetzt.
- POLAR : Basierend auf den Werten der unabhängigen Variablen (meistens θ) nimmt die Funktion eine Stichprobe in EQ; anschließend wird der untere Grenzwert und obere Grenzwert von x und y bestimmt.

- PARAMETRIC : Führt zu einem ähnlichen Ergebnis wie POLAR, bezogen auf die Werte des Parameters, der die Gleichungen für x und y definiert.
- TRUTH : Hat keine Wirkung.
- BAR : Der Bereich der x-Achse wird zwischen 0 und *n+1* gesetzt, wobei *n* die Anzahl der Elemente aus ΣDAT ist. Der Wertebereich von y hängt vom Inhalt von ΣDAT ab. Die unteren Grenzwerte und oberen Grenzwerten von y werden so bestimmt, dass die x-Achse immer in der Grafik eingeschlossen ist.
- HISTOGRAM : Ähnlich wie BAR.
- SCATTER : Setzt die Bereiche der x- und y-Achsen, abhängig vom Inhalt der unabhängigen und abhängigen Variablen aus ΣDAT.

<u>INFO (12)</u>
Die Funktion INFO ist ausschließlich interaktiv (d.h. sie ist nicht programmierbar). Wird die entsprechende Funktionstaste gedrückt, so werden Informationen über die aktuellen Plotparameter angezeigt.

<u>EQ (3)</u>
Der Variablenname EQ ist für das Speichern der aktuellen Gleichung in Plots bzw. dem Lösen von Gleichungen vorbehalten (siehe Kapitel ...). Die Funktionstaste EQ aus diesem Menü kann so verwendet werden, als wäre das Variablenmenü verfügbar. Wenn z.B. [EQ] gedrückt wird, werden die aktuellen Inhalte der Variablen angezeigt.

<u>ERASE (4)</u>
Die Funktion ERASE löscht den aktuellen Inhalt des Grafikfensters. Während der Programmierung wird mit dieser Funktion sichergestellt, dass das Grafikfenster vor dem Plotten einer neuen Grafik gelöscht wird.

<u>DRAX (5)</u>
Die Funktion DRAX zeichnet, falls diese sichtbar ist, die Achsen im aktuellen Plot .

DRAW (6)
Die Funktion DRAW zeichnet den Plot, der in PPAR definiert wurde.

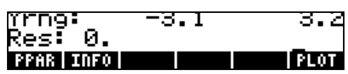
Das Menü PTYPE unter PLOT (1)

Das Menü PTYPE listet die Namen aller zweidimensionalen, im Taschenrechner vorprogrammierten Plottypen auf. Das Menü enthält die folgenden Funktionstasten:

Diese Tasten entsprechen den Plottypen *Function, Conic, Polar, Parametric, Truth* und *Diff Eq*, die weiter oben beschrieben wurden. Wird eine dieser Funktionstasten während der Eingabe eines Programms gedrückt, wird an dieser Stelle im Programm der entsprechende Funktionsaufruf eingefügt. Drücken Sie NXT PLOT, um zum Hauptmenü PLOT zurückzukehren.

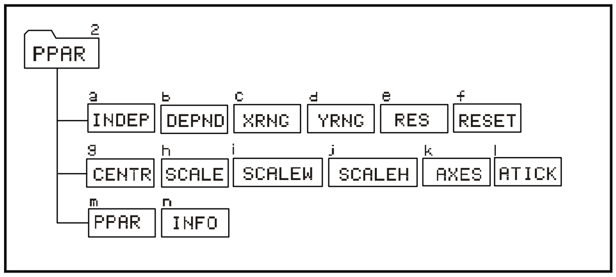
Das Menü PPAR (2)

Im PPAR-Menü stehen verschiedene Optionen für die PPAR-Variable zur Verfügung, welche den folgenden Funktionstasten entsprechen. Drücken Sie NXT, um zum nächsten Menü zu gelangen.

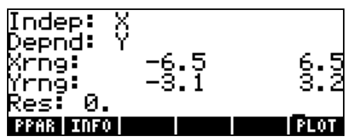


Hinweis: Die hier dargestellten SCALE-Befehle sind SCALE, SCALEW und SCALEH, und zwar in dieser Reihenfolge.

Die folgende Abbildung zeigt die im Menü PPAR enthaltenen Funktionen. Die Buchstaben oberhalb der Funktionen dienen als Referenz in der nachfolgenden Beschreibung der Funktionen.



INFO (n) und PPAR (m)
Wenn Sie INFO drücken oder ↱ PPAR eingeben, während Sie sich in diesem Menü befinden, erscheint eine Liste mit den aktuellen PPAR-Einstellungen. Beispiel:



Diese Informationen haben die folgende Bedeutung: X ist eine unabhängige Variable (Indep), Y ist die abhängige Variable (Depnd), der Bereich der x-Achse liegt zwischen –6,5 und 6,5 (Xrng), der Bereich der y-Achse liegt zwischen -3,1 und 3,2 (Yrng). Die letzte Information auf dem Bildschirm, der Wert von Res (Auflösung), bestimmt das Intervall der unabhängigen Variablen, die für die Erstellung des Plots verwendet wird.

Die Funktionstasten im Menü PPAR(2) sind Befehle, die in Programmen eingesetzt werden können. Diese Befehle lauten:

INDEP (a)
Der Befehl INDEP gibt die unabhängige Variable und deren Plotbereich an. Diese Angaben werden in der Variablen PPAR als dritter Parameter gespeichert. Der Standardwert ist 'X'. Der unabhängigen Variablen können folgende Werte zugeordnet werden:

- Name einer Variablen, z.B. 'Vel'
- Name einer Variablen aus einer Liste, z.B. { Vel }

* Name einer Variablen und ein Bereich aus einer Liste, z.B. { `Vel 0 20` }
* Ein Bereich ohne einen Variablen-Namen, z.B. { `0 20` }
* Zwei Werte, die einen Bereich darstellen, z.B. `0 20`

In einem Programm steht nach all diesen Angaben der Befehl INDEP.

DEPND (b)
Der Befehl DEPND gibt den Namen der abhängigen Variablen an. Bei TRUTH-Plots spezifiziert er außerdem den Plotbereich. Der Standardwert ist Y. Die Angaben für die Variable DEPND sind identisch mit denen für die Variable INDEP.

XRNG (c) und YRNG (d)
Der Befehl XRNG gibt den Plotbereich für die x-Achse an, und der Befehl YRNG den Plotbereich für die y-Achse. Diese Befehle erfordern die Eingabe der unteren bzw. oberen Grenzwerte von x bzw. y. Die Werte für die Achsenbereiche von x und y werden in den ersten zwei Elementen der Variablen PPAR als geordnete Paare (x_{min}, y_{min}) und (x_{max}, y_{max}) gespeichert. Die Standardwerte für x_{min} und x_{max} sind jeweils -6,5 und 6,5. Die Standardwerte für x_{min} und x_{max} sind jeweils –3,1 und 3,2.

RES (e)
Der Befehl RES (RESolution - Auflösung) gibt beim Erstellen eines bestimmten Plots das Intervall zwischen den Werten der unabhängigen Variablen an. Die Auflösung kann entweder in Benutzereinheiten als eine Realzahl oder in Pixel als binäre Integer-Zahl (Zahlen, die mit # anfangen, wie z.B. #10) angegeben werden. Die Auflösung wird in der Variablen PPAR als vierter Parameter gespeichert.

CENTR (g)
Der Befehl CENTR hat als Argument ein geordnetes Paar (x,y) oder einen Wert x, und passt die ersten zwei Elemente in der Variablen PPAR (z.B. (x_{min}, y_{min}) und (x_{max}, y_{max})) so an, dass die Mitte des Plots sich in (x,y) bzw. (x,0) befindet.

SCALE (h)

Der Befehl SCALE bestimmt den Maßstab des Plots, der als Anzahl von Benutzereinheiten pro Tick-Zeichen angegeben wird. Der Standardwert beträgt 1 Benutzereinheit pro Tick-Zeichen. Der Befehl SCALE benötigt zwei Zahlen als Argumente, x_{scale} und y_{scale} für die neuen horizontalen und vertikalen Maßstäbe. Dieser Befehl bewirkt, dass die Parameter (x_{min}, y_{min}) und (x_{max}, y_{max}) in PPAR an den gewünschten Maßstab angepasst werden. Die Mitte des Plots bleibt erhalten.

SCALEW (i)

Wird ein Faktor x_{factor} angegeben, multipliziert der Befehl SCALEW den horizontalen Maßstab mit diesem Faktor. Das W aus SCALEW steht für das englische Wort für Breite (Width). Nach der Ausführung des SCALEW-Befehls werden die Werte x_{min} und x_{max} in PPAR geändert.

SCALEH (j)

Wird ein Faktor y_{factor} angegeben, multipliziert der Befehl SCALEH den vertikalen Maßstab mit diesem Faktor. Das H aus SCALEH steht für das englische Wort für Höhe (Height). Nach der Ausführung des SCALEH-Befehls werden die Werte y_{min} und y_{max} in PPAR geändert.

Hinweis: Die von SCALE, SCALEW oder SCALEH vorgenommenen Änderungen können zum Vergrößern oder Verkleinern eines Plots verwendet werden.

ATICK (l)

Der Befehl ATICK (TICK-Zeichen für Achsen) wird dazu verwendet, Tick-Zeichenvermerke für die Achsen vorzunehmen. Der Befehl ATICK akzeptiert die folgenden Eingabewerte:

- Ein Realwert x: setzt die Tick-Vermerke für beide Achsen (x und y) auf x-Einheiten
- Eine Liste mit zwei Realwerten { x y }: setzt die Tick-Vermerke für beide Achsen (x und y) auf x- bzw. y-Einheiten
- Eine binäre Integer-Zahl #n: setzt die Tick-Vermerke für beide Achsen (x und y) auf #n Pixel

Eine Liste mit zwei binären Integer-Zahlen { #n #m }: setzt die Tick-Vermerke für beide Achsen (x und y) auf #n bzw. #m Pixel.

AXES (k)
Der Eingabewert für den Befehl AXES besteht entweder aus einem geordneten Paar (x,y) oder aus einer Liste {(x,y) *atick* "Beschriftung x-Achse" "Beschriftung y-Achse"}. Der Parameter *atick* wird für die Angabe der Tick-Vermerke verwendet, wie bereits beim Befehl ATICKbeschrieben wurde. Das geordnete Paar ist die Mitte des Plots. Wird für AXES nur ein geordnetes Paar angegeben, so wird nur der Ursprung der Achsen geändert. Das Argument im Befehl AXES, egal ob geordnetes Paar oder Werteliste, wird als fünfter Parameter in PPAR gespeichert.

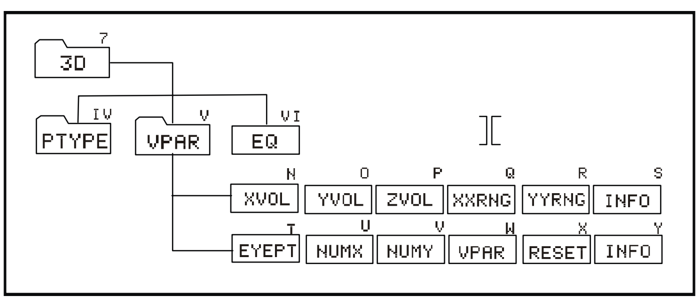
Drücken Sie PLOT, um zum Menü PLOT zurückzukehren.

Drücken Sie NXT, um zum zweiten Menü der Menüreihe PLOT zu gelangen.

RESET (f)
Diese Taste setzt die Plotparameter auf ihre Standardwerte zurück.

Das Menü 3D unter PLOT (7)

Das Menü 3D enthält zwei Untermenüs, PTYPE und VPAR, sowie eine Variable, EQ. Da die Bedeutung von EQ bereits erläutert wurde, werden wir uns hier auf die Inhalte der Menüs PTYPE und VPAR konzentrieren. Die nachstehende Darstellung zeigt die Struktur des 3D-Menüs.



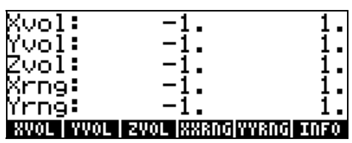
Das Menü PTYPE unter 3D (IV)

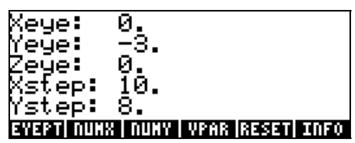
Das Menü PTYPE unter 3D enthält die folgenden Funktionen:

Diese Funktionen entsprechen den Grafikoptionen *Slopefield, Wireframe, Y-Slice, Ps-Contour, Gridmap* und *Pr-Surface*, die bereits in diesem Kapitel beschrieben wurden. Wird eine dieser Funktionstasten während der Eingabe eines Programms gedrückt, wird an dieser Stelle im Programm der entsprechende Funktionsaufruf eingefügt. Drücken Sie NXT 3D, um zum Hauptmenü 3D zurückzukehren.

Das Menü VPAR unter 3D (V)

Die Variable VPAR steht für Volume PARameter (Volumenparameter) und bezieht sich auf einem Parallelepipeden im Raum, in dessen Innerem eine beliebige dreidimensionale Grafik erzeugt wird. Wird im 3D-Menü [VPAR] gedrückt, werden die folgenden Funktionen aufgerufen. Drücken Sie NXT, um zum nächsten Menü zu gelangen.





Als Nächstes wird die Bedeutung dieser Funktionen erläutert:

INFO (S) und VPAR (W)

Wird INFO (S) gedrückt, erscheinen die Informationen, die in der oberen Abbildung auf der linken Seite zu sehen sind. Die Bereiche *Xvol*, *Yvol* und *Zvol* bestimmen die Größe des Parallelepipeds, in dem die Grafik eingefügt wird. *Xrng* und *Yrng* sind die Wertebereiche von x bzw. y als unabhängige Variablen in der Ebene x-y, die für die Erzeugung von Funktionen der Art z = f(x,y) verwendet werden.

Drücken Sie NXT und INFO (Y), um die Informationen zu erhalten, die in der oberen Abbildung auf der rechte Seite dargestellt sind. Dieses sind die Werte für die Position des Blickwinkels für die dreidimensionale Grafik (Xeye, Yeye, Zeye), sowie die Anzahl der Schritte in x und y, die für die Erstellung eines Rasters zur Darstellung der Oberfläche erforderlich sind.

XVOL (N), YVOL (O) und ZVOL (P)

Diese Funktion benötigt die Eingabe des unteren bzw. oberen Grenzwertes und bestimmt die Größe des Parallelepipeds, in dem die Grafik erzeugt werden soll. Diese Werte werden in der Variablen VPAR gespeichert. Die Standardwerte für die Bereiche XVOL, YVOL und ZVOL sind –1 bis 1.

XXRNG (Q) und YYRNG (R)

Diese Funktionen erfordern die Eingabe des unteren bzw. oberen Grenzwertes und bestimmen den Bereich der Variablen x und y, die für die Erstellung von Funktionen der Art z = f(x,y) verwendet werden. Die Standardwerte der Bereiche XXRNG und YYRNG sind die Gleichen wie bei XVOL und YVOL.

EYEPT (T)

Die Funktion EYEPT benötigt als Eingabe die reellen Werte x, y und z als Position des Blickwinkels der dreidimensionalen Grafik. Der Blickwinkel ist ein Punkt im Raum von dem aus die dreidimensionale Grafik beobachtet wird. Wird der Blickwinkel geändert, werden unterschiedliche Ansichten der Grafik dargestellt. Die Abbildung unten zeigt den Blickwinkel in Bezug auf den eigentlichen Grafikraum und seiner Projektion in der Anzeigenebene.

NUMX(U) und NUMY (V)

Die Funktionen NUMX und NUMY bestimmen die Anzahl der Punkte bzw. der Schritte für jede Richtung, die für die Erstellung des Bezugsrasters erforderlich sind, aus dem anschließend die Werte z = f(x,y) berechnet werden.

VPAR (W)

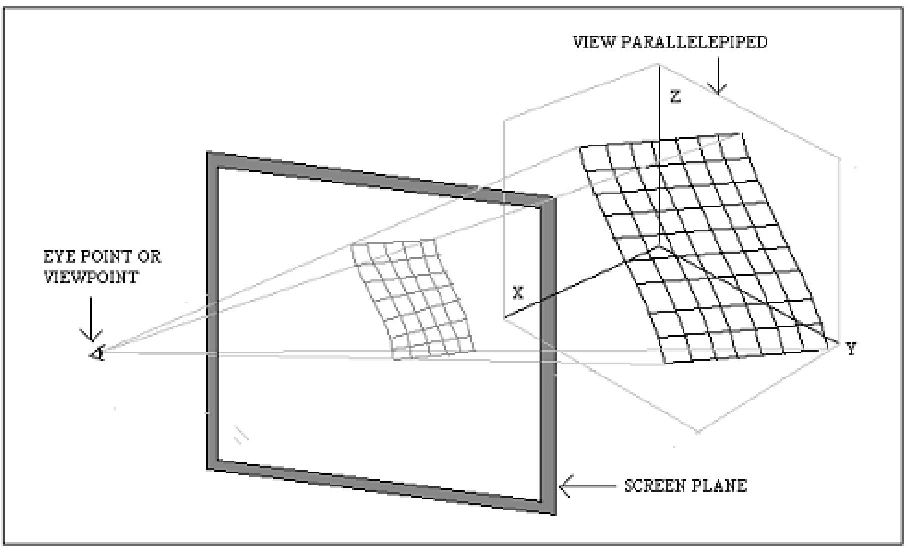
Dient lediglich als Referenz zur Variablen VPAR.

RESET (X)

Die Parameter auf dem Bildschirm werden auf ihre Standardwerte zurückgesetzt.

Drücken Sie NXT 3D , um zum Menü 3D zurückzukehren.

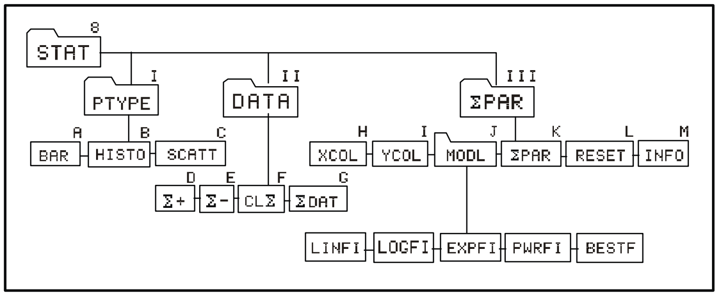
Drücken Sie PLOT, um zum Menü PLOT zurückzukehren.



Das Menü STAT unter PLOT

Das Menü STAT ermöglicht den Zugang zu Plots, die für statistische Analysen verwendet werden. Hier stehen die folgenden Untermenüs zu Verfügung:

Die folgende Abbildung zeigt die Struktur des Menüs STAT aus dem Menü PLOT. Die Zahlen und Buchstaben bei jeder Funktion dienen in der nachstehenden Beschreibung lediglich als Referenz.



Das Menü PTYPE unter STAT (I)

Das Menü PTYPE enthält die folgenden Funktionen:

Diese Tasten entsprechen den Plottypen *Bar (A), Histogram (B)* und *Scatter(C)*, die bereits beschrieben wurden. Wird eine dieser Funktionstasten während der Eingabe eines Programms gedrückt, wird an dieser Stelle im Programm der entsprechende Funktionsaufruf eingefügt. Drücken Sie STAT, um zum Menü STAT zurückzukehren.

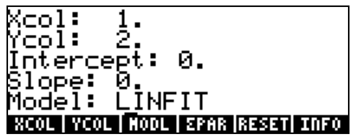
Das Menü DATA unter STAT (II)

Das Menü DATA enthält die folgenden Funktionen:

Die Funktionen in diesem Menü werden zum Manipulieren der statistischen Matrix ΣDAT verwendet. Mit den Funktionen Σ+ (D) und Σ- (E) werden Datenreihen zu bzw. aus der Matrix hinzugefügt oder entfernt. CLΣ (F) löscht die Matrix ΣDAT (G), während die Funktionstaste ΣDAT nur als Referenz für interaktive Anwendungen verwendet wird. Mehr zu diesen Funktionen finden Sie im Kapitel über statistische Anwendungen. Drücken Sie STAT, um zum Menü STAT zurückzukehren.

Das Menü ΣPAR unter STAT (III)

Das Menü ΣPAR enthält die folgenden Funktionen:

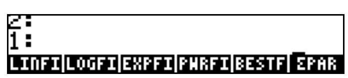


INFO (M) und ΣPAR (K)
Die Taste INFO in ΣPAR enthält die oben abgebildeten Informationen. Diese Informationen sind in der Variablen ΣPAR zu finden. Die angezeigten Werte sind die Standardwerte für Spalte x, Spalte y, den Achsenabschnitt und den Richtungskoeffizienten des Daten-Angleichungsmodells, sowie der Modelltyp, der an die Daten in ΣDAT angeglichen werden muss.

XCOL (H)
Mit dem Befehl XCOL wird angegeben, welche Spalten aus ΣDAT, falls mehrere vorhanden, die Spalte x oder die Spalte der unabgängigen Variablen sein werden.

YCOL (I)
Mit dem Befehl YCOL wird angegeben, welche Spalten aus ΣDAT, falls mehrere vorhanden, die Spalte y oder die Spalte der abhängigen Variablen sein werden.

MODL (J)
Der Befehl MODL bestimmt das auszuwählende Modell für die Datenangleichung in ΣDAT, sofern eine Datenangleichung implementiert wird. Drücken Sie MODL, um die verfügbaren Optionen anzusehen. Das folgende Menü erscheint:



Diese Funktionen entsprechen der linearen, logarithmischen, Exponential-, Potenz- oder der besten Angleichung. Mehr zur Datenangleichung finden Sie weiter unten in diesem Kapitel. Drücken Sie ΣPAR, um zum Menü ΣPAR zurückzukehren.

<u>ΣPAR (K)</u>
ΣPAR ist lediglich eine Referenz zur Variablen ΣPAR für interaktive Aktionen.

<u>RESET (L)</u>
Diese Funktion setzt die Inhalte von ΣPAR auf ihre Standardwerte zurück.

Drücken Sie NXT STAT, um zum Menü STAT zurückzukehren. Drücken Sie [PLOT], um zum Hauptmenü PLOT zurückzukehren.

Das Menü FLAG unter PLOT

Das Menü FLAG ist ein interaktives Menü, daher steht Ihnen die Auswahl zwischen den folgenden Optionen zur Verfügung:

- AXES : Wird diese Option ausgewählt, so werden die Achsen, falls sichtbar, innerhalb der Plot-Oberfläche bzw. des Plot-Volumens angezeigt.
- CNCT : Wird diese Option ausgewählt, wird der Plot so erzeugt, dass einzelne Punkte miteinander verbunden sind.
- SIMU : Wenn diese Option ausgewählt wurde, und es sollen mehr als eine Grafik im gleichen Achsensystem geplottet werden, dann werden alle Grafiken gleichzeitig erzeugt.

Drücken Sie PLOT, um zum Menü PLOT zurückzukehren.

Erzeugen von Plots durch Programme

Bevor ein Plot in einem Programm erzeugt werden kann, müssen die Variablen PPAR, ΣPAR, und/oder VPAR eingestellt werden, je nachdem, ob es sich um eine zweidimensionale Grafik handelt, die mittels einer Funktion, mittels Daten aus ΣDAT oder mittels einer dreidimensionalen Funktion definiert ist. Diese Variablen können Sie mithilfe der bereits vorher beschriebenen Befehle einrichten.

Als Nächstes wird das allgemeine Format von Variablen, die für die Erzeugung von verschiedenen Plottypen erforderlich sind, beschrieben.

Zweidimensionale Grafiken

Die zweidimensionalen Grafiken, die von den Funktionen *Function, Conic, Parametric, Polar, Truth* und *Differential Equation* erzeugt werden, verwenden die Variable PPAR im folgenden Format:

{ (x_{min}, y_{min}) (x_{max}, y_{max}) indep res axes ptype depend }

Die zweidimensionalen Grafiken, die aus Daten aus der statistischen Matrix erzeugt werden, nämlich *Bar, Histogram* und *Scatter*, verwenden die Variable ΣPAR im folgenden Format:

{ x-column y-column slope intercept model }

Genauso wird auch die Variable PPAR im oben angegebenen Format verwendet.

Die Bedeutung der verschiedenen Parameter aus PPAR und ΣPAR wurden in einem vorangegangenen Abschnitt erläutert.

Dreidimensionale Grafiken

Die dreidimensionalen Grafiken, nämlich die Optionen *Slopefield, Wireframe, Y-Slice, Ps-Contour, Gridmap* und *Pr-Surface*, verwenden die Variable VPAR im folgenden Format:

{x_{left}, x_{right}, y_{near}, y_{far}, z_{low}, z_{high}, x_{min}, x_{max}, y_{min}, y_{max}, x_{eye}, y_{eye}, z_{eye}, x_{step}, y_{step}}

Die Wertpaare von x, y und z haben die folgende Bedeutung:

- Dimensionen des Betrachtungsparallelepipeds (x_{left}, x_{right}, y_{near}, y_{far}, z_{low}, z_{high})
- Bereich der unabhängigen Variablen x und y (x_{min}, x_{max}, y_{min}, y_{max})
- Position des Betrachtungspunktes (x_{eye}, y_{eye}, z_{eye})
- Anzahl der Schritte in die Richtungen x und y (x_{step}, y_{step})

Dreidimensionale Grafiken benötigen auch die Variable PPAR mit den oben angegebenen Parametern.

Die Variable EQ

Außer für Plots, die auf ΣDAT basieren, müssen für alle anderen Plots die zu plottenden Funktionen durch Speichern der Ausdrücke oder Referenzen zu diesen Funktionen in der Variablen EQ definiert werden.

Kurz gesagt, muss für die Erzeugung eines Plots in einem Programm gegebenenfalls die Variable EQ geladen werden. Danach werden PPAR, PPAR und ΣPAR oder PPAR und VPAR geladen. Anschließend wird durch Angeben des entsprechenden Plottyps: FUNCTION, CONIC, POLAR, PARAMETRIC, TRUTH, DIFFEQ, BAR, HISTOGRAM, SCATTER, SLOPE, WIREFRAME, YSLICE, PCONTOUR, GRIDMAP oder PARSURFACE der Plot selbst erzeugt.

Beispiele von interaktiven Plots mit dem Menü PLOT

Um besser zu verstehen, wie das Programm mit den PLOT-Befehlen und -Variablen umgeht, versuchen Sie, mit dem Menü PLOT die folgenden interaktiven Plots zu erzeugen.

Beispiel 1 – Funktionsplot:

← USER F3	Aufrufen des Menüs PLOT (*)
PTYPE FUNCT	Wählen von FUNCTION als Plottyp
'√r' ENTER ← EQ	Speichern der Funktion '√r' in EQ
PPAR	Anzeigen der Plot-Parameter
ALPHA ← R ENTER INDEP	Definieren von 'r' als unabhängige Variable

ALPHA ← S ENTER DEPND	Definieren von 's' als abhängige Variable
1 +/- SPC 10 XRNG	Definieren Sie (-1, 10) als x-Bereich
1 +/- SPC 5 YRNG NXT	Definieren Sie (-1, 5) als y-Bereich
{ (0,0) {.4 .2} "Rs" "Sr"}	Definitionsliste der Achsen
AXES	Definieren der Achsenmitte, Ticks, Beschriftungen
NXT PLOT	Rückkehr Menü PLOT
ERASE DRAW NXT LABEL	Löschen des Bildes, Zeichnen der Achsen und Beschriftungen
NXT DRAW	Zeichnen der Funktion und Anzeigen des Bildes
EDIT NXT MENU	Entfernen der Menüeinträge
NXT NXT PICT CANCL	Rückkehr zur Normalanzeige

(*) Das Menü PLOT kann durch die benutzerdefinierte Taste F3 aufgerufen werden, wie in diesem Kapitel bereits beschrieben wurde.

Beispiel 2 – Parametrischer Plot (Gebrauch RAD als Winkel):

← USER F3	Menü PLOT aufrufen
PTYPE PARAM	Wählen von PARAMETRIC als Plottyp
{ 'SIN(t)+i*SIN(2*t)' } ENTER	Definieren der komplexen Funktion X+iY
← EQ	Speichern der komplexen Funktion in EQ
PPAR	Anzeigen der Plot-Parameter
{t 0 6,29} ENTER INDEP	Definieren Sie 't' als unabhängige Variable
ALPHA Y ENTER DEPND	Definieren Sie 'Y' als abhängige Variable
2.2 +/- SPC 2.2 XRNG	Definieren Sie (-2,2; 2,2) als x-Bereich
1.1 +/- SPC 1.1 YRNG NXT	Definieren Sie (-1,1; 1,1) als y-Bereich
{ (0,0) {.4 .2} "X(t)" "Y(t)"} ENTER	Definitionsliste der Achsen

AXES	Definieren der Achsenmitte, Ticks, Beschriftungen
NXT PLOT	Rückkehr zum Menü PLOT
ERASE DRAW NXT LABEL	Löschen des Bildes, Zeichnen der Achsen und Beschriftungen
NXT DRAW	Zeichnen der Funktion und Anzeigen des Bildes
EDIT NXT MENU NXT NXT PICT CANCL	Plot beenden

Beispiel 3 – <u>Polarplot:</u>

← USER F3	Menü PLOT aufrufen
PTYPE POLAR	POLAR als Plottyp wählen
‘1+SIN(θ)’ ENTER ← EQ	Speichern der komplexen Funktion r = f(θ) in EQ
PPAR	Anzeigen der Plot-Parameter
{ θ 0 6,29} ENTER INDEP	Definieren Sie ‘θ’ als unabhängige Variable
ALPHA Y ENTER DEPND	Definieren Sie ‘Y’ als abhängige Variable
3 +/- SPC 3 XRNG	Definieren Sie (-3,3) als x-Bereich
0,5 +/- SPC 2,5 YRNG NXT	Definieren Sie (-0,5; 2,5) als y-Bereich
{ (0,0) {.5 .5} “x” “y”} ENTER	Definitionsliste der Achsen
AXES	Definieren der Achsenmitte, Ticks, Beschriftungen
NXT PLOT	Rückkehr zum Menü PLOT
ERASE DRAW NXT LABEL	Löschen des Bildes, Zeichner der Achsen, Beschriftungen
NXT DRAW	Zeichnen der Funktion und Anzeige des Bildes
EDIT NXT MENU	Entfernen der Menüeinträge
NXT NXT PICT CANCL	Rückkehr zur Normalanzeige

Diese Beispiele zeigen ein Muster für interaktives Erstellen von zweidimensionalen Grafiken durch das Menü PLOT.

1 – PTYPE auswählen
2 – Die zu plottende Funktion in der Variablen EQ speichern (dabei auf das richtige Format achten, z.B. ‘X(t)+iY(t)’ bei PARAMETRIC)
3 – Namen (und gegebenenfalls den Bereich) der unabhängigen und abhängigen Variablen eingeben
4 – Die Angaben für die Achse als Liste eingeben { center atick x-label y-label }
5 – Erzeugen einer vollständig beschrifteten Grafik mit Achsen mit ERASE, DRAX, LABEL, DRAW

Auf die gleiche Weise können Plots mit einem Programm erzeugt werden, aber in diesem Fall muss nach dem Aufruf der Funktion DRAW auch der Befehl PICTURE eingefügt werden, um den Grafikbildschirm wieder in den Stack zu laden.

Beispiele von programm-generierten Plots

In diesem Abschnitt erfahren Sie, wie die Plots aus den letzten drei Beispielen anhand eines Programms erzeugt werden. Bevor Sie mit der Eingabe des Programms beginnen, aktivieren Sie das Menü PLOT, um die Eingabe von Grafikbefehlen (wie z.B. (← USER F3) zu erleichtern.

Beispiel 1 –Funktionsplot: Geben Sie das folgende Programm ein:

`«`	Starten des Programms
`{PPAR EQ} PURGE`	Die aktuellen Werte von PPAR und EQ löschen
`‘√r’ STEQ`	‘√r’ in EQ speichern
`‘r’ INDEP`	Unabhängige Variable auf ‘r’ setzen
`‘s’ DEPND`	Abhängige Variable auf ‘s’ setzen
`FUNCTION`	FUNCTION als Plottyp wählen
`{ (0.,0.) {.4 .2}`	
`“Rs” “Sr” } AXES`	Achseninformationen setzen
`-1. 5. XRNG`	x-Bereich setzen
`-1. 5. YRNG`	y-Bereich setzen
`ERASE DRAW DRAX LABEL`	Löschen des Bildes und zeichnen

	des Plots, der Achsen und der Beschriftungen
`PICTURE »`	Grafikbildschirm wieder in den Stack laden

Speichern Sie das Programm in der Variable PLOT1. Um das Programm auszuführen, drücken Sie VAR, falls erforderlich, und anschließend drücken Sie PLOT1.

Beispiel 2 – Parametrischer Plot: Geben Sie das folgende Programm ein:

`«`	Starten des Programms
`RAD {PPAR EQ} PURGE`	Auf Radianten wechseln, Löschen der Werte der Variablen
`'SIN(t)+i*SIN(2*t)' STEQ`	Speichern von 'X(t)+iY(t)' in EQ
`{ t 0. 6.29} INDEP`	Unabhängige Variable auf 'r' setzen, mit Bereich
`'Y' DEPND`	Abhängige Variable auf 'Y' setzen
`PARAMETRIC`	PARAMETRIC als Plottyp wählen
`{ (0.,0.) {.5 .5} "X(t)" "Y(t)" } AXES`	Achseninformationen setzen
`-2.2 2.2 XRNG`	x-Bereich setzen
`-1.1 1.1 YRNG`	y-Bereich setzen
`ERASE DRAW DRAX LABEL`	Löschen des Bildes und Zeichnen des Plots, der Achsen und der Beschriftungen
`PICTURE`	Grafikbildschirm wieder in den Stack laden
`»`	Programm beenden

Speichern Sie das Programm in der Variable PLOT2. Um das Programm auszuführen, drücken Sie, falls erforderlich VAR und anschließend PLOT2.

Beispiel 3 –Polarplot: Geben Sie das folgende Programm ein:

`«`	Starten des Programms
`RAD {PPAR EQ} PURGE`	Auf Radianten wechseln,

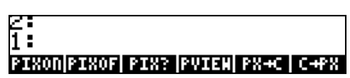
	Löschen der Werte der Variablen
`'1+SIN(θ)' STEQ`	'f(θ)' in EQ speichern
`{ θ 0. 6.29} INDEP`	Unabhängige Variable auf 'θ' setzen, mit Bereich
`'Y' DEPND`	Abhängige Variable auf 'Y' setzen
`POLAR`	POLAR als Plottyp wählen
`{ (0.,0.) {.5 .5} "x" "y"} AXES`	Achseninformationen setzen
`-3. 3. XRNG`	x-Bereich setzen
`-.5 2.5 YRNG`	y-Bereich setzen
`ERASE DRAW DRAX LABEL`	Löschen des Bildes und Zeichnen des Plots, der Achsen und der Beschriftungen
`PICTURE`	Grafikbildschirm wieder in den Stack laden
`»`	Programm beenden

Speichern Sie das Programm in der Variable PLOT3. Um das Programm auszuführen, drücken Sie, falls erforderlich VAR und anschließend PLOT3.

Diese Beispiele zeigen die Verwendung von PLOT-Befehlen in Programmen. Hierbei handelt es sich nur um einfache Übungen für die Programmierung von Plots. Sie sollten versuchen, eigene Übungen in der Programmierung von Plots durchzuführen.

Zeichenbefehle für die Programmierung

Abbildungen in einem Grafikfenster können Sie direkt aus einem Programm, mithilfe von Befehlen wie z.B. die aus dem Menü PICT, welche mit ← PRG NXT PICT aufgerufen werden, einfügen. In diesem Menü stehen nachfolgende Funktionen zur Verfügung. Drücken Sie NXT, um zum nächsten Menü zu gelangen.



Offensichtlich führen die Befehle LINE, TLINE und BOX die gleichen Operationen aus, wie die deren interaktiven Gegenstücke, vorausgesetzt die entsprechenden Eingaben werden getätigt. Diese, sowie die anderen Funktionen im Menü PICT beziehen sich auf das Grafikfenster, in dem die Bereiche von x und y in der Variable PPAR bestimmt werden, wie dies bereits weiter oben für unterschiedliche Typen von Grafiken demonstriert wurde. Nachstehend werden die Funktionen des Befehls PICT beschrieben.

PICT

Diese Funktionstaste entspricht der Variablen PICT, in welcher die aktuellen Inhalte des Grafikfensters gespeichert werden. Der Name der Variablen kann jedoch nicht zwischen Anführungszeichen gesetzt und in der Variablen selber können nur Grafikobjekte gespeichert werden. In dieser Hinsicht unterscheidet sich PICT von allen anderen Variablen im Taschenrechner.

PDIM

Die Funktion PDIM benötigt als Eingabe, entweder zwei geordnete Paare (x_{min},y_{min}) (x_{max} y_{max}) oder zwei binäre Integer-Zahlen #w und #h. PDIM bewirkt, dass die aktuellen Inhalte von PICT durch einen leeren Bildschirm ersetzt werden. Wenn das Argument (x_{min},y_{min}) (x_{max} y_{max}) ist, werden diese Werte zum Bereich der benutzerdefinierten Koordinaten in PPAR. Ist das Argument jedoch #w und #h, bleiben die Bereiche der benutzerdefinierten Koordinaten in PPAR unverändert, aber die Größe der Grafik ändert sich auf #h $\times$ #w Pixel.

PICT und der Grafikbildschirm

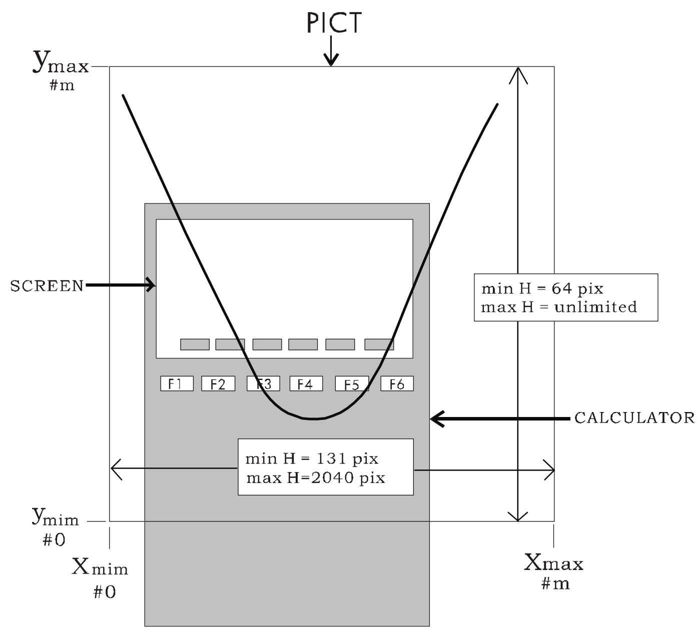
PICT, der Speicherplatz des aktuellen Graphen, kann als ein zweidimensionaler Graph betrachtet werden, der eine Mindestgröße von 131 Pixel breit und 64 Pixel hoch enthält. Die maximale Breite von PICT beträgt 2048 Pixel, ohne Beschränkung der maximalen Höhe. Die Pixel sind die einzelnen Punkte des Displays, die ein- oder ausgeschaltet (dunkel bzw. hell) werden, um Text oder Graphen darzustellen. Das Display des Taschenrechners besteht aus 131 mal 64 Pixel, was der Mindestgröße von PICT entspricht. Falls PICT größer als das Display ist, kann der Graph PICT als eine zweidimensionale Domäne betrachtet werden, die auf dem Display hin und her bewegt werden kann, wie in der nachstehenden Abbildung zu sehen ist.

LINE

Dieser Befehl benötigt als Eingabe zwei geordnete Paare (x_1,y_1) (x_2, y_2) oder zwei Paare von Pixelkoordinaten $\{\#n_1\ \#m_1\}$ $\{\#n_2\ \#m_2\}$. Der Befehl zeichnet eine Linie zwischen den angegebenen Koordinaten.

TLINE

Dieser Befehl (Toggle LINE – umgekehrte Linie) benötigt als Eingabe zwei geordnete Paare (x_1,y_1) (x_2, y_2) oder zwei Paare von Pixelkoordinaten $\{\#n_1\ \#m_1\}$ $\{\#n_2\ \#m_2\}$. Der Befehl zeichnet eine Linie zwischen den angegebenen Koordinaten und schaltet anschließend die eingeschalteten Pixel der Linie aus und umgekehrt.



BOX

Dieser Befehl benötigt als Eingabe zwei geordnete Paare (x_1,y_1) (x_2, y_2) oder zwei Paare von Pixelkoordinaten $\{\#n_1\ \#m_1\}$ $\{\#n_2\ \#m_2\}$. Der Befehl zeichnet ein Kästchen, bei dem die Diagonalen von den zwei eingegebenen Koordinatenpaaren bestimmt werden.

ARC

Dieser Befehl wird zum Zeichnen eines Bogens verwendet. ARC benötigt nachfolgende Objekte als Eingabe:

- Koordinaten der Bogenmitte als (x,y) in benutzerdefinierten Koordinaten oder {#n, #m} in Pixel.
- Radius des Bogens als r (benutzerdefinierte Koordinaten) oder #k (Pixel).
- Anfangswinkel θ_1 und Endwinkel θ_2.

PIX?, PIXON und PIXOFF

Diese Funktionen benötigen als Eingabe die Koordinaten des Punktes in Benutzereinheiten (x,y) oder in Pixel {#n, #m}.

- PIX? überprüft, ob an der Position (x,y) bzw. {#n, #m} der Pixel an ist.
- PIXOFF schaltet den Pixel an der Position (x,y) bzw. {#n, #m} aus.
- PIXON schaltet den Pixel an der Position (x,y) bzw. {#n, #m} ein.

PVIEW

Dieser Befehl benötigt als Eingabe die Koordinaten eines Punktes in Benutzereinheiten (x,y) oder in Pixel {#n, #m}, und setzt die Inhalte aus PICT mit der linken oberen Ecke an den angegebenen Punkt. Wird eine leere Liste als Argument angegeben, erfolgt die Ausgabe des Bildes in der Mitte des Displays. PVIEW aktiviert den Grafikcursor bzw. das Bildmenü nicht. Diese können mit PICTURE aktiviert werden.

PX→C

Die Funktion PX→C konvertiert Pixelkoordinaten {#n #m} in Koordinaten von Benutzereinheiten (x,y).

C→PX

Die Funktion C→PX konvertiert Koordinaten von Benutzereinheiten (x,y) in Pixelkoordinaten {#n #m}.

Programmierbeispiele mit Zeichenfunktionen

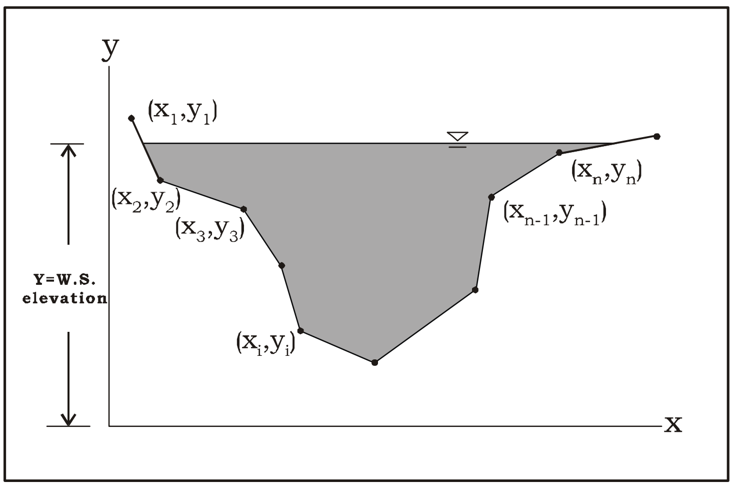
In diesem Abschnitt werden die oben beschriebenen Befehle für die Erzeugung von Grafiken mithilfe von Programmen verwendet. Die Programmcodes sind auf der mitgelieferten Diskette bzw. CD-ROM zu finden.

<u>Beispiel 1</u> – Ein Programm, das Zeichenbefehle verwendet
Das folgende Programm erzeugt eine Zeichnung auf dem Grafikbildschirm. (Dieses Programm wurde nur dafür geschrieben, die Befehle für die Erzeugung von Zeichnungen auf dem Bildschirm zu erläutern).

«	Starten des Programms
DEG	Wählen Sie Grad für Winkelmaße
0. 100. XRNG	x-Bereich setzen
0. 50. YRNG	y-Bereich setzen
ERASE	Bild löschen
(5., 2.5) (95., 47,5) BOX	Kästchen zwischen (5,5) und (95,95) zeichnen
(50., 50.) 10. 0. 360. ARC	Kreis mit der Mitte in (50,50) und r =10 zeichnen
(50., 50.) 12. –180. 180. ARC	Kreis mit der Mitte in (50,50) und r =12 zeichnen
1 8 FOR j	8 Linien im Inneren des Kreises zeichnen
(50., 50.) DUP	Die Linienmitte befindet sich im Punkt (50,50)
'12*COS(45*(j-1))' →NUM	Berechnen von x , das andere Ende bei 50 + x
'12*SIN(45*(j-1))' →NUM	Berechen von y , das andere Ende bei 50 + y
R → C	Konvertieren von x y in (x,y), komplexe Zahlen
+	(50,50) mit (x,y) addieren
LINE	Zeichnen der Linie
NEXT	Ende der FOR-Schleife
{ } PVIEW	Anzeigen des Bildes
»	

Beispiel 2 – Ein Programm zum Plotten eines natürlichen Flussquerschnitts
Diese Anwendung kann sich bei der Feststellung der Fläche und der befeuchteten Randbereiche eines Flussquerschnitts als sehr nützlich erweisen. Normalerweise wird ein Flussquerschnitt anhand einer Reihe von Punkten, und zwar die Koordinaten x und y bezogen auf ein beliebiges Koordinatensystem, vermessen. Diese Punkte werden geplottet und danach eine Entwurfszeichnung des Querschnitts in einer bestimmten Höhe der Wasseroberfläche erzeugt. Die folgende Abbildung zeigt die in diesem Abschnitt beschriebenen Bedingungen.

Dieses Programm, das auf der mitgelieferten Diskette bzw. CD-ROM zu finden ist, verwendet vier Unterprogramme: FRAME, DXBED, GTIFS und INTRP. Das Hauptprogramm, XSECT genannt, benötigt als Eingabe eine Matrix mit den Werten von x und y, sowie die Höhe der Wasseroberfläche Y (siehe Abbildung), in dieser Reihenfolge. Das Programm erzeugt eine Grafik des Querschnitts, wobei die Eingabedaten auf der Grafik als Punkte dargestellt werden. Außerdem wird die freie Oberfläche des Querschnitts angezeigt.



Wir empfehlen Ihnen ein neues Unterverzeichnis für die Speicherung der Programme zu erstellen. Sie können das Unterverzeichnis RIVER (Fluss) nennen, da es sich hier um unregelmäßige, für Flüsse typischen Querschnitte mit offenen Kanälen handelt.

Um das Programm XSECT auszuführen, verwenden Sie die folgenden Datensets. Diese werden als Matrizen mit zwei Spalten eingegeben, wobei die erste Spalte für x und die zweite für y vorgesehen ist. Die Matrizen werden in Variablen mit Namen wie z.B. XYD1 (X-Y Datenset 1) und XYD2 (X-Y Datenset 2) gespeichert. Um das Programm auszuführen, holen Sie ein Datenset in den Stack (z.B. VAR XYD1), geben Sie danach die Höhe der Wasseroberfläche (z.B. 4,0) ein, und drücken Sie anschließend XSECT. Der

Taschenrechner zeigt eine Entwurfszeichnung des Querschnitts mit der entsprechenden Wasseroberfläche an. Drücken Sie ON, um die Grafikanzeige zu beenden.

Versuchen Sie folgende Beispiele:

XVOL 2 XSECT
XVOL 3 XSECT
XVOL 4 XSECT
XVOL 6 XSECT

Bei der Ausführung des Programms XSECT müssen Sie ein wenig Geduld haben. Aufgrund der hohen Anzahl von Grafikfunktionen, ohne die numerische Iterationen mitzuzählen, kann es eine Weile dauern, bis die Grafik erzeugt wird (etwa 1 Minute).

Datenset 1

X	y
0,4	6,3
1,0	4,9
2,0	4,3
3,4	3,0
4,0	1,2
5,8	2,0
7,2	3,8
7,8	5,3
9,0	7,2

Datenset 2

x	y
0,7	4,8
1,0	3,0
1,5	2,0
2,2	0,9
3,5	0,4
4,5	1,0
5,0	2,0
6,0	2,5
7,1	2,0
8,0	0,7
9,0	0,0
10,0	1,5
10,5	3,4
11,0	5,0

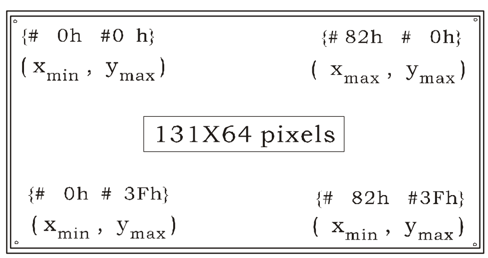
Hinweis: Das Programm FRAME, wie es ursprünglich geschrieben wurde (siehe Diskette oder CD-ROM) behält nicht den richtigen Maßstab der Grafik. Wenn Sie den richtigen Maßstab behalten möchten, ersetzen Sie FRAME durch das folgende Programm:

```
« STOΣ MINΣ MAXΣ 2 COL→ DUP →COL DROP – AXL ABS AXL 20
/ DUP NEG SWAP 2 COL→ + →ROW DROP SWAP → yR xR « 131
DUP R→B SWAP yR OBJ→ DROP – xR OBJ→ DROP - / * FLOOR
R→B PDIM yR OBJ→ DROP YRNG xR OBJ→ DROP XRNG ERASE » »
```

Dieses Programm beschränkt die Breite der Variablen PICT auf 131 Pixel – die minimale Pixelgröße der horizontalen Achse – und passt die Anzahl der Pixel der vertikalen Achse so an, dass der 1:1 Maßstab zwischen der vertikalen und der horizontalen Achse erhalten bleibt.

Pixelkoordinaten

Die folgende Abbildung zeigt die Grafikkoordinaten für einen typischen (minimalen) Bildschirm von 131×64 Pixeln. Pixelkoordinaten werden von der linken oberen Ecke des Bildschirms {# 0h # 0h} gemessen. Dies entspricht den benutzerdefinierten Koordinaten (x_{min}, y_{max}). Die maximalen Koordinaten in Pixel entsprechen der rechten unteren Ecke des Bildschirms {# 82h #3Fh}; dieser Punkt entspricht den benutzerdefinierten Koordinaten (x_{max}, y_{min}). Die Koordinaten der zwei Ecken werden in der nachfolgenden Abbildung sowohl in Pixel als auch in benutzerdefinierten Koordinaten dargestellt.

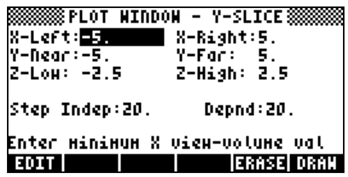


Animieren von Grafiken

In diesem Abschnitt erfahren Sie, wie Sie mit dem Plottyp Y-Slice animierte Grafiken erzeugen können. Nehmen wir an, dass Sie die Wanderwelle f(X,Y) = 2,5 sin(X-Y) animieren wollen. Betrachten wir X als Zeit in der Animation, die Plots der Funktion f(X,Y) gegen Y als unterschiedliche Werte für X erstellt. Zum Erstellen der Grafik gehen Sie wie folgt vor:

- ← 2D/3D gleichzeitig drücken. Wählen Sie Y-Slice für TYPE. '2.5*SIN(X-Y)' für EQ. 'X' für INDEP. Drücken Sie NXT OK.

- ← WIN gleichzeitig drücken (im RPN-Modus). Verwenden Sie die folgenden Werte:



- Drücken Sie ERASE DRAW. Geben Sie dem Taschenrechner ein wenig Zeit, damit er alle erforderlichen Grafiken erzeugen kann. Wenn die Operation abgeschlossen ist, wird auf dem Bildschirm eine sinusförmige Wanderwelle angezeigt.

Animation von Grafiksammlungen

Mit der Funktion ANIMATE kann eine Reihe von Grafiken, die in den Stack geladen wurden, animiert werden. Sie können eine Grafik auf dem Grafikbildschirm durch Verwenden der Befehle aus den Menüs PLOT und PICT erzeugen. Um die erzeugte Grafik in den Stack abzulegen, verwenden Sie PICT RCL. Wenn Sie *n* Grafiken auf den Ebenen *n* bis *1* des Stacks haben, können Sie mit dem Befehl *n* ANIMATE aus den Grafiken im Stack eine Animation erstellen.

Beispiel 1 – Animation einer Welle auf der Wasseroberfläche

Als Beispiel geben Sie das folgende Programm ein, welches 11 Grafiken erzeugt, sodass in der Mitte des Grafikbildschirms ein Kreis abgebildet wird und der Radius dieses Kreises in jeder nachfolgenden Grafik um einen konstanten Wert erhöht wird.

« Starten des Programms

RAD	Setzen der Winkeleinheit auf Radianten
131 R→B 64 R→B PDIM	PICT auf 131x64 Pixel setzen
0 100 XRNG 0 100 YRNG	x- und y-Bereiche auf 0-100 setzen
1 11 FOR j	Schleife bei j = 1 ..11 starten
ERASE	Aktuelles PICT löschen
(50., 50.) '5*(j-1)' →NUM	Die Mittelpunkte der Kreise (50,50)
0 '2*π' →NUM ARC	Kreis mit einem Radius von r = 5(j-1) zeichnen
PICT RCL	Aktuelles PICT in den Stack ablegen
NEXT	Ende der FOR-NEXT-Schleife
11 ANIMATE	Animieren
»	Programm beenden

Speichern Sie dieses Programm in der Variablen PANIM (Plot ANIMation). Um das Programm auszuführen, drücken Sie (falls erforderlich) VAR und PANIM. Der Taschenrechner braucht über eine Minute, um die Grafiken zu erzeugen und die Animation zu starten. Deshalb sollten Sie sich ein wenig gedulden. Auf dem Display erscheint das Symbol einer Sanduhr für eine scheinbar lange Zeit. Danach erscheint eine Animation, die an Wellen erinnert, die sich beim Eintauchen eines kleinen Steines auf der Wasseroberfläche bilden. Drücken Sie ON, um die Animation zu beenden.

Die 11 Grafiken, die vom Programm erzeugt wurden, stehen im Stack weiterhin zur Verfügung. Wenn Sie die Animation erneut ausführen möchten, verwenden Sie einfach: 11 ANIMATE. (Die Funktion ANIMATE können Sie mit ← PRG NXT GROB NXT ANIMA aufrufen). Die Animation wird erneut ausgeführt. Drücken Sie ON, um die Animation wieder zu beenden. Beachten Sie, dass die Zahl 11 weiterhin auf Ebene 1 des Stacks bleibt. Drücken Sie ←, um diese aus dem Stack zu entfernen.

Nehmen wir an, dass Sie die Zahlen, die diese Animation bilden, in einer Variablen speichern wollen. Sie können eine Liste, sagen wir WLIST, mit diesen Zahlen zusammenstellen, wenn Sie wie folgt vorgehen:

/ / PRG TYPE →LIST ' ALPHA ALPHA W L I S T ALPHA STO▸

Drücken Sie VAR, um die Liste mit den Variablen wiederherzustellen. Die Variable WLIST sollte jetzt in der Auflistung der Funktionstasten erscheinen. Verwenden Sie das folgende Programm, um die Liste der Variablen erneut zu animieren:

«	Starten des Programms
WLIST	WLIST-Liste in den Stack ablegen
OBJ→	Liste zerlegen, Stack-Ebene 1 = 11
ANIMATE	Animation starten
»	Programm beenden

Speichern Sie dieses Programm in der Variablen RANIM (Re-ANIMate). Drücken Sie RANIM, um das Programm auszuführen.

Das folgende Programm animiert die Grafiken aus WLIST vor und zurück:

«	Starten des Programms
WLIST DUP	WLIST-Liste in den Stack ablegen, eine Kopie erstellen
REVLIST +	Reihenfolge umkehren, die beiden Listen verknüpfen
OBJ→	Liste in seine Elemente zerlegen, Ebene 1 = 22
ANIMATE	Animation starten
»	Programm beenden

Speichern Sie dieses Programm in der Variablen RANI2 (Re-ANImate Version 2). Drücken Sie RANI2, um das Programm auszuführen. Die Animation simuliert jetzt eine Welle auf der Wasseroberfläche, die von den Wänden eines kreisförmigen Behälters zur Mitte reflektiert wird. Drücken Sie ON, um die Animation zu beenden.

Beispiel 2 - Animation von Potenzfunktionen
Nehmen wir an, Sie wollen den Plot der Funktionen f(x) = x^n, n = 0, 1, 2, 3, 4 im selben Achsensystem animieren. Dazu können Sie das folgende Programm verwenden:

«	Starten des Programms
RAD	Winkeleinheit auf Radianten setzen
131 R→B 64 R→B PDIM	PICT auf 131×64 Pixel setzen
0 2 XRNG 0 20 YRNG	x- und y-Bereiche setzen
0 4 FOR j	Schleife wird bei j = 0,1,…,4 gestartet
'X^j' STEQ	'X^j' in der Variable EQ ablegen
ERASE	Aktuelles PICT löschen
DRAX LABEL DRAW	Achsen, Beschriftungen, Funktion zeichnen
PICT RCL	Aktuelles PICT in den Stack ablegen
NEXT	Ende der FOR-NEXT-Schleife
5 ANIMATE	Animieren
»	

Speichern Sie dieses Programm in der Variablen PWAN (PoWer function Animation – Animation der Potenzfunktion). Um das Programm auszuführen, drücken Sie, (falls erforderlich) VAR und PWAN. Der Taschenrechner zeichnet jede einzelne Potenzfunktion bevor die Animation, in der die fünf Funktionen eine nach der anderen geplottet werden, gestartet wird. Drücken Sie ON, um die Animation zu beenden.

Weitere Informationen zu der Funktion ANIMATE

Die Funktion ANIMATE, wie sie in den zwei vorigen Beispielen verwendet wurde, benötigt als Eingabe die zu animierenden Grafiken sowie die Anzahl dieser Grafiken. Sie können zusätzliche Informationen für die Animation angeben, wie z.B. das Zeitintervall zwischen der Darstellung der Grafiken und die Anzahl der Wiederholungen der Darstellung. Das allgemeine Format der Funktion ANIMATE für solche Fälle ist wie folgt:

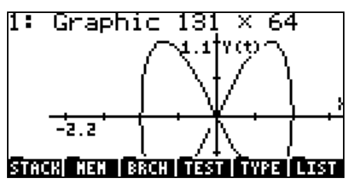
```
n-graphs   { n {#X #Y} delay rep }  ANIMATE
```

n stellt die Anzahl der Grafiken dar, {#X #Y} steht für die Pixelkoordinaten der rechten unteren Ecke der zu plottenden Fläche (siehe Abbildung unten), delay sind die Sekunden zwischen den Darstellungen von aufeinander folgenden Grafiken der Animation und rep ist die Anzahl der Wiederholungen der Animation.

Grafikobjekte (GROBs)

Das Wort GROB kommt von GRaphics OBjects (Grafikobjekte) und es entspricht der Pixel-für-Pixel-Beschreibung eines Bildes auf dem Display. Somit wird ein Bild in ein GROB konvertiert, entsteht aus diesem eine Reihe von binären Ziffern (m Englisch *b*inary dig*its* = *bits*), d.h. Nullen und Einsen. Nehmen wir das folgende Beispiel, um die GROBs, sowie die Konvertierung von Bildern in GROBs zu erläutern.

Sobald eine Grafik auf dem Taschenrechner erstellt wird, wird diese Grafik als Inhalt einer besonderen Variablen PICT gespeichert. Der Inhalt von PICT kann mit der nachfolgenden Tastenfolge angezeigt werden:
PICT RCL (← PRG NXT PICT PICT ← RCL).
Auf Ebene 1 des Stacks wird line Graphic 131 64 angezeigt (falls die standardmäßige Größe des Displays verwendet wird), gefolgt von einer Entwurfszeichnung des oberen Teils der Grafik. Beispiel:



Drücken Sie ▼, wird die Grafik aus Ebene 1 auf dem Bildschirm des Taschenrechners angezeigt. Drücken Sie CANCL, um zur Normalanzeige des Taschenrechners zurückzukehren.

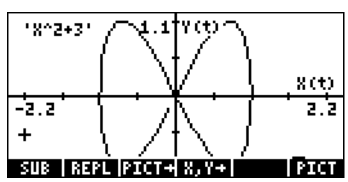
Die Grafik in Ebene 1 ist immer noch nicht im GROB-Format, obwohl sie definitionsmäßig ein Grafikobjekt ist. Um die Grafik aus dem Stack in ein GROB zu konvertieren, müssen Sie die folgende Eingabe vornehmen:

3 ENTER ← PRG NXT GROB →GROB. Nun erscheinen die folgenden Informationen auf Ebene 1:

Der erste Teil der Beschreibung ähnelt dem, was wir ursprünglich hatten, und zwar `Graphic 131×64`, aber diesmal wird es als `Graphic 13128 × 8` dargestellt. Die Grafikanzeige jedoch wird hier von einer Reihe von Nullen und Einsen ersetzt, welche die Pixel des Originalgraphen darstellen. Somit wurde der Originalgraph in seine entsprechende Bit-Darstellung konvertiert.

Auch Gleichungen können in GROBs konvertiert werden. Geben Sie mit dem EquationWriter die Gleichung 'X^2+3' auf Ebene 1 des Stacks ein und drücken Sie anschließend 1 ENTER ← PRG NXT GROB →GROB. In Stack-Ebene 1 befindet sich nun ein GROB wie folgt dargestellt:

Diese zum Grafikobjekt konvertierte Gleichung kann jetzt in der Grafikanzeige angesehen werden. Zum Wiederherstellen der Grafik drücken Sie ◄. Bewegen Sie den Cursor an eine leere Stelle des Graphen und drücken Sie EDIT NXT NXT REPL. Die Gleichung 'X^2-5' wird in den Graphen eingefügt. Beispiel:



Durch dieses Einfügen von Gleichungen und Text in GROBs können diese zum Dokumentieren der Grafiken eingesetzt werden.

Das Menü GROB

Das Menü GROB kann mit ← PRG NXT GROB →GROB aufgerufen werden und enthält die folgenden Funktionen. Um zum nächsten Menü zu gelangen, drücken Sie NXT.

→GROB

Von diesen Funktionen haben wir bereits: SUB, REPL (aus dem Grafikmenü EDIT), ANIMATE [ANIMA] und →GROB verwendet. ([*PRG*] ist nur eine Möglichkeit, zum Programmmiermenü zurückzukehren.) Sie haben sicher bemerkt, dass in den letzten zwei Beispielen bei der Konvertierung der Grafik in ein GROB die Zahl 3 verwendet wurde, wobei bei der Konvertierung der Gleichung in ein GROB die Zahl 1 verwendet wurde. Dieser Parameter der Funktion →GROB gibt die Größe des zu konvertierenden Objekts an, und zwar 0 oder 1 für kleine Objekte, 2 für mittelgroße Objekte und 3 für große Objekte. Nachstehend werden die übrigen Funktionen des Menüs GROB beschrieben.

BLANK

Die Funktion BLANK, mit den Argumenten #n und #m, erstellt ein leeres Grafikobjekt mit der Breite und Höhe, die durch die Werte #n bzw. #m definiert werden. Diese Funktion ist vergleichbar mit der Funkzion PDIM aus dem Menü GRAPH.

GOR

Die Funktion GOR (Graphics OR) benötigt als Eingabe $grob_2$ (ein Ziel-GROB), ein Koordinatensystem und $grob_1$. Diese Funktion bewirkt die Überlagerung von $grob_1$ über $grob_2$ (oder PICT), beginnend bei den angegebenen Koordinaten. Die Koordinaten können in Benutzereinheiten (x,y) oder in Pixel {#n #m} angegeben werden. GOR verwendet die OR-Funktion, um den Zustand (an oder aus) jedes Pixels im überlappenden Bereich zwischen $grob_1$ und $grob_2$ zu ermitteln.

GXOR

Die Funktion GXOR (Graphics XOR) hat die gleiche Funktion wie GOR, in diesem Fall wird der endgültige Zustand der Pixel jedoch im überlappenden Bereich zwischen den Objekten *grob*$_1$ und *grob*$_2$ mit XOR ermittelt.

Hinweis: Wird in GOR oder GXOR *grob2* durch PICT ersetzt, erfolgt keine Ausgabe. Um die Ausgabe anzusehen, müssen Sie PICT mit PICT RCL oder PICTURE wieder in den Stack laden.

→LCD

Nimmt das angegebene GROB und zeigt es auf dem Bildschirm beginnend in der linken oberen Ecke.

LCD→

Kopiert den Inhalt des Stacks und der Menüanzeige in ein GROB von 131 x 64 Pixeln.

SIZE

Die Funktion SIZE, wenn sie auf ein GROB angewendet wird, zeigt die Größe des GROBs in Form zweier Zahlen. Die erste Zahl (in Stack-Ebene 2) ist die Breite des Grafikobjekts, die zweite Zahl (in Stack-Ebene 1) ist die Höhe.

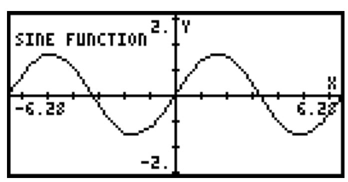
Beispiel eines Programms unter Verwendung von GROB

Das folgende Programm erzeugt den Graphen einer Sinusfunktion, einschließlich eines Rahmens, der mit der Funktion BOX gezeichnet wird, und ein GROB, das der Beschriftung dient. Nachfolgend das Listing des Programmcodes:

«	Starten des Programms
RAD	Winkeleinheit auf Radianten setzen
131 R→B 64 R→B PDIM	PICT auf 131×64 Pixel setzen
-6,28 6,28 XRNG –2. 2. YRNG	x- und y-Bereiche setzen
FUNCTION	FUNCTION als Typ für den Graphen auswählen
'SIN(X)' STEQ	Speichern der Sinusfunktion in EQ

ERASE DRAX LABEL DRAW	Löschen, dann Zeichnen der Achsen, Beschriftungen, Grafik
(-6,28;-2.) (6,28;2.) BOX	Rahmen um die Grafik zeichnen
PICT RCL	Inhalt von PICT im Stack laden
"SINE FUNCTION"	Zeichenkette für die Grafikbeschriftung in den Stack laden
1 →GROB	Zeichenkette in einen kleinen GROB konvertieren
(-6.; 1,5) SWAP	Koordinaten des GROBs für die Beschriftung
GOR	PICT mit der Beschriftung GROB kombinieren
PICT STO	Kombiniertes GROB in PICT speichern
{ } PVIEW	PICT in den Stack laden
»	Beenden des Programms

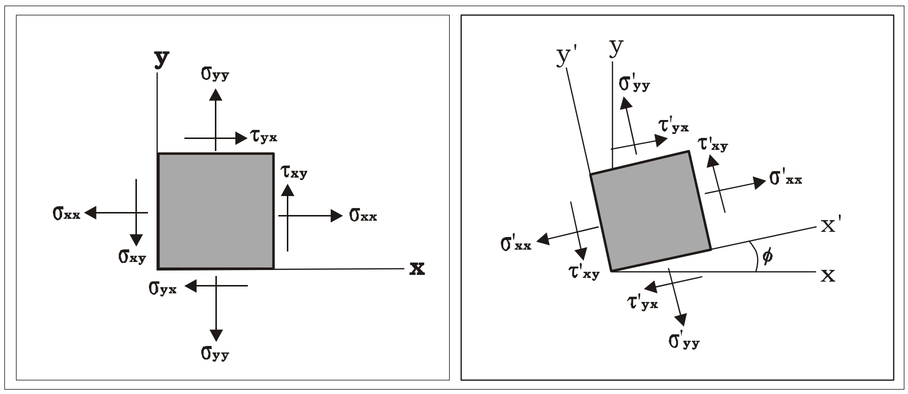
Speichern Sie das Programm unter den Namen GRPR (GROB PRogram). Zum Ausführen des Programms drücken Sie GRPR. Die Ausgabe sieht wie folgt aus:



Ein Programm mit Plot- und Zeichenfunktionen

In diesem Abschnitt entwickeln wir ein Programm, mit dem der Mohr'sche Kreis für einen gegebenen zweidimensionalen Spannungszustand erzeugt, gezeichnet und beschriftet wird. Die Abbildung auf der linken Seite zeigt einen gegebenen Spannungszustand in zwei Dimensionen, wobei σ_{xx} und σ_{yy} normale Spannungen und $\tau_{xy} = \tau_{yx}$ Schubspannungen sind. Die Abbildung auf der rechten Seite zeigt den Spannungszustand, wenn das Element um einen

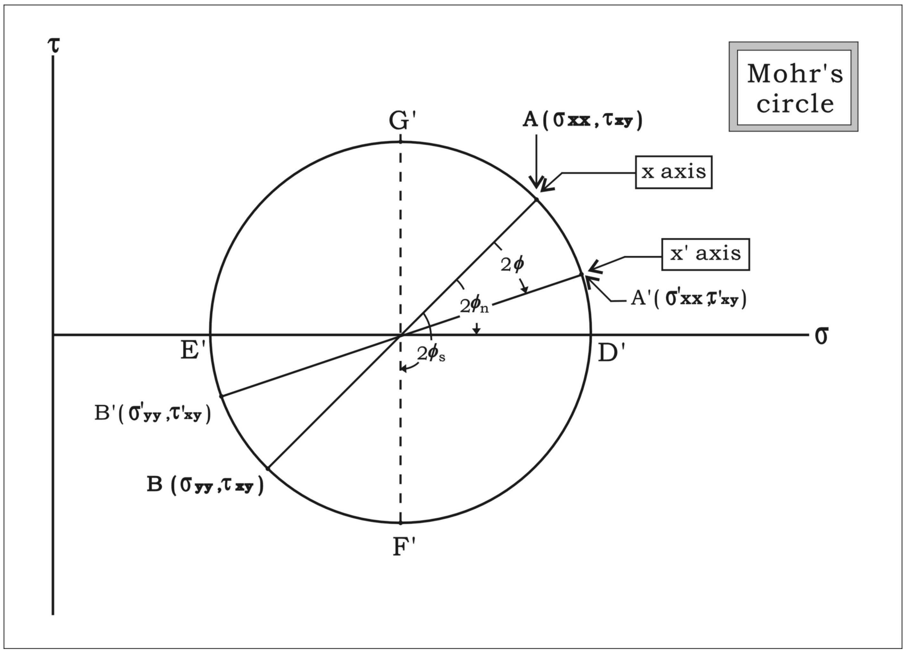
Winkel von ϕ gedreht wird. In diesem Fall sind die normalen Spannungen σ'_{xx} und σ'_{yy} und die Schubspannungen τ'_{xy} und τ'_{yx}.



Das Verhältnis zwischen dem ursprünglichen Spannungszustand (σ_{xx}, σ_{yy}, τ_{xy}, τ_{yx}) und dem Spannungszustand nachdem die Achsen von f (σ'_{xx}, σ'_{yy}, τ'_{xy}, τ'_{yx}) gegen den Uhrzeigersinn gedreht wurden, kann mit der unten gezeigten Abbildung grafisch dargestellt werden.

Für die Darstellung des Mohr'schen Kreises verwenden wir ein Kartesisches Koordinatensystem, in dem die x-Achse den normalen Spannungen (σ) und die y-Achse den Schubspannungen (τ) entspricht. Suchen Sie die Punkte A(σ_{xx},τ_{xy}) und B (σ_{yy}, τ_{xy}), und zeichnen Sie das Segment AB. Die Mitte des Kreises liegt im Punkt C, wo das Segment AB die Achse σ_n überschneidet. Beachten Sie, dass die Koordinaten des Punktes C ($\frac{1}{2}\cdot(\sigma_{yy} + \sigma_{xy})$, 0) sind. Wenn ein Kreis von Hand gezeichnet wird, können Sie diesen mit dem Zirkel zeichnen, da die Position der Kreismitte C und die zwei weiteren Punkte A und B bekannt sind.

Nehmen wir an, dass das AC-Segment die x-Achse im ursprünglichen Spannungszustand darstellt. Wollen Sie den Spannungszustand eines Achsensystems x'-y' ermitteln, welches um einen Winkel von ϕ gegen den Uhrzeigersinn in Bezug auf das ursprüngliche Achsensystem x-y gedreht ist, zeichnen Sie ein Segment A'B', dessen Mitte sich in C befindet und das um einen Winkel von 2ϕ im Uhrzeigersinn in Bezug auf AB gedreht ist. Die Koordinaten von Punkt A′ ergeben die Werte (σ'_{xx},τ'_{xy}), während die von Punkt B′ (σ'_{yy},τ'_{xy}) ergeben.



Der Spannungszustand bei dem die Schubspannung τ'_{xy} gleich Null ist, zu sehen im Segment D′E′, erzeugt die sogenannten *Hauptspannungen*, σ^{P}_{xx} (in Punkt D′) und σ^{P}_{yy} (in Punkt E′). Sie erhalten die Hauptspannungen, wenn Sie das Koordinatensystem x′-y′ um einen Winkel von ϕ_n gegen den Uhrzeigersinn, bezogen auf das Koordinatensystem x-y drehen. Im Mohr'schen Kreis beträgt der Winkel zwischen den Segmenten AC und D′C $2\phi_n$.

Der Spannungszustand, in dem die Schubspannung einen Spitzenwert τ'_{xy} erreicht, wird vom Segment F′G′ bestimmt. Unter diesen Umständen sind beide normalen Spannungen gleich ($\sigma'_{xx} = \sigma'_{yy}$). Dieser Drehung entspricht der Winkel ϕ_s. Der Winkel zwischen den Segmenten AC und F'C in der Abbildung ist $2\phi_s$.

Modulare Programmierung

Um das Programm, welches zum Plotten des Mohr'schen Kreises in einem gegebenen Spannungszustand verwendet wird, zu entwickeln, verwenden wir die modulare Programmierung. Im Prinzip ist dies nichts Anderes als ein Zerlegen des Programms in mehrere Unterprogramme, welche im Taschenrechner als separate Variablen angelegt werden. Diese Unterprogramme werden dann von einem Hauptprogramm zusammengefasst, welches wir *MOHRCIRCL* benennen. Zuerst legen wir im HOME-Verzeichnis ein Unterverzeichnis mit dem Namen MOHRC an und öffnen dann dieses Verzeichnis, um die Programme einzugeben.

Als Nächstes müssen wir im Unterverzeichnis das Hauptprogramm und dessen dazugehörige Unterprogramme anlegen.

Das Hauptprogramm MOHRCIRCL verwendet die folgenden Unterprogramme:

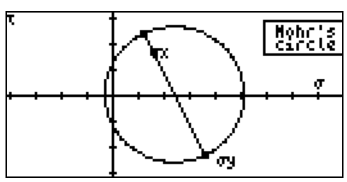
- INDAT : Fordert den Benutzer zur Eingabe von σx, σy, τxy auf und erstellt eine Liste σL = {σx, σy, τxy} als Ausgabe.
- CC&r : Verwendet σL als Eingabe, erzeugt σc = ½(σx+σy), r = Radius des Mohr'schen Kreises, φn = Winkel für Hauptspannungen, als Ausgabe.
- DAXES : Verwendet σc und r als Eingabe, ermittelt die Achsenbereiche und zeichnet die Achsen für die Grafik des Mohr'schen Kreises
- PCIRC : Verwendet σc, r und φn als Eingabe, zeichnet den Mohr'schen Kreis und erstellt einen parametrischer Plot
- DDIAM : Verwendet σL als Eingabe, zeichnet das Segment AB (siehe Abbildung oben mit dem Mohr'schen Kreis), verbindet die Punkte aus den Eingabedaten im Mohr'schen Kreis
- σLBL : Verwendet σL als Eingabe, setzt die Beschriftungen "σx" und "σy" zur Identifizierung der Punkte A und B.
- σAXS : Setzt die Beschriftungen "σ" und "τ" für die jeweiligen Achsen (x bzw. y).
- PTTL : Betitelt die Abbildung mit "Mohr's circle".

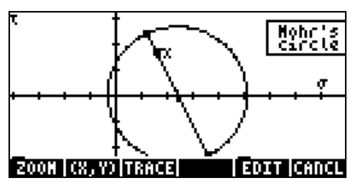
Ausführen des Programms

Wenn Sie die Programme in der oben aufgeführten Reihenfolge eingegeben haben, müssen die folgenden Variablen: PTTL, σAXS, PLPNT, σLBL, PPTS, DDIAM im Unterverzeichnis MOHRC vorhanden sein. Drücken Sie NXT, werden zusätzlich die folgenden Variablen angezeigt: PCIRC, DAXES, ATN2, CC&r, INDAT, MOHRC. Vor Neuordnen der Variablen führen Sie das Programm durch Drücken der Funktionstaste MOHRC erst einmal aus. Verwenden Sie folgende Tastenfolgen:

MOHRC	Startet das Hauptprogramm MOHRCIRCL
2 5 ▼	Geben Sie σx = 25 ein
7 5 ▼	Geben Sie σy = 75 ein
5 0 ENTER	Geben Sie τxy = 50 ein und beenden Sie die Dateneingabe.

An dieser Stelle startet das Programm MOHRCIRCL mit dem Aufrufen der Unterprogramme zum Erstellen der Form. Haben Sie bitte Geduld. Der daraus resultierende Mohr'sche Kreis wird wie auf dem Bild links aussehen.





Da diese Ansicht aus PICT durch die Funktion PVIEW erzeugt wird, erhalten wir keine weiteren Informationen aus dem Plot, außer der Figur selbst. Um weitere Informationen aus dem Mohr'schen Kreis zu erhalten, muss das Programm erst mit ON beendet werden. Drücken Sie anschließend die Taste ◄, um die Inhalte von PICT in der Grafikumgebung wieder herzustellen. Der Mohr'sche Kreis wird diesmal wie auf dem Bild rechts aussehen (siehe oben).

Drücken Sie die Funktionstasten TRACE und (X,Y). Im unteren Bereich des Displays wird der Wert von ϕ, der dem Punkt A(σx, τxy) entspricht, d.h., ϕ = `0 (2.50E1, 5.00E1)`, angezeigt.

Drücken Sie die rechte Pfeiltaste (▶), um den Wert von ϕ zu erhöhen und den entsprechenden Wert von (σ'_{xx}, τ'_{xy}) anzuzeigen. Beispielsweise haben wir für ϕ = 45° die Werte (σ'_{xx}, τ'_{xy}) = (1,00E2; 2,50E1) = (100, 25). Der Wert von σ'_{yy} befindet sich bei einem Winkel von 90° davor, d.h. bei ϕ = 45 + 90 = 135°. Drücken Sie die rechte Pfeiltaste ▶, bis dieser Wert von ϕ erreicht wird, und Sie erhalten das folgende Ergebnis: (σ'_{yy}, τ'_{xy}) = (-1,00E-10;-2,5E1) = (0, 25).

Um die normalen Hauptwerte zu finden, drücken Sie ◀, bis der Cursor wieder an der Überschneidung des Kreises mit dem positiven Bereich der σ-Achse liegt. Für diesen Punkt finden wir die Werte ϕ = 59° und (σ'_{xx}, τ'_{xy}) = (1,06E2;-1;40E0) = (106; -1,40). Jetzt erwarten wir, dass der Wert von τ'_{xy} = 0 im Bereich der Hauptachsen liegt. Was aber geschieht, ist Folgendes: Da wir die Auflösung der unabhängigen Variablen auf $\Delta\phi$ = 1° beschränkt haben, verpassen wir den Punkt bei dem die Schubspannung Null ist. Wird ◀ noch einmal gedrückt, erhalten wir die Werte ϕ = 58° und (σ'_{xx}, τ'_{xy}) = (1,06E2;5,51E-1) = (106; 0,551). Diese Information besagt, dass irgendwo zwischen ϕ = 58° und ϕ = 59°, die Schubspannung τ'_{xy} Null ist.

Drücken Sie ON, um den eigentlichen Wert von ϕn zu ermitteln. Geben Sie danach die entsprechende Liste für die Werte {σx σy τxy}, in unserem Fall `{ 25  75  50 } [ENTER]`, ein.

Drücken Sie anschließend PRNST. Das letzte Ergebnis in der Ausgabe, 58,2825255885°, ist der eigentliche Wert von ϕn.

Ein Programm zum Berechnen von Hauptspannungen

Der oben beschriebene Vorgang für die Berechnung von ϕn kann wie folgt programmiert werden:

Programm PRNST:

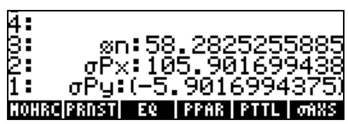
«	Das Programm PRNST (PRiNcipal Stresses – Hauptspannungen) starten
INDAT	Geben Sie die Daten wie beim Programm MOHRCIRC ein
CC&r	Ermitteln Sie σc, r und fn, wie im

	Programm MOHRCIRC
“ϕn” →TAG	Markieren der Winkel für Hauptspannungen
3 ROLLD	Markierten Winkel in Stack-Ebene 3 verschieben
R→C DUP	Konvertieren von σc und r in (σc, r), anschließend duplizieren
C→R + “σPx” →TAG	Hauptspannung σPx berechnen, dann markieren
SWAP C→R - “σPy” →TAG	Spannung σPy austauschen, berechnen, dann markieren.
»	Das Programm PRNST beenden

Um das Programm auszuführen, verwenden Sie:

VAR PRNST	Starten des Programms PRNST
2 5 ▼	Geben Sie σx = 25 ein
7 5 ▼	Geben Sie σy = 75 ein
5 0 ENTER	Geben Sie τxy = 50 ein und beenden Sie die Dateneingabe.

Die Lösung lautet:



Sortieren der Variablen im Unterverzeichnis

Beim ersten Lauf des Programms MOHRCIRCL wurden zwei neue Variablen – PPAR und EQ – erzeugt. Dies ist der zum Plotten des Kreises notwendige Plot-PARameter und die Gleichungsvariablen (EQ). Wir schlagen vor, die Variablen im Unterverzeichnis neu anzuordnen, sodass die Programme MOHRC und PRNST die ersten beiden Beschriftungen der Funktionstasten belegen. Dazu müssen Sie die Liste { MOHRCIRCL PRNST } wie folgt erstellen: VAR ← {} MOHRC PRNST ENTER

Sortieren Sie danach die Liste mit: ← PRG MEM DIR ORDER.

Nach Ausführen des Funktionsaufrufs ORDER drücken Sie [VAR]. Sie werden feststellen, dass nun die Programme MOHRCIRCL und PRNST, wie erwartet, die ersten zwei Variablen im Menü darstellen.

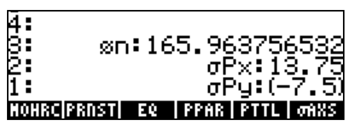
Ein zweites Beispiel zum Berechnen des Mohr'schen Kreises

Ermitteln Sie die Hauptspannung für den Spannungszustand, der durch σ_{xx} = 12.5 kPa, σ_{yy} = -6,25 kPa und τ_{xy} = - 5,0 kPa definiert ist. Zeichnen Sie den Mohr'schen Kreis und ermitteln Sie aus der Figur die Werte von σ'_{xx}, σ'_{yy} und τ'_{xy} bei einem Winkel von $\phi = 35°$.

Zum Ermitteln der Hauptspannungen können wir das Programm [PRNST], wie nachfolgend gezeigt, anwenden:

[VAR] [PRNST]	Starten des Programms PRNST
[1][2][.][5][▼]	Geben Sie σx = 12,5 ein
[6][.][2][5][+/-][▼]	Geben Sie σy = -6,25 ein
5[+/-][ENTER]	Geben Sie τxy = -5 ein und beenden Sie die Dateneingabe.

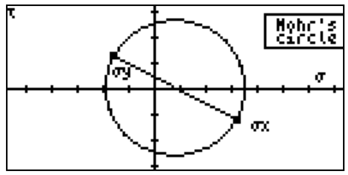
Die Lösung lautet:



Um den Mohr'schen Kreis zu zeichnen, verwenden Sie das Programm [MOHRC], wie folgt:

[VAR] [MOHRC]	Starten des Programms PRNST
[1][2][.][5][▼]	Geben Sie σx = 12.5 ein
[6][.][2][5][+/-][▼]	Geben Sie σy = -6.25 ein
5[+/-][ENTER]	Geben Sie τxy = -5 ein und beenden Sie die Dateneingabe.

Die Lösung lautet:



Beim Ermitteln der Spannungswerte, die einer Drehung von 35° des Winkels der gespannten Partikel entsprechen, gehen Sie wie folgt vor:

ON ◄	Bildschirm löschen, PICT auf dem Grafikbildschirm anzeigen
TRACE (X,Y).	Den Cursor über den Kreis bewegen, wo ϕ und (x,y) angezeigt werden.

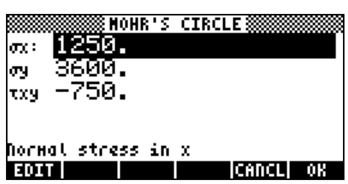
Drücken Sie zunächst die rechte Pfeiltaste ► bis Sie ϕ = 35 erreichen. Die entsprechenden Koordinaten sind (1,63E0; -1,05E1), d.h. bei ϕ = 35°, σ'_{xx} = 1,63 kPa und σ'_{yy} = -10,5kPa.

Eingabemaske des Programms für den Mohr'schen Kreis

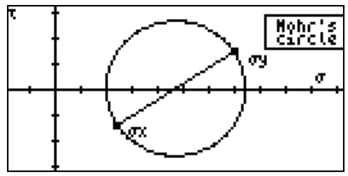
Eine weitere Möglichkeit die Daten einzugeben, bietet sich durch Ersetzen des Unterprogramms INDAT durch folgendes Programm, welches eine Eingabemaske aktiviert.

```
« "MOHR'S CIRCLE" { { "σx:" "Normal stress in x" 0 }
{ "σy:" "Normal stress in y" 0 } { "τxy:" "Shear stress"
0} }   { } { 1 1 1 } { 1 1 1 }  INFORM DROP »
```

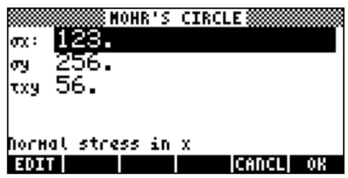
Starten Sie nun dieses Ersatzprogramm, wird beim Ausführen von MOHRC die unten abgebildete Eingabemaske erzeugt:



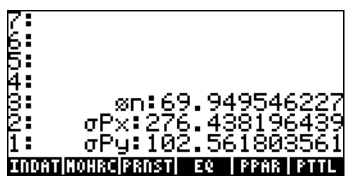
Drücken Sie OK, um den Lauf des Programms fortzusetzen: Das Ergebnis ist die folgende Figur:



Da das Programm INDAT auch für das Programm PRNST (PRiNcipal STresses) verwendet wird, wird beim Ausführen dieses Programms auch eine Eingabemaske verwendet, wie z.B.:



Nach Drücken von OK erhalten Sie das folgende Ergebnis:

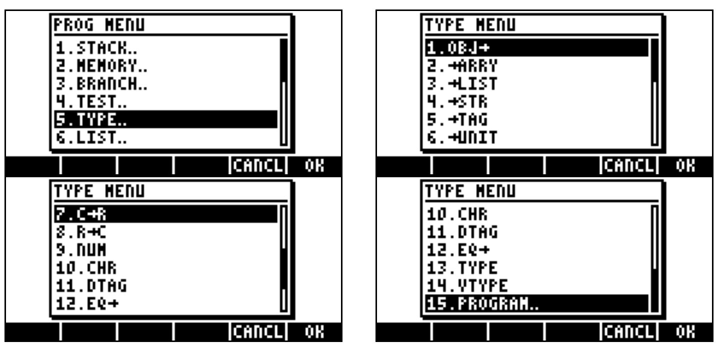


Kapitel 23
Zeichenketten

Zeichenketten sind zwischen Anführungszeichen eingeschlossene Objekte des Taschenrechners. Der Taschenrechner behandelt diese als Texte. So kann beispielsweise die Zeichenkette "SINE FUNCTION" in ein GROB (Grafikobjekt) zum Benennen einer Grafik umgewandelt werden, oder als Ausgabe eines Programms verwendet werden. Jede Folge von Zeichen, die ein Anwender als Eingabe für ein Programm eintippt, wird als Zeichenkette behandelt. Zusätzlich sind auch viele Programmausgaben Zeichenketten.

Funktionen im Untermenü TYPE basierend auf Zeichenketten

Das Untermenü TYPE kann über das Menü PRG (Programmierung) aufgerufen werden, ← PRG . Im Menü TYPE stehen folgende Funktionen zur Verfügung:

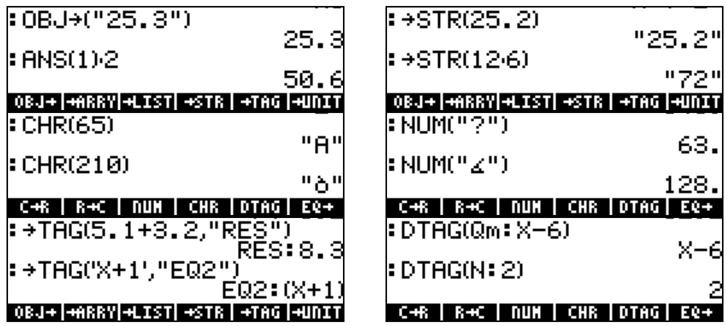


Zu den Funktionen im TYPE-Menü, die dazu dienen, Strings zu bearbeiten, gehören:

OBJ→ : Konvertiert einen String in das Objekt, das es darstellt
→STR : Konvertiert ein Objekt in seinen entsprechenden String
→TAG : Markiert eine Menge
DTAG : Löscht die Markierung von einer markierten Menge (de-tag)
CHR : Erzeugt einen String mit einem Zeichen entsprechend der als Argument verwendeten Zahl.

NUM : Gibt den Code für erste Zeichen in einem String zurück

Nachfolgend einige Anwendungsbeispiele dieser Funktionen.



Verknüpfen von Zeichenketten

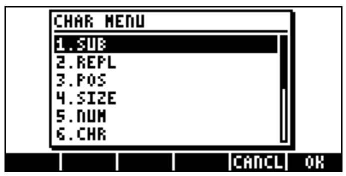
Strings können mit Hilfe des + Zeichens verknüpft (zusammen geführt) werden, beispielsweise:

Die Verknüpfung von Strings ist ein praktischer Weg, Ausgaben eines Programms zu erzeugen. Verknüpfen Sie beispielsweise "YOU ARE " AGE + " YEAR OLD", wird, wenn in der Variablen AGE 25 gespeichert ist, als "YOU ARE 25 YEAR OLD" ausgegeben.

Das Menü CHARS

Das Untermenü CHARS kann über das Menü PRG (Programmierung) aufgerufen werden, ← PRG .

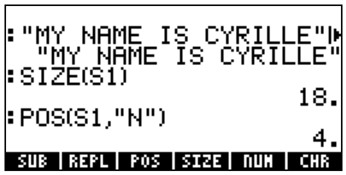
Im Menü CHARS stehen folgende Funktionen zur Verfügung:

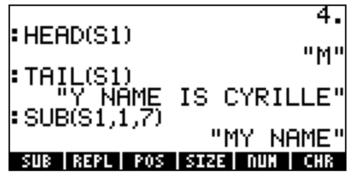


Die Operationen NUM, CHR, OBJ→ und →STR wurden bereits vorgestellt. Außerdem haben wir die Funktionen SUB und REPL in Bezug auf Grafiken kennengelernt. Die Funktionen SUB, REPL, POS, SIZE, HEAD und TAIL haben die folgenden Funktionen:

SIZE : Größe eines Sub-Strings in einer Zeichenkette (einschließlich der Leerstellen)
POS : Position des ersten Vorkommens eines Zeichens in einem String
HEAD : Ermittelt das erste Zeichen eines Strings
TAIL : Entfernt das erste Zeichen eines Strings
SUB : Wählt einen Sub-String mit gegebener Anfangs- und Endposition
REPL : Ersetzt Zeichen eines Strings durch einen Sub-String, beginnend an einer vorgegebenen Position
SREPL : Ersetzt einen Sub-String durch einen anderen Sub-String

Zum besseren Verständnis der Arbeitsweise dieser Funktionen machen Sie folgende Übung: Speichern Sie den String "MY NAME IS CYRILLE" in die Variable S1. Wir verwenden diese Zeichenkette, um zu sehen, wie die Funktionen des Menüs CHARS arbeiten:





Die Zeichenliste

Alle Zeichen, die auf dem Taschenrechner zur Verfügung stehen, können über die Tastenfolge → CHARS erreicht werden. Wenn Sie ein Zeichen hervorheben, beispielsweise das Zeichen Line Feed ↵, sehen Sie unten links auf der Anzeige die Tastenfolge, die dieses Zeichen aufruft (in diesem Fall →.) und den numerischen Code des Zeichens (in diesem Fall 10).

Zeichen, die nicht definiert sind, erscheinen in der Zeichenliste als dunkle Rechtecke (▪) und unten auf der Anzeige wird (None) angezeigt, obwohl für alle ein numerischer Code existiert. Bei numerischen Zeichen wird der entsprechende numerische Code angezeigt.

Buchstaben zeigen den Code α (d.h., ALPHA) gefolgt von dem entsprechenden Buchstaben. Wenn Sie beispielsweise M auswählen, wird unten links αM angezeigt, was bedeutet, dass ALPHA M gedrückt werden muss. Andererseits können Sie m auch über die Tastenkombination α←M oder ALPHA ← M anzeigen.

Griechische Buchstaben wie beispielsweise σ zeigen den Code α→S oder ALPHA → S. Einigen Zeichen wie ρ ist keine Tastenfolge zugeordnet. Daher besteht für diese Zeichen nur die Möglichkeit, das entsprechende Zeichen in der Zeichenliste auszuwählen und dann durch Drücken von ECHO1 oder ECHO zu übertragen.

Drücken Sie ECHO1, wird nur ein Zeichen in den Stack übertragen und der Taschenrechner kehrt sofort in die Normalanzeige zurück. Verwenden Sie hingegen ECHO zum Übertragen mehrerer Zeichen in den Stack. Drücken Sie ON, um zur Normalanzeige zurückzukehren.

Näheres zur Verwendung von Sonderzeichen finden Sie in Anhang D.
Tastenkombinationen, mit denen Sie Sonderzeichen erstellen können, finden Sie in Anhang G.

Kapitel 24
Objekte des Taschenrechners und Flags

Zahlen, Listen, Vektoren, Matrizen, Formeln usw. werden als Objekte des Taschenrechners bezeichnet. Diese werden in 30 verschiedene Typen unterteilt. Eine nähere Beschreibung finden Sie weiter unten in diesem Kapitel. Flags sind Variablen, die zum Steuern von Eigenschaften des Taschenrechners verwendet werden können. Flags wurden bereits in Kapitel 2 näher erläutert.

Beschreibung der Objekte des Taschenrechners

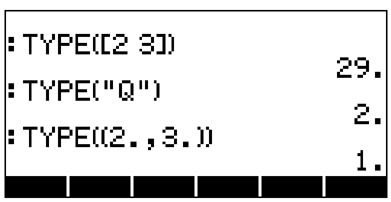
Der Taschenrechner erkennt die folgenden Objekte:

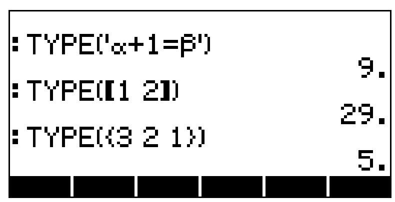
Nummer	Typ	Beispiel
0	Reele Zahl	-1.23E-5
1	Komplexe Zahl	(-1.2,2.3)
2	Zeichenkette	"Hello, world "
3	Array aus reellen Zahlen	[[1 2][3 4]]
4	Array aus komplexen Zahlen	[[(1 2) (3 4)] [(5 6) (7 8)]
5	Liste	{3 1 'PI'}
6	Globaler Name	X
7	Lokaler Name	y
8	Programme	<< → a 'a^2' >>
9	Algebraisches Objekt	'a^2+b^2'
10	Binäre Integer-Zahl	# A2F1E h
11	Grafik-Objekt	Graphic 131×64
12	Markiertes Objekt	R: 43.5
13	Einheiten-Objekt	3_m^2/s
14	XLIB-Name	XLIB 342 8
15	Verzeichnis	DIR Σ END
16	Bibliothek	Library 1230"...
17	Sicherungs-Objekt	Backup MYDIR
18	Integrierte Funktionen	COS
19	Integrierte Befehle	CLEAR

Nummer	Typ	Beispiel
21	Erweiterte reelle Zahl	Long Real
22	Erweiterte komplexe Zahl	Long Complex
23	Verknüpftes Array	Linked Array
24	Zeichen-Objekt	Character
25	Code-Objekt	Code
26	Bibliotheks-Daten	Library Data
27	Externes Objekt	External
28	Integer-Zahl	3423142
29	Externes Objekt	External
30	Externes Objekt	External

Funktion TYPE

Mit dieser Funktion wird der Typ eines Objekts bestimmt und kann entweder über das Menü PRG/TYPE () oder über den Befehlskatalog aufgerufen werden. Als Argument der Funktion wird das gesuchte Objekt eingesetzt. Die Funktion gibt den Typ des Objekts als eine der oben aufgeführten Nummern zurück.





Funktion VTYPE

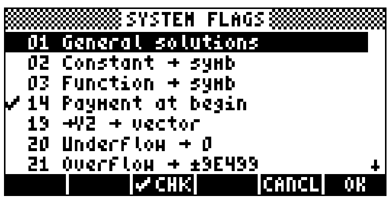
Diese Funktion ist ähnlich wie die Funktion TYPE, bezieht sich aber auf einen Variablennamen. Sie gibt den Typ des in dieser Variablen gespeicherten Objekts zurück.

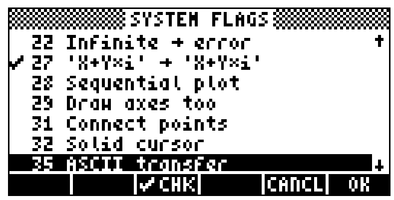
Taschenrechner-Flags

Ein Flag ist eine Variable, die entweder gesetzt oder nicht gesetzt ist. Der Status des Flags beeinflusst das Verhalten des Taschenrechners bei einem Systemflag, einem Programmflag oder einem Anwenderflag. Eine Beschreibung der einzelnen Typen finden Sie nachfolgend.

Systemflags

Auf Systemflags kann über MODE FLAGS zugegriffen werden. Drücken Sie die Nach-Unten-Pfeiltaste, um die Flags mit ihrer Nummer und einer Kurzbeschreibung anzuzeigen. Die ersten beiden Anzeigeseiten der Systemflags sehen sie unten:





Sie werden viele dieser Flags erkennen, da Sie im Menü MODES ein- oder ausgeschaltet werden (z.B. Flag 95 für den algebraischen Modus, 103 für den komplexen Modus usw.). In diesem Handbuch haben wir den Unterschied zwischen CHOOSE boxes und SOFT-Menüs dargestellt. Diese werden durch Setzen oder Abwählen des Systemflags 117 ausgewählt. Ein anderes Beispiel für Systemflags sind die Flags 60 und 61, die sich auf die Konstanten-Bibliothek auswirken (siehe CONLIB in Kapitel 3). Die Flags arbeiten folgendermaßen:

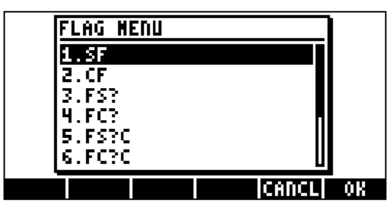
- Anwenderflag 60: löschen (Vorgabe): SI-Einheiten, setzen: ENGL-Einheiten
- Anwenderflag 61: löschen (Vorgabe): Einheiten verwenden, setzen: nur Werte

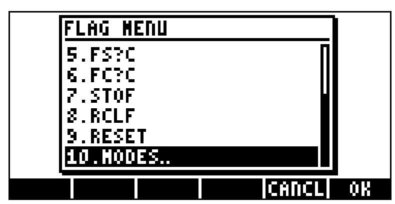
Funktionen zum Setzen und Ändern von Flags

Diese Funktionen können zum Setzen oder Löschen von Anwender- oder Systemflags oder zum Überprüfen des Status derselben verwendet werden. Der Zugriff auf Systemflags erfolgt mit diesen Funktionen über negative Integer-Zahlen. So wird auf das Systemflag 117 über -117 zugegriffen. Für Anwenderflags werden hierzu positive Integer-Zahlen verwendet. Es ist wichtig, dass Anwenderflags nur in der Programmierung, zur Steuerung des Programmflusses, verwendet werden.

Funktionen zum Manipulieren der Flags im Taschenrechner finden Sie im Menü PRG/MODES/FLAG. Das Menü PRG wird über die Tastenfolge PRG aufgerufen. Die folgenden Anzeigefenster zeigen, wie (mit dem Systemflag 117 auf CHOOSE boxes eingestellt) auf das Menü FLAG zugegriffen wird:

Im Menü FLAG sind die folgenden Funktionen enthalten:





Die Funktionen arbeiten wie folgt:

SF Flag setzen
CF Flag löschen
FS? Gibt 1 zurück, wenn das Flag gesetzt ist, 0, wenn es nicht gesetzt ist
FC? Gibt 1 zurück, wenn das Flag leer ist (nicht gesetzt), 0, wenn es gesetzt ist

FS?C   Testet Flag wie FS und löscht es dann
FC?C   Testet Flag wie FC und löscht es dann
STOF   Speichert neue Systemflag-Einstellungen
RCLF   Lädt die bestehenden Flag-Einstellungen
RESET   Setzt die gegenwärtigen Feldwerte zurück (kann zum Zurücksetzen eines Flag verwendet werden)

Anwenderflags

Für die Programmierung stehen dem Anwender die Flags von 1 bis 256 zur Verfügung. Diese beeinflussen die Funktion des Taschenrechners nicht.

Kapitel 25
Datum- und Zeit-Funktionen

In diesem Kapitel stellen wir einige Funktionen und Berechnungen mit Zeiten und Daten vor.

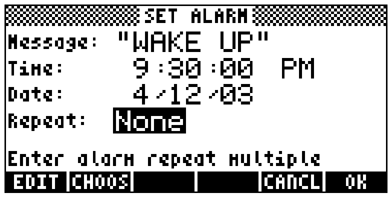
Das Menü TIME

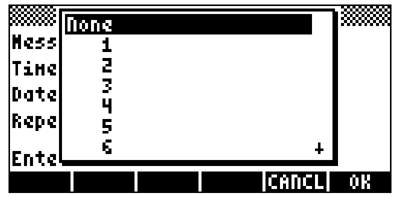
Das Menü TIME wird über die Tastenfolge ↱ TIME (die Taste 9) aufgerufen und bietet die nachfolgend beschriebenen Funktionen:

Alarm einrichten

Die Option *2. Set alarm..* bietet eine Eingabemaske zum Einrichten eines Alarms durch den Anwender. Die Eingabemaske sieht wie folgt aus:

Das Eingabefeld Message: bietet Ihnen die Möglichkeit, einen Text zur Beschreibung des Alarms einzugeben. In das Eingabefeld Time: können Sie die gewünschte Zeit eingeben. Mit dem Feld Datum stellen Sie das Datum für den Alarm ein (oder bei wiederholtem Alarm den Tag für die erste Aktivierung). Sie können beispielsweise den folgenden Alarm einrichten: Die linke Abbildung zeigt einen Alarm ohne Wiederholung. Die rechte Abbildung zeigt, nach Drücken der Taste CHOOS, die Optionen für Wiederholungen. Nach Drücken der Tase OK ist der Alarm eingestellt.





Alarme suchen

Mit der Option *1. Browse alarms…* aus dem Menü TIME können Sie durch Ihre gegenwärtigen Alarme blättern. Haben Sie beispielsweise den obigen Alarm eingegeben, zeigt Ihnen die Option die nachfolgende Anzeige:

Diese Anzeige enthält vier Funktionstasten:

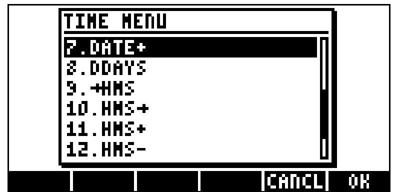
EDIT : Editieren des Alarms über die Eingabemaske
NEW : Eingeben eines neuen Alarms
PURG : Löschen eines Alarms
OK : Rückkehr zur Normalanzeige

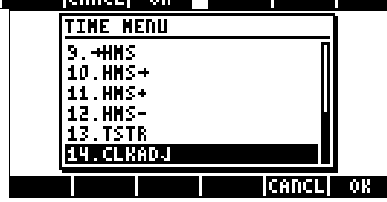
Datum und Uhrzeit einstellen

Die Option *3. Set time, date..* bietet eine Eingabemaske zum Einstellen der Uhrzeit und des Datums. Details hierzu wurden bereits in Kapitel 1 vorgestellt.

TIME-Funktionen

Die Option *4. Tools…* stellt eine Reihe von Funktionen für Operationen mit Uhren und Kalkulationen mit Zeit und Datum zur Verfügung. In der folgenden Abbildung sehen Sie die einzelnen TIME-Funktionen:



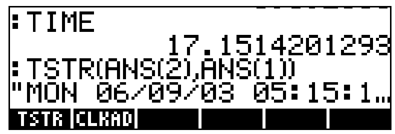


Die Funktionensweise dieser Funktionen werden nachfolgend erläutert:

DATE	:	Stellt das gegenwärtige Datum in den Stack
→DATE	:	Stellt das Systemdatum auf den eingegebenen Wert
TIME	:	Stellt die momentane Zeit im Format HH.MMSS in den Stack.
→TIME	:	Stellt die momentane Zeit im Format HH.MMSS auf den eingegebenen Wert ein.
TICKS	:	Stellt die Systemzeit als binäre Integer-Zahl in Zeit-Ticks mit 1 Tick = 1/8192 Sekunden zur Verfügung
ALRM..	:	Untermenü mit Funktionen zur Manipulation von Alarmen (Beschreibung folgt weiter unten)
DATE+	:	Addiert oder subtrahiert eine Anzahl an Tagen zu einem Datum
DDAYS(x,y)	:	Gibt die Anzahl von Tagen zwischen den Daten x und y zurück
→HMS	:	Konvertiert die Zeit aus einer Dezimalzahl in das Format HH.MMSS
HMS→	:	Konvertiert die Zeit aus dem Format HH.MMSS in eine Dezimalzahl
HMS+	:	Addiert zwei Zeiten im Format HH.MMSS
HMS-	:	Subtrahiert zwei Zeiten im Format HH.MMSS
TSTR(time, date)	:	Konvertiert Zeit, Datum in eine Zeichenkette
CLKADJ(x)	:	Addiert x Ticks zur Systemzeit (1 Tick = 1/8192 s)

Die Funktionen →DATE, →TIME, CLKADJ werden zur Einstellung der Zeit und des Datums verwendet. Für diese Funktionen werden keine Beispiele angeführt.

Nachfolgend finden Sie Beispiele für die Funktionen DATE, TIME und TSTR:

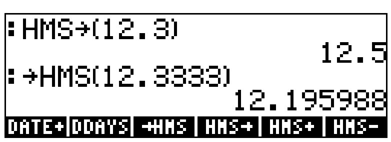


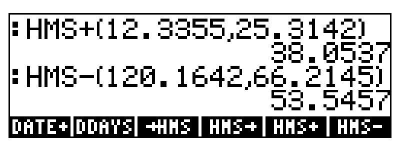
Berechnungen mit Daten

Zur Berechnung mit Daten verwenden Sie die Funktionen DATE+ und DDAYS. Nachfolgend finden Sie ein Beispiel für diese Funktionen zusammen mit einem Beispiel der Funktion TICKS:

Berechnungen mit Zeit

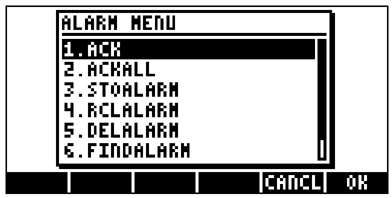
Die Funktionen →HMS, HMS→, HMS+ und HMS- werden zur Manipulation von Werten im Format HH.MMSS verwendet. Hierbei handelt es sich um das gleiche Format wie bei der Berechnung von Winkeln in Grad, Minuten und Sekunden. Daher sind diese Funktionen nicht nur für Zeit-, sondern auch für Winkelberechnungen von Vorteil. Nachfolgend einige Beispiele:





Alarm-Funktionen

In dem Menü TIME/Tools.../ALARM... finden Sie folgende Funktionen:



Die Funktionsweise dieser Funktionen wird als nächstes erläutert:

ACK	: Bestätigt einen überfälligen Alarms
ACKALL	: Bestätigt alle überfälligen Alarme
STOALARM(x)	: Speichert den Alarm (x) in die Systemalarmliste
RCLALARM(x)	: Lädt den Alarm (x) aus der Systemalarmliste
DELALARM(x)	: Löscht den Alarm (x) aus der Systemalarmliste
FINDALARM(x)	: Gibt den ersten fälligen Alarm nach einer angegegeben Zeit zurück

Das Argument x in der Funktion STOALARM besteht aus einer Liste, die ein Referenzdatum (mm.ddyyy), eine Tageszeit im 24-Stunden-Format (hh.mm), eine Zeichenkette mit dem Alarmtext und die Anzahl der Wiederholungen enthält. Beispielsweise `TOALARM({6.092003,18.25,"Test",0}`. Das Argument x in allen anderen Alarm-Funktionen ist eine positive Integer-Zahl, die sich auf den Alarm bezieht, der aufgerufen, gelöscht oder gefunden werden soll.

Da Alarmeinstellungen leicht über das Menü TIME (siehe oben) erfolgen können, beziehen sich die Alarm-Funktionen eher auf die Programmierung des Taschenrechners.

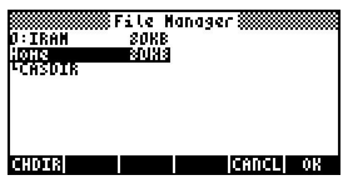
Kapitel 26

Speicherverwaltung

In Kapitel 2 des Benutzerhandbuchs wurden Sie mit dem Erstellen und Verwalten von Variablen und Verzeichnissen vertraut gemacht. In diesem Kapitel werden wir die Speicherverwaltung des Taschenrechners hinsichtlich Speicherpartitionen und die Technik der Datensicherung erörtern.

Speicheraufbau

Der Taschenrechner besitzt insgesamt 80 KB für den Betrieb des Taschenrechners und die Speicherung von Daten (Benutzerspeicher).
Um zu sehen, wie der Arbeitsspeicher partitioniert ist, benutzen Sie die Funktion FILES („ ¡). Nachfolgend sehen Sie ein mögliches Ergebnis:



Auf diesem Bildschirm wird angezeigt, dass ein Speicheranschluss (Anschluss 0) vorhanden ist, der das Verzeichnis HOME umfasst (siehe Kapitel 2 des Benutzerhandbuchs).

Port 0 und das HOME-Verzeichnis teilen sich den gleichen Speicherbereich. Je mehr Daten also im HOME-Verzeichnis gespeichert werden, desto weniger Platz ist für Port 0 verfügbar. Wie bereits erwähnt, beträgt der gesamte Speicher für den Speicherbereich von Anschluss 0/Verzeichnis HOME 80 KB.

Port 0 und dem HOME-Verzeichnis das RAM (Random Access Memory). Das RAM muss ständig über die Batterien mit Strom versorgt werden. Um Datenverlust des RAMs zu verhindern, ist eine Pufferbatterie CR2032 integriert. Sie zusätzliche Details am Ende dieses Kapitels.

Das HOME-Verzeichnis

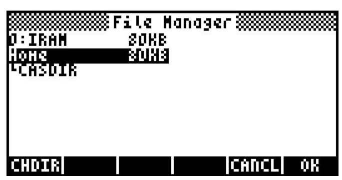
Wenn Sie den Taschenrechner verwenden, erstellen Sie Variablen zum Speichern von Zwischen- und Endergebnissen. Einige Taschenrechnerfunktionen, wie Grafik und Statistik, erzeugen ihre eigenen Variablen zum Speichern von Daten. Diese Variablen werden dann im HOME-Verzeichnis oder einem der Unterverzeichnisse abgelegt. Einzelheiten zum Manipulieren von Variablen und Verzeichnissen können Sie in Kapitel 2 des Benutzerhandbuches nachlesen.

Speicher-Port

Anders als das HOME-Verzeichnis kann der Speicher-Port nicht in Unterverzeichnisse unterteilt werden. Dieser kann nur Sicherungsobjekte oder Bibliotheken aufnehmen. Diese Objektarten werden nachfolgend beschrieben.

Prüfen von Objekten im Speicher

Verwenden Sie die Funktion FILES („ ¡), um zu sehen, welche Objekte im Speicher abgelegt sind. Die Abbildung zeigt das HOME-Verzeichnis, das aus einem Verzeichnis mit dem Namen CASDIR besteht.



Durch Bewegen des Cursors im Verzeichnisbaum nach unten können zusätzliche Verzeichnisse angezeigt werden, oder Sie bewegen diesen nach oben, um einen Speicher-Port auszuwählen. Wurde ein Verzeichnis, ein Unterverzeichnis oder ein Port ausgewählt, drücken Sie OK, um sich den Inhalt des ausgewählten Objekts anzusehen.
Zusätzlich können Sie auf einen Port über das Menü LIB (‚ á , zusammen mit der Taste 2) zugreifen.

Ist eine Bibliothek aktiv, wird sie hier angezeigt. Dieser Speicheranschluss wird Durch Drücken der Softmenütasten für Anschluss 0 geöffnet. Zusätzliche Informationen zu Bibliotheken finden Sie weiter unten.

Sicherungs-Objekte

Sicherungs-Objekte dienen dazu, Daten von Ihrem HOME-Verzeichnis in einen Speicher-Port zu kopieren. Zweck dieser Speicherung ist, den Inhalt dieser Objekte für zukünftige Verwendung zu sichern. Sicherungs-Objekte haben folgende Merkmale:

- Sie können nur im Speicher der Ports existieren (das bedeutet, dass Sie keine Objekte im HOME-Verzeichnis sichern können, obwohl Sie beliebig viele Kopien dieser Objekte erzeugen können)
- Der Inhalt von Sicherungs-Objekten kann nicht modifiziert werden (Sie können jedoch das Objekt zurück in das HOME-Verzeichnis kopieren, dort modifizieren und es dann wieder sichern)
- Sie können ein einzelnes Objekt oder ein ganzes Verzeichnis als einzelnes Sicherungs-Objekt speichern. Sie können jedoch kein einzelnes Sicherungs-Objekt aus mehreren aus einem Verzeichnis ausgewählten Objekten erstellen.

Wenn Sie ein Sicherungs-Objekt erstellen, erzeugt der Taschenrechner eine Prüfsumme (*checksum* oder CRC), welche auf den Binärdaten des Objekts basiert. Dieser Wert wird zusammen mit dem Sicherungs-Objekt gespeichert und dient dem Taschenrechner dazu, die Integrität dieser Objekte zu überwachen. Wenn Sie ein Sicherungs-Objekt in das HOME-Verzeichnis zurückschreiben, errechnet der Taschenrechner erneut die Prüfsumme und vergleicht diese mit dem gespeicherten Wert. Werden Unterschiede festgestellt, warnt Sie der Taschenrechner, dass die zurückgeschriebenen Daten beschädigt sein könnten.

Sichern von Objekten im Port-Speicher

Ähnlich wie das Kopieren von Variablen von einem Unterverzeichnis in ein anderes (siehe Kapitel 2 des Bedienerhandbuchs), werden auch die Sicherungs-Objekte vom Arbeitsspeicher in einen Port-Speicher geschrieben.

So können Sie beispielsweise den Dateimanager (FILES) zum Kopieren und Löschen von Sicherungss-Objekten verwenden, genauso wie Sie diesen für normale Taschenrechner-Objekte einsetzen würden. Wie nachfolgend beschrieben gibt es zusätzlich spezielle Befehle zum Manipulieren von Sicherungs-Objekten.

HOME sichern und neu laden

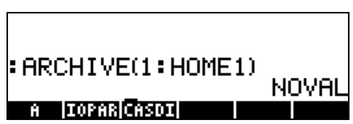
Sie können den gesamten Inhalt des HOME-Verzeichnisses in einem einzigen Sicherungs-Objekt speichern. Dieses Objekt enthält dann alle Variablen, Tastenzuordnungen und Alarme, die zu dem Zeitpunkt im HOME-Verzeichnis definiert waren. Zusätzlich können Sie den Inhalt Ihres HOME-Verzeichnisses aus einem vorher gespeicherten Sicherungs-Objekt wiederherstellen. Nachfolgend wird die Vorgehensweise beschrieben:

Sichern des HOME-Verzeichnisses

Um das momentane HOME-Verzeichnis im algebraischen Modus zu sichern, geben Sie folgenden Befehl ein:

ARCHIVE(:Port_Number: Backup_Name)

Hier ist für Port_Number nur der Wert 0 möglich, und Backup_Name ist der Name des Backup-Objekts, in dem der Inhalt von HOME gespeichert wird. Der _ : Container wird durch die Tastenfolge ::_ eingegeben. Um beispielsweise den Inhalt von HOME in HOME1 im Port 0 zu sichern, verwende Sie:



Um das momentane HOME-Verzeichnis im RPN-Modus zu sichern, geben Sie folgenden Befehl ein:

: Port_Number : Backup_Name ENTER ARCHIVE

Wiederherstellen des HOME Verzeichnisses

Um das momentane home-verzeichnis im algebraischen Modus wiederherzustellen, geben Sie folgenden Befehl ein:

RESTORE(: Port_Number : Backup_Name)

Um das HOME-Verzeichnis aus Ihrem Sicherungs-Objekt HOME 1 wiederherzustellen, geben Sie ein:

`RESTORE(:0:HOME1)`

Im RPN-Modus geben Sie ein:

: Port_Number : Backup_Name ENTER RESTORE

Hinweis: Wenn Sie ein HOME-Verzeichnis wiederherstellen, passieren zwei Dinge:

- Das gesicherte Verzeichnis überschreibt das momentane HOME-Verzeichnis. Alle nicht gesicherten Daten des momentanen HOME-Verzeichnisses gehen verloren.
- Der Taschenrechner startet neu. Die Inhalte der History oder des Stacks gehen verloren.

Speichern, Löschen und Wiederherstellen von Sicherungs-Objekten

Um ein Sicherungs-Objekt zu speichern, wählen Sie eine der folgenden Möglichkeiten:

- Wählen Sie den Dateimanager (← FILES), um das Objekt in den Port zu kopieren. Nach dieser Methode hat das Sicherungs-Objekt den gleichen Namen wie das Original.
- Kopieren Sie das Objekt mit STO in den Port. Um beispielsweise im algebraischen Modus die Variable A in ein Sicherungs-Objekt mit dem Namen AA im Port 0 zu speichern, geben Sie nachstehende Tastenfolge ein:

  A STO▶ ← :: 0 ▶ ALPHA A ALPHA A ENTER

- Um eine Sicherung Ihres HOME-Verzeichnisses zu erstellen, verwenden Sie den Befehl ARCHIVE (siehe oben).

Um ein Sicherungs-Objekt in einem Port zu löschen:

- Verwenden Sie den Dateimanager (↰ FILES) zum Löschen eines Objekts, genau so wie Sie es vom Löschen einer Variablen im HOME-Verzeichnis kennen (siehe Kapitel 2 des Bedienerhandbuches).
- Verwenden Sie den Befehl PURGE wie folgt:
  Im algebraischen Modus: PURGE(: Port_Number : Backup_Name)
  Im RPN-Modus geben Sie ein: Port_Number : Backup_Name PURGE

Um ein Sicherungs-Objekt wiederherzustellen:

- Wählen Sie den Dateimanager (↰ FILES) zum Kopieren des Objekts aus dem Speicher-Port ins HOME-Verzeichnis .
- Nachdem das Sicherungs-Objekt wiederhergestellt wurde, führt der Taschenrechner eine Integritätsprüfung durch Kalkulieren des CRC-Wertes durch. Eine Differenz zwischen der errechneten und der gespeicherten Prüfsumme resultiert in einer Fehlermeldung über beschädigte Daten.

Verwenden von Daten aus Sicherungs-Objekten

Obwohl Sie den Inhalt eines Sicherungs-Objekts nicht direkt modifizieren können, kann dieser Inhalt für Rechenoperationen verwendet werden. Sie können beispielsweise ein Programm, das Sie als Sicherungs-Objekt gespeichert haben, starten oder in einem Sicherungs-Objekt gesicherte Daten für ein Programm verwenden. Um gesicherte Programme oder Daten zu verwenden, verwenden Sie den Dateimanager (↰ FILES) zum Kopieren des Sicherungs-Objekts in die Anzeige. Alternativ können Sie ein gesichertes Programm mit dem Befehl EVAL starten oder gesicherte Daten mit der Funktion RCL wiederherstellen. Gehen Sie wie folgt vor:

- Im algebraischen Modus:
  - Um ein Sicherungs-Objekt auszuwerten, geben Sie ein:
    EVAL Argument(s), : Port_Number : Backup_Name)
  - Um ein Sicherungs-Objekt in die Befehlszeile zu laden, geben Sie ein:
    RCL(: Port_Number : Backup_Name)

- Im RPN-Modus:
    - Um ein Sicherungs-Objekt auszuwerten, geben Sie ein: Argument(s) ENTER : Port_Number : Backup_Name EVAL
    - Um ein Sicherungs-Objekt auszuwerten, geben Sie ein: : Port_Number : Backup_Name ENTER RCL

Verwenden von Bibliotheken

Bibliotheken sind vom Anwender erstellte Programme in binärer Sprache, die in den Taschenrechner geladen werden können und aus einem Unterverzeichnis des HOME-Verzeichnisses aufgerufen werden. Sie können als reguläre Variablen in den Taschenrechner geladen, installiert und an das HOME-Verzeichnis angehängt werden.

Installieren und Anhängen von Bibliotheken

Um eine Bibliothek zu installieren, listen Sie den Bibliotheksinhalt aus dem Stack auf (mit der Funktionstaste ↱ oder der Funktion RCL) und speichern sie diesen entweder in Port 0. Um beispielsweise eine Bibliothek in einen Port zu installieren, geben Sie ein:

- Im algebraischen Modus: STO(Library_variable, port_number)
- Im RPN-Modus: Library_variable ENTER port_number STO▸

Nach der Installation der Bibliothek im Port-Speicher muss die Bibliothek an das HOME-Verzeichnis angehängt werden. Dies kann durch Neustart des Taschenrechners erfolgen (Taschenrechner aus- und wieder einschalten) oder durch gleichzeitiges Drücken der Tasten ON F3 . Jetzt kann die Bibliothek verwendet werden. Um das Menü der Bibliotheksaktivierung anzuzeigen, verwenden Sie das Menü LIB (↱ LIB). Der Bibliotheksname wird nun in dem Menü aufgeführt.

Bibliotheksnummern

Wenn Sie das Menü LIB (↱ LIB) verwenden und die Funktionstaste von Port 0 drücken, sehen Sie eine Auflistung der Bibliotheksnummern. Jeder Bibliothek ist eine vierstellige Nummer zugeordnet. Diese Nummern werden von der Funktion, die diese Bibliotheken erstellt hat, zugeordnet und beim Löschen einer Bibliothek verwendet.

Bibliothek löschen

Um eine Bibliothek in einem Port zu löschen, geben Sie folgendes ein:

- Im algebraischen Modus: PURGE(:port_number: lib_number)
- Im RPN-Modus: port_number : lib_number PURGE

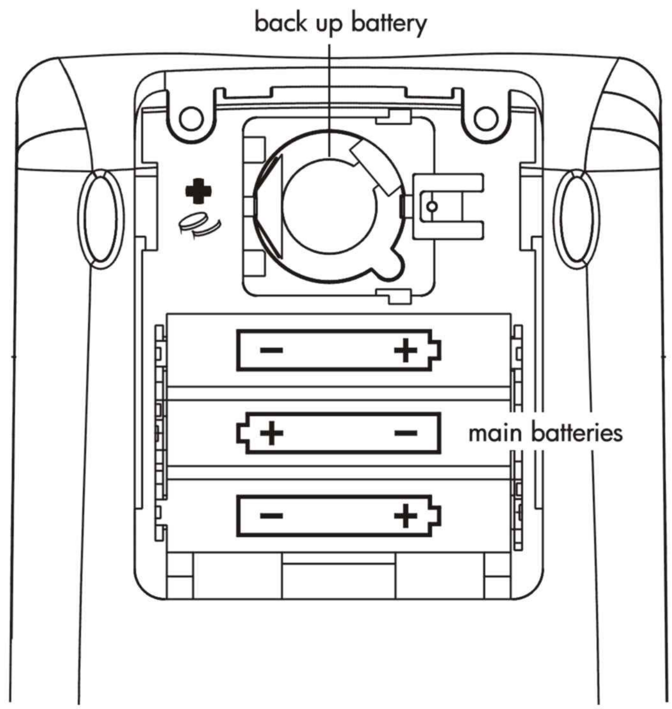
lib_number ist die oben beschriebene Bibliotheksnummer.

Bibliothek erstellen

Bibliotheken können in den Programmiersprachen Assembler und System RPL oder unter Verwendung einer Matrixerstellungsbibliothek, z. B. LBMKR, erstellt werden. Das letzte Programm ist im Internet verfügbar (siehe beispielsweise http://www.hpcalc.org). Details zur Programmierung in Assembler oder RPL liegen außerhalb des Umfangs dieses Handbuchs. Weitere Informationen über dieses Gebiet finden Sie im Internet.

Pufferbatterie

Um den flüchtigen Speicher beim Austausch der Batterien vor Datenverlust zu schützen, ist der Taschenrechner mit einer Pufferbatterie CR2032 ausgestattet. Die Batterie sollte alle 5 Jahre ausgetauscht werden. Wenn die Batterie in ihrer Leistung nachlässt, erscheint eine entsprechende Meldung in der Anzeige. Die nachfolgende Abbildung zeigt Ihnen die Position der Batterie oben im Gehäuse auf der Rückseite des Taschenrechners.


back up battery
main batteries

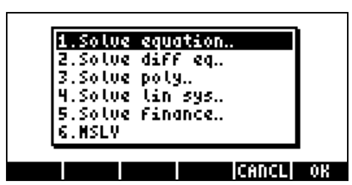
Anhang A
Benutzen von Eingabeformularen

Dieses Beispiel, bei dem Zeit und Datum gesetzt werden, veranschaulicht die Verwendung von Eingabeformularen im Taschenrechner. Einige Grundregeln:

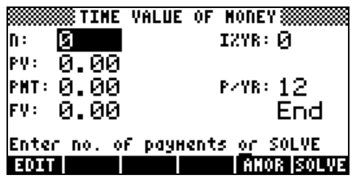
- Benutzen sie die Pfeiltasten (◄►▼▲), um von einem Feld des Eingabeformulars zum nächsten zu springen.
- Drücken Sie eine beliebige der CHOOS Funktionstasten, um die zur Verfügung stehenden Optionen für ein gegebenes Feld des Eingabeformulars zu sehen.
- Benutzen Sie die Pfeiltasten (◄►▼▲), um die gewünschte Option für ein gegebenes Feld auszuwählen, und drücken Sie die Funktionstaste OK (F6) des Soft-Menüs, um die Auswahl zu aktivieren.
- In einigen Fällen muss ein Häkchen zur Auswahl einer Option in einem Eingabeformular gesetzt werden. In diesen Fall wird die Taste ✓CHK des Soft-Menüs zur Aus- und Abwahl des Häkchens verwendet.
- Drücken Sie die Taste CANCL des Soft-Menüs, um ein Eingabeformular zu schließen und zur Stack-Anzeige zurückzukehren. Zum Schließen des Eingabeformulars können Sie auch die Taste ENTER oder ON drücken.

Beispiel – Benutzen von Eingabeformularen im Menü NUM.SLV

Bevor wir diese Themen im Detail besprechen, zeigen wir Ihnen einige der Eigenschaften von Eingabeformularen der finanzmathematischen Anwendung des numerischen Lösers. Starten Sie den numerischen Löser durch Drücken von ↱ NUM.SLV (zusammen mit der 7 Taste). Ein Auswahlfeld mit folgenden Optionen wird angezeigt:



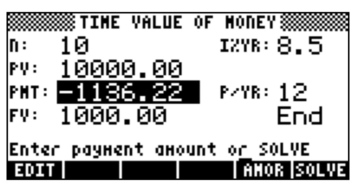
Um mit der finanzmathematischen Berechnung zu beginnen, wählen Sie mit der Pfeiltaste (▼) die Position *5 Solve finance* aus. Drücken Sie die OK-Taste, um die Anwendung zu starten. Es erscheint ein Eingabeformular mit Eingabefelder für eine bestimmte Anzahl von Variablen (n, I%YR, PV, PMT, FV).



In diesem besonderen Fall können wir allen Variablen Werte zuordnen, bis auf eine, also sagen wir, n = 10, I%YR = 8,5, PV = 10000, FV = 1000, und lösen für die Variable PMT (Die Bedeutung dieser Variablen wird später erklärt). Versuchen Sie Folgendes:

10 OK	Geben Sie ein n = 10
8,5 OK	Geben Sie ein I%YR = 8,5
10000 OK	Geben Sie ein PV = 10000
▼ 1000 OK	Geben Sie ein FV = 1000
▲ ◄ SOLVE	Wählen und Lösen Sie PMT

Es erscheint folgende Anzeige:

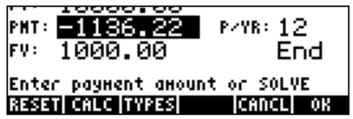


In diesem Eingabeformular werden Sie folgende Beschriftungen bei den Funktionstasten feststellen:

EDIT Zum Bearbeiten des hervorgehobenen Feldes
AMOR Amortisations-Menü – spezielle Option für diese Anwendung

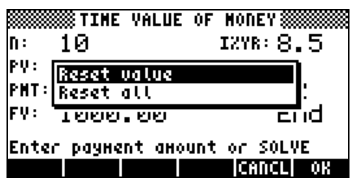
SOLVE Zum Lösen des hervorgehobenen Feldes

Durch Drücken von NXT werden die folgenden Beschriftungen auf den Funktionstasten angezeigt:

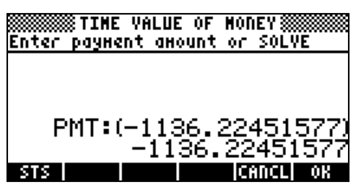


RESET Zurücksetzen aller Felder auf Standardwerte
CALC Zugriff auf den Stack für Berechnungen
TYPES Bestimmung des Objekttyps des hervorgehobenen Feldes
CANCL Operation abbrechen
OK Eingabe annehmen

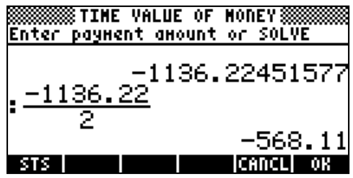
Wenn Sie die Taste RESET drücken, werden Sie aufgefordert, zwischen folgenden beiden Optionen zu wählen:



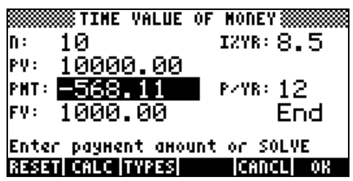
Wenn Sie *Reset value* (Wert zurücksetzen) auswählen, wird lediglich der hervorgehobene Wert auf den Standardwert zurückgesetzt. Wählen Sie statt dessen *Reset all*, werden alle Felder auf ihre Standardwerte zurückgesetzt (normalerweise 0). An dieser Stelle können Sie Ihre Auswahl akzeptieren und die Aktion ausführen (durch Drücken von OK) oder die Operation abbrechen (durch Drücken von CANCL). Drücken Sie nun CANCL. Drücken CALC, um auf den Stack zuzugreifen. Sie sehen folgende Anzeige:



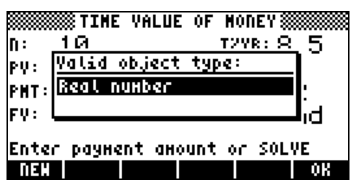
Nun haben Sie Zugriff auf den Stack, und es wird der letzte hervorgehobene Wert des Eingabeformulars angezeigt. Nehmen wir an, Sie möchten diesen Wert halbieren. Nachdem Sie 1136,22/2 eingegeben haben, erscheint folgende Anzeige im ALG-Modus:



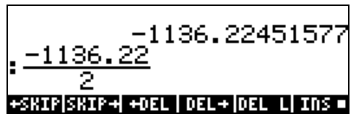
(Im RPN-Modus hätten wir 1136,22 ENTER 2 ENTER ÷ eingeben müssen). Drücken Sie OK, um diesen neuen Wert einzugeben. Das Eingabeformular sieht nun etwa so aus:



Drücken Sie TYPES, damit der Datentyp im Feld PMT angezeigt wird (dies ist das hervorgehobene Feld). Sie erhalten folgende Spezifikation:



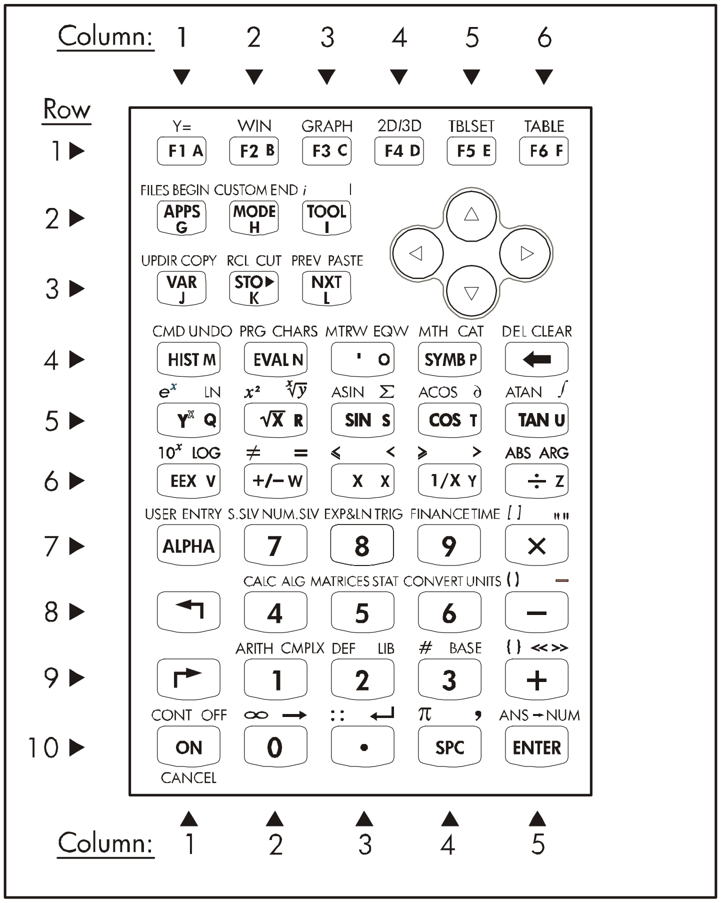
Dies zeigt an, dass der Wert des Feldes PMT eine reelle Zahl sein muss. Drücken Sie OK, um zum Eingabeformular zurückzukehren. und drücken Sie L, um wieder ins erste Menü zu gelangen. Drücken Sie anschließend die Taste ENTER oder die Taste ON, um zum Stack zurückzukehren. Nun werden die folgenden Werte angezeigt:



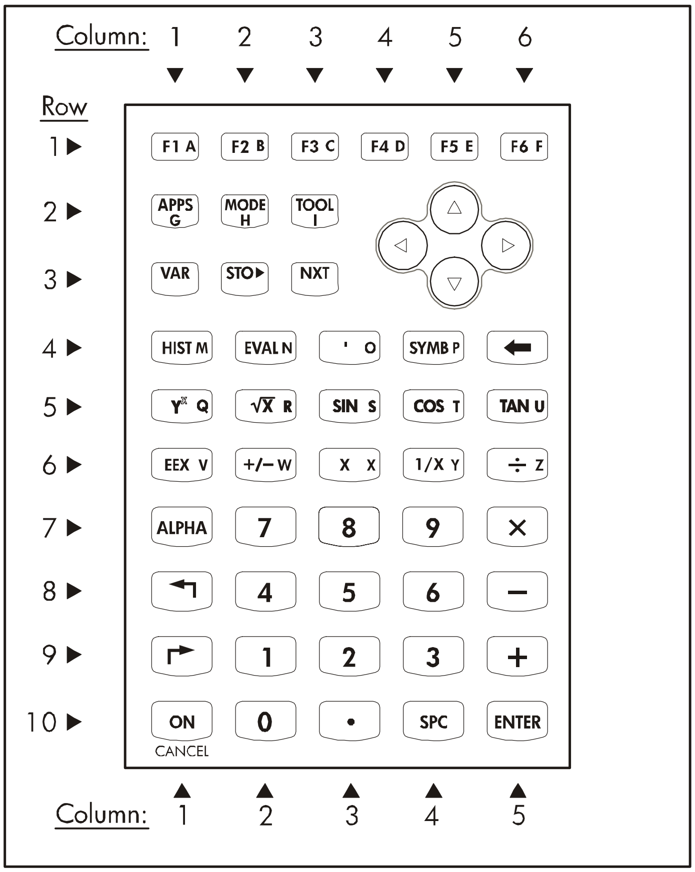
Das obere Ergebnis ist der Wert, der im ersten Teil der Übung für PMT errechnet wurde. Der zweite Wert ist das Ergebnis der Berechnung, die wir durchgeführt haben, um den Wert von PMT neu zu definieren.

Anhang B
Die Tastatur des Taschenrechners

In der nachfolgenden Abbildung sehen Sie die Tastatur Ihres Taschenrechners mit durchnummerierten Zeilen und Spalten.



Diese Abbildung zeigt 10 Reihen von Tasten, in Kombination mit 3, 5 oder 6 Spalten. Reihe 1 hat 6 Tasten, Reihe 2 und 3, jeweils 3, während Reihe 4 bis 10 jeweils 5 Tasten aufweisen. In der rechten oberen Ecke, in Höhe der Reihen 2 und 3, befinden sich 4 Pfeiltasten. Jeder einzelnen Taste sind drei, vier oder fünf verschiedene Funktionen zugeordnet. In der Abbildung unten werden die Hauptfunktionen der Tasten dargestellt. Um die Hauptfunktionen auszuführen, drücken Sie einfach die entsprechende Taste. Wir werden die Tasten je nach Reihe und Spalte in der diese sich gemäß der Abbildung oben befinden, beschreiben, d. h. *Taste (10,1)* ist die Taste *ON* (Einschalttaste).



Hauptfunktionen der Tasten in der Taschenrechnertastatur

Hauptfunktionen der Tasten

Die Tasten F1 bis F6 sind mit Menüfunktionen verbunden, welche am unteren Rand des Taschenrechner-Displays angezeigt werden. Somit können mit diesen Tasten je nach aktivem Menü unterschiedliche Funktionen ausgeführt werden.

- Die Pfeiltasten ▲▼◀▶ werden zur Navigation in Richtung der gedrückten Pfeiltaste verwendet (d. h. nach oben, unten, links und rechts).
- Die Funktion *APPS* startet das Anwendungsmenü.
- Die Funktion *MODE* startet das Menü Modus.
- *TOOL* aktiviert ein Menü mit verschiedenen Werkzeugen zum Bearbeiten von Variablen und für die Hilfestellung.
- Die Funktion *VAR* zeigt die im aktiven Verzeichnis gespeicherten Variablen. *STO* wird zum Speichern von Inhalten in Variablen verwendet.
- *NXT* wird zum Anzeigen weiterer Funktionstasten oder Variablen im Verzeichnis verwendet mit der Funktion HIST können Sie im algebraischen Modus die History starten, d. h. eine Sammlung aktueller Befehle in diesem Modus.
- *EVAL* wird zum Auswerten (Berechnen) algebraischer und numerischer Ausdrücke verwendet. Mit der Taste Apostroph ['] können Sie einen Satz von Apostrophs für algebraische Ausdrücke eingeben.
- *SYMB* aktiviert das symbolische Operationsmenü.
- Die Löschtaste ◀ wird zum Löschen von Zeichen in einer Zeile verwendet.
- Die Taste y^x berechnet die Potenz x der Zahl y.
- $\sqrt{x}$ berechnet die Quadratwurzel einer Zahl.
- Die Tasten *SIN*, *COS* und *TAN* berechnen entsprechend den Sinus, Kosinus oder Tangens einer Zahl.
- *EEX* wird zur Eingabe der Zehnerpotenz einer Zahl verwendet (z. B. 5×10^3 wird als 5 EEX 3 eingegeben und dann als *5E3* angezeigt).
- Die Taste +/- ändert das Vorzeichen eines Eintrags. Mit der Taste *X* geben Sie das Zeichen X in Großbuchstaben ein.
- 1/x berechnet die Inverse einer Zahl. Die Tasten +, −, × und ÷ werden zum Ausführen der Grundrechenarten eingesetzt (entsprechend Addition, Subtraktion, Multiplikation und Division).
- Die *ALPHA* Taste funktioniert in Kombination mit anderen Tasten und wird für die Eingabe alphabetischer Zeichen verwendet.

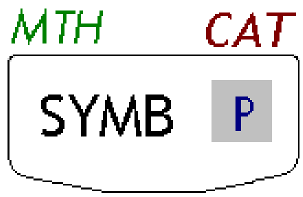
- Die linke- ⇦ bzw. die rechte ⇨ Shift-Taste wird in Kombination mit anderen Tasten zum Aktivieren von Menüs verwendet.
- Der Zahlenblock (Ziffern *0* bis *9*) wird zur Eingabe der Ziffern des Dezimalzahlensystems verwendet.
- Es gibt einen Dezimalpunkt (.) und eine Leertaste (*SPC*).
- Die Taste *ENTER* wird zur Eingabe einer Zahl, eines Ausdrucks oder einer Funktion zum Anzeigen dieser im Stack verwendet.
- Die Taste *ON* dient zum Einschalten des Taschenrechners.

Funktionen in Kombination mit der Alt-Taste

Die linke grüne *Shift-Taste*, *Taste (8,1)*, die rechte rote *Shift-Taste, Taste (9,1)* und die blaue ALPHA-*Taste, Taste (7,1)* können gleichzeitig mit anderen Tasten gedrückt werden, um die auf der Tastatur angezeigten alternativen Funktionen zu aktivieren. So sind beispielsweise der SYMB Taste (*Taste (4,4)*), die folgenden sechs Funktionen wie folgt zugeordnet:

SYMB	Hauptfunktion zum Starten des Menüs SYMB (SYMBolic)
⇦ MTH	in Kombination mit der linken Shift-Taste wird das Menü MTH (Math) gestartet
⇨ CAT	in Kombination mit der rechten Shift-Taste wird die Funktion CATalog (Katalog) gestartet
ALPHA P	Tastenkombination mit ALPHA, dient zur Eingabe des Großbuchstabens P
ALPHA ⇦ P	Kombination von ALPHA- und linke Shift-Taste zum Einfügen des Kleinbuchstabens p
ALPHA ⇨ P	Kombination von ALPHA- und rechte Shift-Taste zum Einfügen des Symbols π

Von den sechs dieser Taste zugeordneten Funktionen werden lediglich die ersten vier auf der Taste selbst angezeigt. So sieht Ihre Tastenanzeige auf der Tastatur aus:



Beachten Sie, dass durch Farbe und Position der Beschriftungen auf der Taste, **SYMB**, *MTH*, *CAT* und **P** bestimmt wird, bei welcher Funktion es sich um die Hauptfunktion (**SYMB**) handelt und welche der drei weiteren Funktionen der jeweiligen Tastenkombination zugeordnet ist: linke Shift-Taste [↰](*MTH*), rechte Shift-Taste [↱] (*CAT*) und [ALPHA] (**P**).

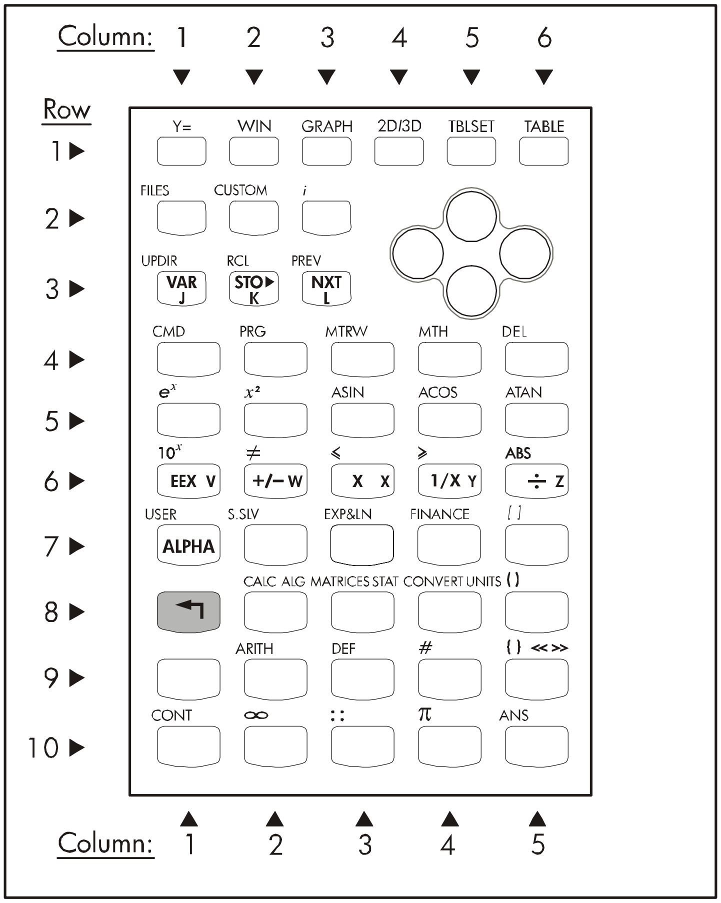
Diagramme zur Veranschaulichung der Funktionen oder Zeichen aus Kombinationen der Taschenrechnertasten mit der linken Shift-Taste [↰], der rechten Shift-Taste [↱], ALPHA [ALPHA], ALPHA + linke Shift-Taste [ALPHA][↰] und ALPHA + rechte Shift-Taste [ALPHA][↱] werden nachfolgend vorgestellt. In diesen Diagrammen werden die für die einzelnen Tastenkombination ausgegebenen Zeichen oder Funktionen auf weißem Hintergrund dargestellt. Bei aktivierter linken oder rechten Shift-Taste oder ALPHA-Taste wird diese auf schattiertem Hintergrund dargestellt. Tasten die nicht aktiviert werden können, sind auf schwarzem Hintergrund dargestellt.

Funktionen in Kombination der linken Shift-Taste

In der nachfolgenden Abbildung sehen Sie Funktionen, die sobald die linke-Shift-Taste, [↰], aktiviert ist, Zeichen oder Menüs, die verschiedenen Tasten des Taschenrechners zugeordnet sind, anzeigen.

- Die sechs Funktionen, die in Kombination mit der linken Shift-Taste, den Funktionstasten [F1] bis [F6] zugewiesen sind, dienen dem Einstellen und Erstellen von Grafiken und Tabellen. Wenn Sie diese Funktionen innerhalb des algebraischen Taschenrechnermodus verwenden, drücken Sie erst die linke Shift-Taste [↰] und anschließend eine der Tasten in Reihe 1. Verwenden Sie diese Funktionen im *RPN*-Modus des Taschenrechners, müssen Sie die linke Shift-Taste [↰]gleichzeitig mit der von Ihnen ausgewählten Taste in der ersten Reihe drücken. Die Funktion *Y=* wird zur Eingabe einer Funktion *y=f(x)* zum Plotten verwendet, die Funktion *WIN* wird zum Einstellen der Parameter für das Plot-Fenster verwendet, *GRAPH* erzeugt einen Graphen, mit der Funktion *2D/3D* wählen Sie den Typ des zu erzeugenden Graphen aus, *TBLSET* wird zum Einstellen der Parameter für eine Wertetabelle einer Funktion verwendet, mit *TABLE* können Sie eine Tabelle mit den Werten einer Funktion erstellen.

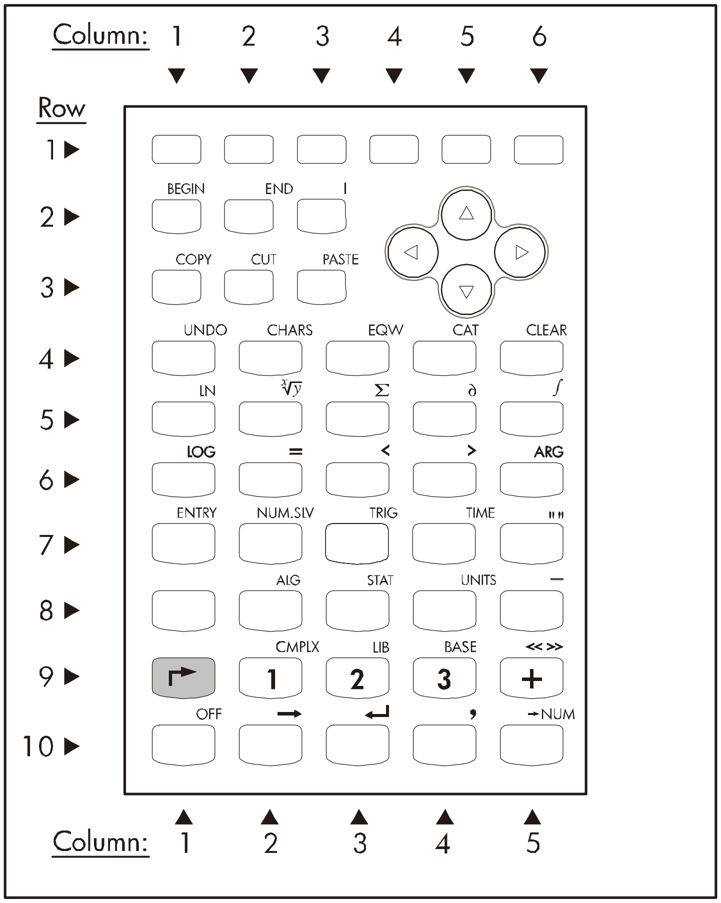
- *FILE* aktiviert den Datei-Browser des Taschenrechners.
- *CUSTOM* startet die Optionen für das allgemeine Menü, das *i* wird zum Eintragen der imaginären Zahl *i* in den Stack verwendet ($i^2 = -1$).
- *UPDIR* bringt Sie in ein übergeordnetes Verzeichnis in der Struktur des Taschenrechners.
- *RCL* wird zur Wiederherstellung von Werten von Variablen verwendet.
- *PREV* zeigt die 6 Tasten der vorhergehenden Menü-Optionen.
- Die Funktion *CMD* zeigt die letzten Befehle, die Funktion *PRG* aktiviert die Programmiermenüs, die Funktion *MTRW* aktiviert den MatrixWriter.



Funktionen in Kombination mit der linken Shift-Taste

- *CMD* zeigt die letzten verwendeten Befehle.
- *PRG* aktiviert die Programmiermenüs.
- Das Menü MTRW startet den MatrixWriter.
- *MTH* aktiviert das Menü mit den mathematischen Funktionen.
- Die Taste *DEL* wird zum Löschen von Variablen verwendet.
- Die Taste e^x berechnet die Exponentialfunktion von x.
- x^2 berechnet das Quadrat von x (wird als Funktion SQ bezeichnet).
- Die Funktionen ASIN, ACOS und ATAN berechnen entsprechend die Funktionen Arcussinus, Arcuskosinus und Arcustangens.

- Die Funktion 10^x berechnet den Antilogarithmus von x.
- Die Tasten $\neq$, $\leq$ und $\geq$ werden zum Vergleichen von reellen Zahlen verwendet.
- Die Funktion *ABS* berechnet den absoluten Wert einer reellen Zahl oder die Magnitude einer komplexen Zahl oder eines Vektors.
- Die Funktion *USER* aktiviert die anwenderdefinierte Menütastatur.
- Die Funktion *S.SLV* aktiviert das symbolische Löser-Menü.
- *EXP&LN* aktiviert das Menü zum Ersetzen von Gliedern in Exponential- und natürlichen Logarithmen.
- *FINANCE* aktiviert eine Menü für finanzmathematische Funktionen.
- Die Funktion *CALC* aktiviert ein Menü für verschiedene Rechenmethoden.
- Die Funktion *MATRICES* aktiviert ein Menü zum Erstellen und Manipulieren von Matrizen.
- Die Funktion *CONVERT* aktiviert das Menü zum Konvertieren von Einheiten und anderer Ausdrücke.
- *ARITH* aktiviert ein Menü mit arithmetischen Funktionen.
- Die Taste *DEF* wird zum Definieren einer einfachen Funktion als Variable im Menü des Taschenrechners verwendet.
- Die Taste *CONT* wird zur Fortsetzung einer Taschenrechneroperation verwendet.
- Die Taste *ANS* stellt den letzten Wert des Taschenrechners wieder her, sofern dieser im algebraischen Modus verwendet wird.
- Die Tasten *[ ]*, *( )* und *{ }* werden zum Einfügen von Klammern verwendet.
- Die # Taste zur Eingabe von Nummern außerhalb der aktiven Zahlenbasis.
- Das Zeichen ∞ wird zur Eingabe des Zeichens Unendlich in einen Ausdruck verwendet.
- Die Taste Pi π wird zur Eingabe des Symbols π verwendet (das Verhältnis der Länge eines Kreisumfangs zu dessen Radius).
- Werden die Pfeiltasten mit der linken Shift-Taste kombiniert, wird der Cursor zum ersten Zeichen in Richtung der gedrückten Taste bewegt.



Funktionen in Kombination mit der rechten Shift-Taste

Funktionen in Kombination mit der rechten Shift-Taste

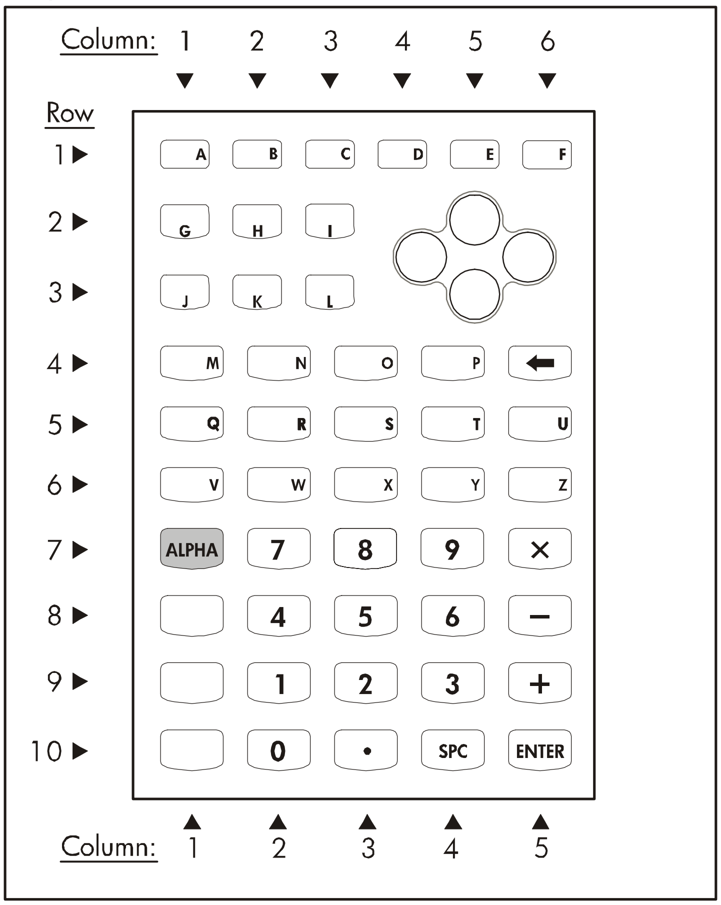
In der obigen Abbildung sehen Sie Funktionen, Zeichen und Menüs, die den Tasten des Taschenrechners zugeordnet sind, wenn die rechte Shift-Taste () aktiviert ist.

- Die Funktionen *BEGIN, END, COPY, CUT* und *PASTE* werden zum Bearbeiten verwendet.
- Die Taste *UNDO* wird zum Rückgängigmachen der letzten Operation im Taschenrechner verwendet.
- Die Funktion *CHARS* aktiviert das Menü für Sonderzeichen.

- Die Funktion *EQW* dient zum Starten des EquationWriters.
- Die Funktion *CAT* wird zum Aufrufen des Katalogs verwendet.
- Die Funktion *CLEAR* löscht die Tastatur.
- Die Funktion *LN* berechnet den natürlichen Logarithmus.
- Die Funktion $\sqrt[x]{y}$ berechnet die x-te Wurzel von y.
- Die Funktion Σ (der große griechische Buchstabe Sigma) dient zum Berechnen von Summen.
- Die Funktion ∂ wird zum Berechnen der Ableitung verwendet.
- Die Funktion ∫ wird zum Berechnen der Integrale verwendet.
- Die Funktion *LOG* berechnet den Zehnerlogarithmus.
- Die Funktion *ARG* berechnet das Argument einer komplexen Zahl.
- Die Funktion *ENTRY* ändert den Eingabemodus in der Bearbeitung.
- Die Funktion *NUM.SLV* startet den numerischen Löser.
- Die Funktion *TRIG* startet die trigonometrischen Austauschfunktionen.
- Die Funktion *TIME* startet das Zeitmenü.
- Die Funktion *ALG* startet das algebraische Menü.
- *STAT* startet das statistische Menü.
- Die Funktion *UNITS* startet das Menü für Maßeinheiten.
- Die Funktion *CMPLX* startet das Menü für komplexe Zahlen.
- Die Funktion *LIB* aktiviert die Bibliotheksfunktionen.
- Die Funktion *BASE* aktiviert das numerische Menü zur Konvertierung von Basen.
- Die Taste OFF schaltet den Taschenrechner aus. Die Taste *→NUM* erzeugt einen numerischen (oder Gleitpunkt-) Wert eines Ausdrucks.
- Mit den Tasten " " können Sie ein Paar von Anführungszeichen eingeben, wird bei der Eingabe von Alpha-Zeichenketten verwendet.
- Die Taste __ fügt einen Unterstrich ein.
- Die << >> Taste fügt das Symbol für ein Programm ein.
- Die Taste → fügt einen Pfeil ein, stellt eine Programmeingabe dar.
- Die Taste ↵ erzeugt eine Zeilenschaltung in Programmen oder Alpha-Zeichenketten.
- Die Taste Komma (,) fügt ein Komma ein.
- Werden die Pfeiltasten mit der rechten Shift-Taste kombiniert, wird der Cursor zu dem am weitesten entfernten Zeichen in Richtung der gedrückten Taste bewegt.

ALPHA Zeichen

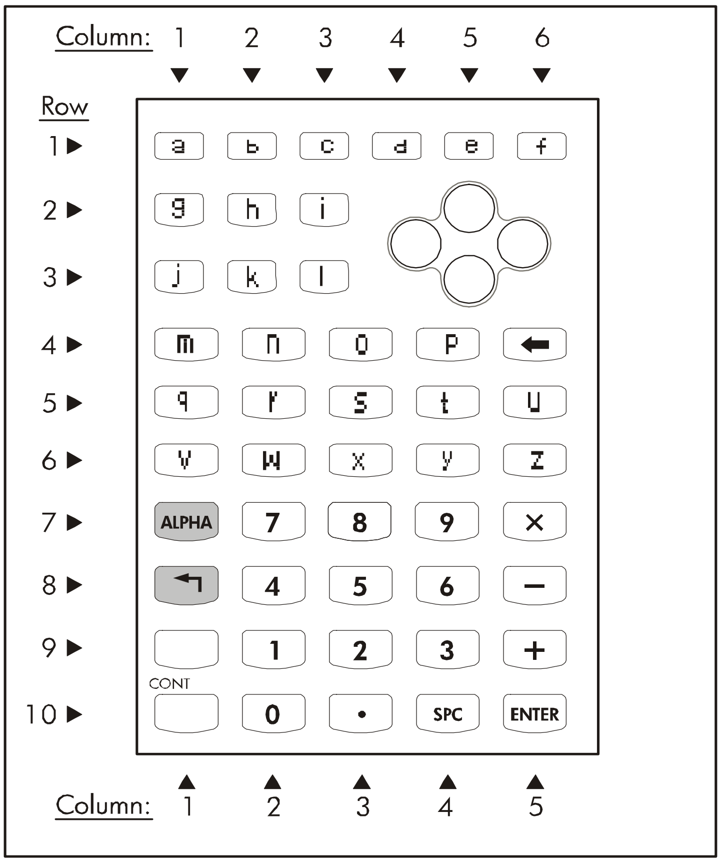
Nachfolgende Abbildung zeigt die Zeichen, die den verschiedenen Taschenrechnertasten zugeordnet sind, wenn ALPHA (ALPHA) aktiviert ist. Beachten Sie, dass die Funktion (ALPHA) hauptsächlich zur Eingabe von Großbuchstaben des englischen Alphabets (von *A* bis *Z*) verwendet werden. Die Zahlen, mathematischen Symbole (-, +), Dezimalpunkt (.) und die Leertaste (*SPC*) entsprechen den Hauptfunktionen dieser Tasten. Die Funktion (ALPHA) erzeugt ein Sternchen (*), wenn diese mit dem Malzeichen kombiniert wird, d. h. (ALPHA)(×).



Funktionen in Kombination mit der Taste Alpha (ALPHA)

Zeichenkombinationen mit Alpha und der linken Shift-Taste

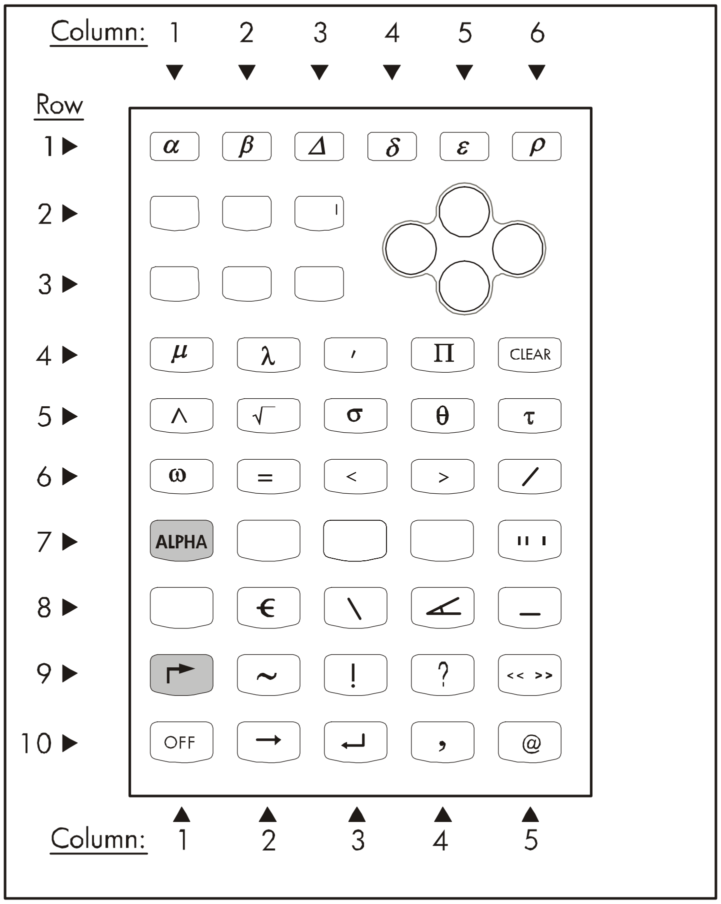
Nachfolgende Abbildung zeigt die Zeichen, welche verschiedenen Taschenrechnertasten zugeordnet sind, wenn die Taste ALPHA [ALPHA] mit der linken Shift-Taste [↰] kombiniert wird. Beachten Sie, dass die [ALPHA][↰]Kombination hauptsächlich zur Eingabe von Kleinbuchstaben des englischen Alphabets (von *A* bis *Z*) verwendet wird. Die Zahlen, mathematischen Symbole (-, *+*, *×*), Dezimalpunkt (.) und die Leertaste (*SPC*) entsprechen den Hauptfunktionen dieser Tasten. Die Tasten ENTER und CONT funktionieren wie deren Hauptfunktion, auch bei Verwenden in Kombination mit den Tasten [ALPHA][↰].



Funktionen von Alpha [ALPHA][↰] in Kombination mit der linken Shift-Taste

Zeichenkombinationen mit Alpha und der rechten Shift-Taste

Nachfolgende Abbildung zeigt die Zeichen, welche verschiedenen Taschenrechnertasten zugeordnet sind, wenn ALPHA (ALPHA) mit der rechten Shift-Taste (↱) kombiniert wird.



Funktionen von Alpha (ALPHA)(↱) in Kombination mit der rechten Shift-Taste

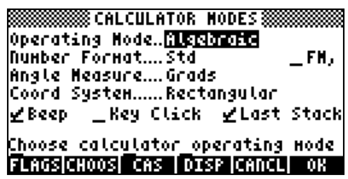
Beachten Sie, dass die Kombination (ALPHA)(↱) hauptsächlich zur Eingabe von Sonderzeichen in den Stack verwendet wird. Die Tasten CLEAR, OFF, →, ↵, Komma (,), sowie die Taste OFF arbeiten in ihrer ursprünglicher Funktion, auch dann wenn die Kombination (ALPHA)(↱) verwendet wird. Die

Sonderzeichen, die Sie mit der Kombination ALPHA → eingeben können, sind griechische Buchstaben (α, β, Δ, δ, ε, ρ, μ, λ, σ, θ, τ, ω, und Π). Weitere Sonderzeichen, die Sie mit der Kombination ALPHA → eingeben können, sind |, ', ^, =, <, >, /, ", \, __, ~, !, ?, <<>> und @.

Anhang C
CAS-Einstellungen

CAS steht für Computer-Algebra-System. Dies ist das mathematische Herzstück des Taschenrechners, in dem die symbolischen mathematischen Operationen und Funktionen programmiert sind. Das CAS-Modul bietet eine Reihe von Einstellungen, die nach Typ oder Operationsart eingestellt werden können. Um die möglichen CAS-Einstellungen anzusehen, gehen Sie wie folgt vor:

- Drücken Sie die Schaltfläche MODE zum Starten der Eingabemaske CALCULATOR MODES.



Am unteren Teil des Displays finden Sie nachfolgende Funktionstasten:
Optionen:

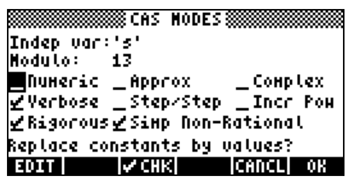
- FLAGS Stellt ein Menü zum Manipulieren von Flags (*) im Taschenrechner zur Verfügung.
- CHOOS Ermöglicht dem Anwender, die verschiedenen Felder im Formular auszuwählen.
- CAS Stellt eine Eingabemaske zum Ändern der CAS-Einstellungen bereit.
- DISP Stellt eine Eingabemaske zum Ändern der Display-Einstellungen bereit.
- CANCL Schließt die aktuelle Eingabemaske und wechselt zur Normalansicht.
- OK Verwenden Sie diese Taste zum Bestätigen Ihrer Einstellungen.

(*)-Flags sind Variablen im Taschenrechner, welche über Zahlen angesprochen entweder "gesetzt" oder "nicht gesetzt" sein können und bestimmte Anwendungsoptionen des Taschenrechners ändern.

Drücken Sie die Taste NXT, erhalten Sie eine Anzeige der noch verbleibenden Optionen in der Eingabemaske CALCULATOR MODES:

RESET Der Anwender kann eine hervorgehobene Option rückgängig machen.
CANCL Schließt die aktuelle Eingabemaske und wechselt zur Normalansicht.
OK Verwenden Sie diese Taste zum Bestätigen Ihrer Einstellungen.

- Um zum ursprünglichen Menü der Eingabemaske CALCULATOR MODES zurückzukehren, drücken Sie die Taste NXT. Interessant für uns an dieser Stelle sind die Änderungen der CAS-Einstellungen. Diese erreichen Sie Drücken der Funktionstaste CAS. Die voreingestellten Werte der CAS-Einstellungen werden nachstehend aufgezeigt:



- Benutzen Sie die Pfeiltasten (◀ ▶ ▼ ▲), um zwischen den einzelnen Optionen der Eingabemaske CAS MODES zu navigieren: .

- Um irgendeine der obigen Einstellungen aus- oder abzuwählen, wählen Sie zuerst den Unterstrich vor der von Ihnen gewünschten Option und bewegen Sie die Funktionstaste ✓CHK solange, bis die gewünschte Einstellung erreicht ist. Sobald eine Option gewählt wurde, erscheint ein Häkchen über dem Unterstrich (d. h. im obigen Beispiel die Option *Rigorous* und *Simp Non-Rational*). Nicht ausgewählte Optionen enthalten kein Häkchen über dem Unterstrich vor der Option (im obigen Beispiel sind das die Optionen *_Numeric, _Approx, _Complex, _Verbose, _Step/Step, _Incr Pow*).

- Nachdem Sie nun die gewünschten Optionen für die Eingabemaske des CAS MODES ausgewählt haben, drücken Sie die Funktionstaste OK.

Damit kehren Sie zur Eingabemaske CALCULATOR MODES zurück. Um zur Normalansicht des Taschenrechners zurückzukehren, drücken Sie erneut die Taste OK.

Auswählen der unabhängigen Variablen

Viele der vom CAS-Modul zur Verfügung gestellten Funktionen verwenden eine vordefinierte unabhängige Variable. Standardmäßig wird eine solche Variable mit dem Großbuchstaben *X*, wie in der nachfolgenden Eingabemaske CAS MODES angezeigt, ausgewählt. Der Anwender kann diesen Buchstaben, jedoch mit jedem anderen Buchstaben oder einer Zahlen- und Buchstaben-Kombination (eine Variable muss mit einem Buchstaben beginnen) ersetzen, indem er das Feld *Indep var* (unabhängige Variable) in der Eingabemaske CAS MODES ändert.

Im Verzeichnis {HOME CASDIR} existiert eine Variable mit dem Namen VX , die standardmäßig den Wert 'X' annimmt. Dies ist der bevorzugte Name für die unabhängige Variable in algebraischen und Infinitesimalrechnungsanwendungen. Deshalb wird für die meisten Übungen dieses Kapitels X als die unbekannte Variable verwendet. Benutzen Sie eine andere unabhängige Variable, z. B. bei der HORNERschen Funktion, kann das CAS-Modul nicht ordnungsgemäß ausgeführt werden.

Die Variable VX hat ihren festen Platz im Verzeichnis {HOME CASDIR}. Weitere CAS Variablen in Ihrem {HOME CASDIR} sind z. B. REALASSUME (REALA), MODULO (MODUL), CASINFO (CASIN) usw.

Sie können den Wert der Variablen VX ändern, indem Sie einen neuen algebraischen Namen in dieser speichern, beispielsweise. 'x', 'y', 'm', usw. Vorzugsweise sollten Sie aber 'X' als Ihre Variable VX für die Beispiele in dieser Anleitung beibehalten.

Vermeiden Sie in Ihren Programmen oder Gleichungen die Variable VX zu verwenden, um diese nicht mit der CAS-Variablen VX zu verwechseln. Wenn Sie sich jedoch auf die x-Komponente der Geschwindigkeit beziehen möchten, können Sie dafür entweder vx oder Vx benutzen.

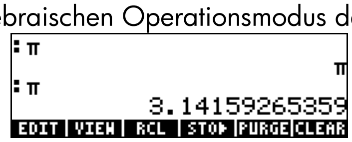
Auswählen von Modulen

Die Option *Modulo* im Eingabefeld CAS MODES stellt eine Zahl (Standardwert = 13) dar, die in der modularen Arithmetik verwendet wird. Weitere Details zur modularen Arithmetik werden an der entsprechenden Stelle in dieser Anleitung beschrieben.

Numerischer vs. symbolischer CAS-Modus

Wird der *Numeric* (numerische) CAS-Modus ausgewählt, werden bestimmte vordefinierte Konstanten des Taschenrechners mit vollständigem Gleitpunktwert angezeigt. Die Option _*Numeric* ist standardmäßig nicht ausgewählt, was bedeutet, dass die vordefinierten Konstanten im Display als Symbol anstelle eines Wertes angezeigt werden.

Die nachfolgende Abbildung zeigt die Werte der Konstanten π (dem Verhältnis zwischen Länge des Kreisumfangs zum Radius) in symbolischem Format, gefolgt von deren numerischem oder Gleitpunkt-Format. Dieses Beispiel entspricht dem algebraischen Operationsmodus des Taschenrechners.



Nachfolgend das gleiche Beispiel im RPN-Modus:

Näherungs- vs. exakter CAS-Modus

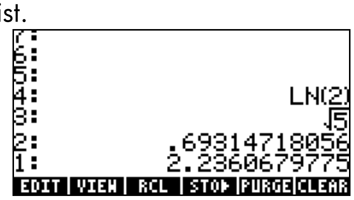
Ist der Modus _*Approx* (Näherungsmodus) ausgewählt, werden symbolische Operationen (beispielsweise. bestimmte Integrale, Quadratwurzeln usw.) numerisch berechnet. Ist der Modus _*Approx* nicht ausgewählt (der exakte Modus ist aktiv), werden symbolische Operationen, wann immer dies möglich ist, als geschlossene algebraische Ausdrücke berechnet.

Nachfolgende Abbildung zeigt eine Reihe von symbolischen Ausdrücken, welche mit aktiviertem algebraischen Modus eingegeben wurden:

Im algebraischen Modus werden die vom Anwender eingegebenen Objekte auf der linken Seite des Displays angezeigt, die Ergebnisse werden auf der rechten Seite des Displays angezeigt. Die obigen Ergebnisse zeigen die symbolischen Ausdrücke für *ln(2)*, den natürlichen Logarithmus von 2, und $\sqrt{5}$, die Quadratwurzel von 5. Ist die Option *_Numeric* (numerische) im CAS-Modul ausgewählt, werden die nachfolgenden Operationen wie folgt angezeigt:

Die zur Eingabe dieser Werte erforderlichen Tastenanschläge im algebraischen Modus lauten ↱ LN 2 ENTER √x 5 ENTER

Die gleichen Berechnungen können auch im RPN-Modus durchgeführt werden. Stack-Ebene *3:* und *4:* zeigen in diesem Fall die exakten CAS-Einstellungen (d. h. die Option *_Numeric* CAS ist nicht ausgewählt), während in Stack-Ebene *1:* und *2:* denselben Fall veranschaulichen, wenn die Option *_Numeric* im CAS-Modul ausgewählt ist.



Die dazu erforderlichen Tastenanschläge lauten 2 ↱ LN 5 √x

Eine Abkürzung auf der Tastatur zum schnellen Wechsel zwischen dem Modus APPROX und EXACT kann durch gleichzeitiges Drücken der rechten Shift-Taste und der Taste ENTER , d. h. ↱ (halten) ENTER, genutzt werden.

Reelle Zahlen vs. Integer-Zahlen

In CAS-Operationen werden Integer-Zahlen verwendet, um die volle Genauigkeit bei Berechnungen beizubehalten. Reelle Zahlen werden als Mantisse mit einem Exponenten gespeichert und haben eine begrenzte Präzision. Im Modus APPROX hingegen werden eingegebene Integer-Zahlen automatisch in die entsprechende reelle Zahl umgewandelt, wie Sie nachfolgend sehen können:

Wenn der Taschenrechner eine Integer-Zahl gefolgt von einem Dezimalpunkt anzeigt, bedeutet dies, dass der Taschenrechner diese in ihre reelle Darstellung umgewandelt hat. Dies bedeutet, dass zum Zeitpunkt der Eingabe der Zahl in den Taschenrechner der Modus APPROX ausgewählt war.

Es wird empfohlen, den Modus EXACT als Standard-CAS-Modus auszuwählen, und erst in den Modus APPROX zu wechseln, wenn der Taschenrechner dies bei der Durchführung einer Operation benötigt.

Zusätzliche Informationen zu reellen und Integer-Zahlen, wie auch anderen Objekten im Taschenrechner, finden Sie in Kapitel 2.

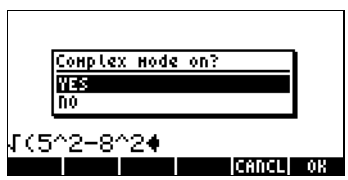
Komplex vs. reeller CAS-Modus

Eine komplexe Zahl ist eine Zahl *a+bi*, wobei *i*, definiert durch $i^2 = -1$, die imaginäre Einheit darstellt (Elektrotechniker ziehen das *j* dafür vor) und *a* und *b* reelle Zahlen sind. So ist beispielsweise die Zahl *2 + 3i* eine komplexe Zahl. Zusätzliche Informationen zu komplexen Zahlen finden Sie in Kapitel 4 dieser Anleitung.

Ist die CAS-Option _*Complex* ausgewählt und das Ergebnis der Berechnung eine komplexe Zahl, dann wird das Ergebnis entweder als *a+bi* oder als geordnetes Paar (*a,b*) angezeigt. Wenn der Modus _*Complex* nicht ausgewählt ist (d. h. der Modus REAL ist aktiv), und das Ergebnis der Berechnung ist eine komplexe Zahl, werden Sie dazu aufgefordert, in den Modus Complex zu wechseln. Bestätigen Sie diesen Wechsel nicht, erhalten Sie eine Fehlermeldung.

Beachten Sie, dass im Modus COMPLEX das CAS-Modul breiter gefächerte Operationen als im Modus REAL durchführen kann, jedoch dementsprechend auch langsamer sein wird. Es wird empfohlen, den Modus REAL als Standard CAS-Modus auszuwählen und nur dann in den Modus COMPLEX zu wechseln, wenn der Taschenrechner dies bei der Durchführung einer Operation benötigt.

Das nachfolgende Beispiel zeigt die Berechnung der Menge $\sqrt{5^2 - 8^2}$ im algebraischen Modus mit ausgewählter Option Real des CAS-Moduls. In diesem Fall werden Sie gefragt, ob Sie in den Modus COMPLEX wechseln möchten:

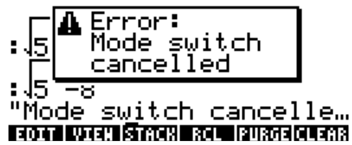


Drücken Sie nun die Funktionstaste OK () drücken, wird die _Complex-Option erzwungen und das Ergebnis sieht wie folgt aus:

Die Tastenanschläge, die oben verwendet wurden, lauten:

√x ↰ () 5 ↰ yˣ 2 + 8 ↰ yˣ 2 ENTER

Benutzen Sie die Taste F6, wenn Sie dazu aufgefordert werden, in den Modus COMPLEX zu wechseln. Möchten Sie den Wechsel in den Modus COMPLEX nicht durchführen, erhalten Sie nachfolgende Fehlermeldung:



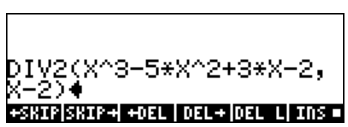
Ausführlicher vs. kurzer CAS-Modus

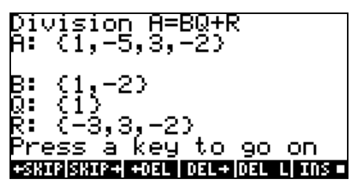
Ist die *_Verbose* (ausführliche) CAS-Option ausgewählt, werden verschiedene Anwendungen mit Kommentaren im Display ausgegeben. Ist diese Option nicht ausgewählt, erhalten Sie keine Kommentare zu jenen Anwendungen. Die Kommentarzeilen erscheinen während der Rechenoperation in den oberen Zeilen des Taschenrechners.

Step-By-Step-CAS-Modus

Ist die Option *_Step/step* (schrittweise) ausgewählt, werden bestimmte Operationen im Einzelschrittmodus auf dem Display angezeigt. Bei nicht ausgewählter Option *_Step/step* werden keine Zwischenberechnungen angezeigt.

Bei ausgewählter Option Step/step werden beispielsweise bei eine schrittweise Division zweier Polynome, und zwar $(X^3\text{-}5X^2\text{+}3X\text{-}2)/(X\text{-}2)$, die nachfolgenden Bildschirme angezeigt. Dies erreicht man über die Funktion DIV2, wie nachfolgend gezeigt. Drücken Sie ENTER, um den ersten Schritt anzuzeigen:

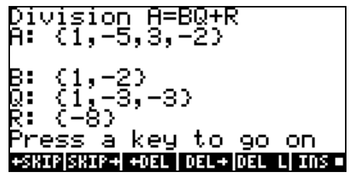


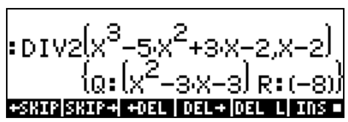


Die Anzeige verrät uns, dass der Taschenrechner eine Division zweier Polynome A/B durchführt, sodass A = BQ + R, wobei Q den Quotienten und R den Restwert darstellt. Für den vorliegenden Fall gilt $A = X^3\text{-}5X^2\text{+}3X\text{-}2$ und B

= X-2. Diese Polynome werden im Display als Auflistung ihrer Koeffizienten dargestellt. So stellt beispielsweise der Ausdruck A: {1,-5,3,-2} das Polynom A = X^3-5X^2+3X-2, B:{1,-2} das Polynom B = X-2, Q: {1} das Polynom Q = X und R:{-3,3,-2} das Polynom R = -3X^2+3X-2 dar.

Drücken Sie an dieser Stelle beispielsweise die Taste ENTER. Drücken Sie ENTER erneut, um weitere Schritte anzuzeigen:





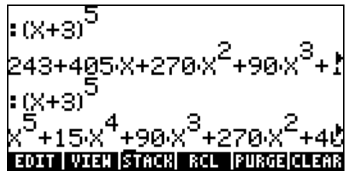
Die Zwischenschritte stellen Koeffizienten der Quotienten und Restwerte der schrittweisen synthetischen Division dar, so als wären diese manuell durchgeführt worden, d. h.:

$$\frac{X^3-5X^2+3X-2}{X-2}=X^2+\frac{-3X^2+3X-2}{X-2}=$$

$$X^2-3X+\frac{-3X-2}{X-2}=X^2-3X-3X-\frac{8}{X-2}.$$

Erhöhen der Potenzen im CAS-Modus

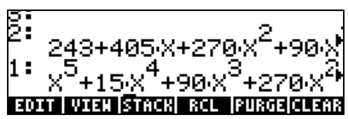
Bei ausgewählter CAS-Option *_Incr pow*, werden die einzelnen Glieder der Polynome in aufsteigender Potenz der unabhängigen Variablen dargestellt. Ist die CAS-Option *_Incr pow* nicht ausgewählt, werden die Glieder der Polynome in absteigender Potenz der unabhängigen Variablen dargestellt. Nachfolgend ein Beispiel im algebraischen Modus:



Im ersten Fall wird das Polynom $(X+3)^5$ in aufsteigender Folge der Potenz von X aufgelistet, während im zweiten Beispiel das Polynom in absteigender Reihenfolge der Potenz von X angezeigt wird. Die für beide Fälle erforderlichen Tastenanschläge sind:

[←] () [X] [+] [3] [▶] [Y^x] [5] [ENTER]

Im ersten Fall war die Option *_Incr pow* ausgewählt, während diese im zweiten Fall nicht ausgewählt war. Nachfolgend das gleiche Beispiel im RPN-Modus:



Für dieses Ergebnis wurde die gleiche Tastenfolge verwendet:

['] [←] () [X] [+] [3] [▶] [Y^x] [5] [ENTER] [EVAL]

Genaue CAS-Einstellung

Steht die Einstellung des CAS-Moduls auf *_Rigorous*, wird der algebraische Ausdruck *|X|*, d. h. der absolute Wert, nicht auf *X* vereinfacht. Ist die Option *_Rigorous* nicht ausgewählt, wird der algebraische Ausdruck *|X|* auf *X* vereinfacht.

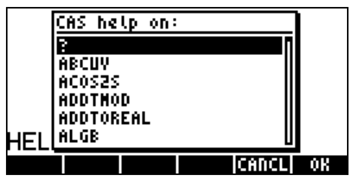
Das CAS-Modul kann bei nicht eingestelltem genauen Modus eine größere Vielzahl von Problemen lösen. Das Ergebnis hingegen oder die Domäne, in welcher die Ergebnisse angewendet werden können, ist möglicherweise jedoch stärker eingeschränkt.

Vereinfachen von nicht rationellen CAS-Einstellungen

Ist die Option *_Simp Non-Rational* ausgewählt, werden nicht-rationelle Ausdrücke automatisch vereinfacht. Ist diese Option nicht ausgewählt, werden nicht-rationelle Ausdrücke nicht automatisch vereinfacht.

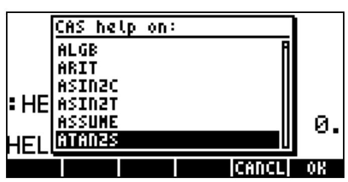
Verwenden der CAS-Hilfefunktion

Schalten Sie den Taschenrechner ein, und drücken Sie die Taste TOOL, um das Menü TOOL zu starten. Zum Aktivieren der Hilfefunktion (HELP) drücken Sie anschließend die Funktionstaste F2, gefolgt von der Taste ENTER (die Taste ganz unten rechts auf der Tastatur des Taschenrechners):



Sie erhalten eine Liste aller CAS-Befehle in alphabetischer Reihenfolge. Verwenden Sie die Pfeiltaste ▼ zur Navigation durch die Liste. Um in der Liste nach oben zu gehen, verwenden Sie die Pfeiltaste ▲. Die Pfeiltasten befinden sich auf der rechten Seite des Displays zwischen der ersten und vierten Zeile in Ihrem Taschenrechner.

Angenommen, Sie suchen Informationen zum Befehl ATAN2S (Arcustangens-zu-Sinus-Funktion). Drücken Sie die Pfeiltaste ▼ so lange, bis der Befehl ATAN2S in der Liste hervorgehoben ist.



Beachten Sie, dass in dieser Instanz nur den Funktionstasten F5 und F6 Befehle zugeordnet sind, nämlich:

CANCL F5 CANCeL zum Abbrechen der Hilfefunktion
OK F6 OK zum Aktivieren der Hilfefunktion für den ausgewählten Befehl

Drücken Sie die Taste CANCL F5 , wird die Funktion HELP (Hilfe) übergangen und der Taschenrechner kehrt zur Normalansicht zurück.

Um die Auswirkung der Taste OK in der Funktion HELP zu sehen, wiederholen wir die oben verwendeten Schritte zur Auswahl des Befehls ATAN2S aus der Liste der CAS-Befehle: HELP F2 ENTER ▼ ▼ …(10 mal)

Um Informationen zum Befehl ATAN2S zu erhalten, drücken Sie die Taste OK F6 .

Die Hilfefunktion zeigt an, dass der Befehl oder die Funktion ATAN2S den Wert von *atan(x)*, der Arcustangenswert eines Wertes *x*, durch dessen Äquivalente als *asin* (Arcussinus) Funktion darstellt, d. h.

Die vierte und fünfte Zeile im Display zeigt ein Anwendungsbeispiel der Funktion ATAN2S. Zeile vier, ATAN2S(ATAN(X)), ist die Operation, die durchgeführt werden soll und Zeile fünf, ASIN(X/√(X^2+1)), stellt das Ergebnis dar.

Die untere Zeile im Display, welche mit dem Wort *See*: (siehe) beginnt, ist ein Verweis auf eine Auflistung weiterer verwandter CAS-Befehle zu ATAN2S im Taschenrechner.

Beachten Sie, dass in diesem Fall den Funktionstasten sechs Befehle zugeordnet sind (Sie können dies mit der Taste NXT überprüfen und werden feststellen, dass keine weiteren Menüeinträge vorhanden sind). Die Befehle der Funktionstasten sind wie folgt:

EXIT F1 Beenden der Hilfefunktion
ECHO F2 Beispielbefehl in den Stack kopieren und verlassen
SEE1 F3 Wechselt zum ersten Link (falls einer existiert) in der Referenzliste.

SEE2	F4	Wechselt zum zweiten Link (falls einer existiert) in der Referenzliste.
SEE3	F5	Wechselt zum dritten Link (falls einer existiert) in der Referenzliste
MAIN	F6	Rückkehr zur Befehlsliste MAIN (Haupt) der Hilfefunktion

In diesem Fall möchten wir das Beispiel durch Drücken der Tasten ECHO F2 in den Stack ECHOen (kopieren). Die daraus resultierende Anzeige sieht wie folgt aus.

Nun zeigen vier Zeilen des Displays jeweils eine Ausgabe an. Die ersten beiden Zeilen von oben entsprechen dem ersten Beispiel aus der Funktion HELP, in der wir die Hilfeanforderung abbrechen. Die dritte Zeile von oben zeigt den aktuellen Aufruf der Funktion HELP, und die letzte Zeile zeigt das ECHO des Beispielbefehls an. Um den Befehl zu aktivieren, drücken Sie die Taste ENTER. Die Lösung lautet:

Beachten Sie, dass, sobald neue Zeilen erzeugt werden, das Display (oder der Stack) die bestehenden Zeilen nach oben verschiebt und den unteren Teil des Displays mit weiteren Ausgabezeilen füllt.

Die in diesem Abschnitt beschriebene Funktion HELP ist bei der Definition von vielen CAS-Befehlen im Taschenrechner vorhanden. Wann immer angemessen beinhaltet jeder Eintrag in der CAS-Hilfefunktion auch ein Beispiel des Befehls, sowie Verweise, wie in obigem Beispiel dargestellt.

Um direkt, ohne die Pfeiltasten zu benutzen, einen bestimmten Befehl in der Hilfefunktion anzusteuern, kann auch nur der erste Buchstabe dieses Befehls eingegeben werden. Angenommen, Sie möchten Informationen zum Befehl IBP (Integration by Parts – Integration durch Teile) erhalten, können Sie nach angezeigter Liste der Hilfefunktionen zunächst die Taste ALPHA (erste Taste in der vierten Reihe von unten) und dann die Taste i (die gleiche Taste wie die TOOL Taste), d. h. ALPHA I drücken. Dies bringt Sie automatisch zu dem ersten Befehl, der mit dem Buchstaben *i* beginnt, und zwar IBASIS. Drücken Sie dann die Pfeiltaste ▽ zweimal, befinden Sie sich bereits auf dem IBP Befehl. Durch Drücken der Tasten OK F6 wird die Hilfefunktion für diesen Befehl gestartet. Ein Drücken der Tasten MAIN F6 bringt Sie zur Befehlsübersicht zurück. Sie verlassen die Hilfefunktion durch Drücken der Tasten EXIT F1 .

Referenzen für nicht-CAS-Befehle

Die Hilfefunktion enthält Einträge für alle für das CAS (Computer Algebraic System – algebraisches System des Taschenrechners) entwickelten Befehle. Es gibt eine große Anzahl anderer Funktionen und Befehle, die ursprünglich für die HP 48G-Reihe entwickelt wurden, die jedoch nicht in der Hilfefunktion enthalten sind. Gute Referenzen für jene Befehle finden Sie in der Referenzanleitung der HP 48g-Reihe (HP Teile Nr. 00048-90126) und der fortgeschrittenen Bedienungsanleitung (HP Teile Nr. 00048-90136), beide 1993 von Hewlett-Packard Company, Corvallis, Oregon veröffentlicht.

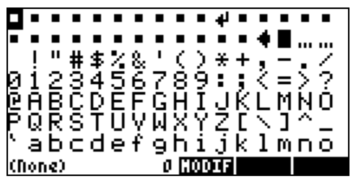
Nutzungsbedingungen für CAS

Die Anwendung von CAS setzt beim Benutzer ein fundiertes mathematisches Wissen voraus. Die CAS-Software wird ohne gesetzliche Garantie geliefert. Wenn nicht anders schriftlich erwähnt, wird die Software vom Copyright-Besitzer ohne Mängelgewähr und ohne Garantie, weder stillschweigender noch zugesagter Art, zur Verfügung gestellt, einschließlich aber nicht beschränkt auf Garantien über Gebrauchstauglichkeit und Eignung für einen bestimmten Zweck. Qualität und Leistung betreffende Risiken der CAS-Software liegen beim Anwender. Sollte die CAS-Software defekt sein, trägt der Anwender alle Kosten für Reparatur oder Fehlerbeseitigung.

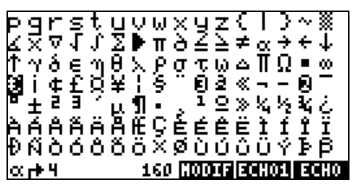
Wenn nicht durch geltendes Recht ausgeschlossen, übernimmt der Copyright-Besitzer keinerlei Haftung für Schäden, einschließlich allgemeiner, spezieller, direkter oder indirekter Schäden, die sich aus der Verwendung der CAS-Software ergeben (einschließlich aber nicht beschränkt auf Datenverlust, falscher Datenverarbeitung oder Verlusten, die der Anwender oder Dritte durch Fehler der CAS-Software in der Zusammenarbeit mit anderen Programmen erleiden), selbst wenn der Anwender oder Dritte auf die Möglichkeit von Schäden hingewiesen wurden. Wenn durch geltendes Recht erforderlich, kann der Schadensersatz, der vom Copyright-Besitzer zu zahlen ist, den Betrag, den Hewlett-Packard dem Copyright-Besitzer für die CAS-Software gezahlt hat, nicht überschreiten.

Anhang D
Zusätzlicher Zeichensatz

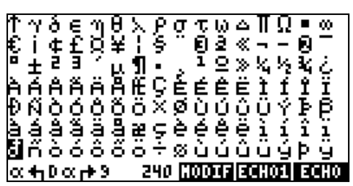
Während Sie alle englischen Groß- und Kleinbuchstaben und alle Ziffern direkt über die Tastatur erreichen. Es gibt jedoch insgesamt 255 verschiedene Zeichen in Ihrem Taschenrechner, beispielsweise Sonderzeichen wie θ, λ, usw., die in algebraischen Ausdrücken verwendet werden. Wir benötigen, um auf diese Sonderzeichen zugreifen zu können, die Tastenkombination CHARS (zusammen mit der Taste EVAL). Auf dem Display wird Folgendes angezeigt:



Mit den Pfeitasten können wir in diesem Zeichensatz navigieren. Eine Bewegung nach unten bewirkt zum Beispiel die Anzeige weiterer Zeichen auf dem Display:



Bewegen wir uns mit den Pfeiltasten weiter nach unten, sehen wir folgende Zeichen:



Eines der Zeichen ist immer hervorgehoben. Die untere Zeile des Displays zeigt das Tastenkürzel des hervorgehobenen Zeichens, sowie den diesem Zeichen entsprechenden ASCII-Code an (beispielsweise ist in der obigen Anzeige das Tastenkürzel α←Dα→9 hervorgehoben, also ALPHA ← D ALPHA → 9, mit ASCII-Code *240*). Das Display zeigt auch drei Funktionen, die den Funktionstasten f4, f5, und f6 zugeordnet sind. Diese Funktionen sind:

MODIF: Öffnet eine grafische Anzeige, mit der der Anwender das hervorgehobene Zeichen verändern kann. Verwenden Sie diese Funktion vorsichtig, denn sie ändert das Zeichen bis zum nächsten Zurücksetzen des Taschenrechners. (Stellen Sie sich den Effekt vor, wenn die grafische Anzeige des Zeichens 1 so abgeändert wird, dass es wie eine 2 aussieht).

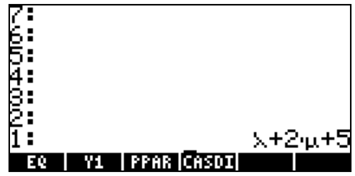
ECHO1: Kopiert das hervorgehobene Zeichen in die Befehlszeile oder den EquationWriter (EQW) und verlässt die Zeichensatz-Anzeige (d. h. ein einzelnes Zeichen wird in den Stack zurückgegeben).

ECHO: Kopiert das hervorgehobene Zeichen in die Befehlszeile oder den EquationWriter (EQW), doch der Cursor verbleibt in der Zeichensatz-Anzeige, um dem Anwender eine Möglichkeit zum Auswählen weiterer Zeichen gibt (d. h., schreibt eine Zeichenfolge in den Stack). Zum Verlassen der Zeichensatz-Anzeige drücken Sie ENTER.

Nehmen wir zum Beispiel an, Sie müssen den Ausdruck $\lambda^2 + 2\mu + 5$ eingeben.

Hier ein Vorschlag zur Vorgehensweise, wobei wir den Stack sowohl im algebraischen als auch im RPN-Modus benutzen können:

Verwenden Sie die Tastenkombination ' → CHARS, um in die Zeichensatz-Anzeige zu gelangen. Benutzen Sie anschließend die Pfeiltasten, um das Zeichen λ.hervorzuheben. Drücken Sie ECHO1 (d. h. die Taste F5), und fahren Sie mit der Tastenkombination + 2 × → CHARS fort. Nun benutzen Sie die Pfeiltasten, um das Zeichen μ.hervorzuheben. Drücken Sie ECHO1 (d. h. die Taste F5), und beenden Sie den Ausdruck mit der Tastenfolge + 5 ENTER. Hier sind die Ergebnisse dieser Übung im ALG- und im RPN Modus:



Nachfolgend eine Auflistung der gebräuchlichsten ALPHA ↱ Tastenkombinationen:

Griechische Buchstaben

α	(Alpha)	ALPHA ↱ A
β	(Beta)	ALPHA ↱ B
δ	(Delta)	ALPHA ↱ D
ε	(Epsilon)	ALPHA ↱ E
θ	(Theta)	ALPHA ↱ T
λ	(Lambda)	ALPHA ↱ N
μ	(Mu)	ALPHA ↱ M
ρ	(Rho)	ALPHA ↱ F
σ	(Sigma)	ALPHA ↱ S
τ	(Tau)	ALPHA ↱ U
ω	(Omega)	ALPHA ↱ V
Δ	(Großbuchstabe Delta)	ALPHA ↱ C
Π	(Großbuchstabe Pi)	ALPHA ↱ P

Andere Zeichen

~	(Tilde)	ALPHA ↱ 1
!	(Faktorielle)	ALPHA ↱ 2
?	(Fragezeichen)	ALPHA ↱ 3
\	(Schrägstrich rückwärts)	ALPHA ↱ 5
∡	(Winkelsymbol)	ALPHA ↱ 6
@	(At-Zeichen)	ALPHA ↱ ENTER

Einige gebräuchliche Zeichen, die nicht über einfache Tastenkombinationen erreicht werden können, sind x (x bar), γ (gamma), η (eta), Ω (Grossbuchstabe Omega). Diese Zeichen müssen aus der Zeichensatz-Anzeige CHARS "geechot" werden: ⇨ CHARS .

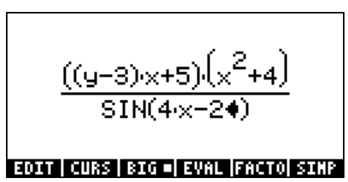
Anhang E
Auswahlbaum im EquationWriter

Der Ausdruck Baum ist ein Diagramm, das anzeigt, auf welche Weise der EquationWriter einen Ausdruck darstellt (interpretiert). Die Form des Ausdrucksbaums hängt von gewissen Regeln ab, bekannt als Operationshierarchie. Die Regeln lauten wie folgt:

1. Operationen in Klammern werden zuerst ausgeführt, beginnend von der innersten bis hin zur äußersten Klammer, und innerhalb des Ausdrucks von links nach rechts.
2. Anschließend werden Argumente von Funktionen, von links nach rechts, ausgeführt.
3. Danach folgen die Funktionen, ebenfalls von links nach rechts.
4. Diesen folgen Zahlenpotenzen, ebenfalls von links nach rechts.
5. Multiplikationen und Divisionen kommen anschließend, auch diese von links nach rechts.
6. Additionen und Subtraktionen werden als letzter Rechenvorgang von links nach rechts ausgeführt.

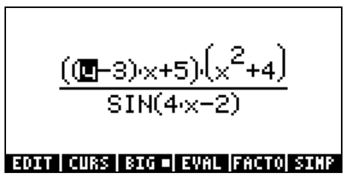
Das Ausführen von links nach rechts bedeutet, wenn zwei Operationen derselben Hierarchie, beispielsweise zwei Multiplikationen, in einem Ausdruck vorhanden sind, wird zunächst die links befindliche Multiplikation vor der zweiten, rechts von dieser befindlichen, ausgeführt, usw.

Nehmen wir als Beispiel den folgenden, im EquationWriter angezeigten Ausdruck:

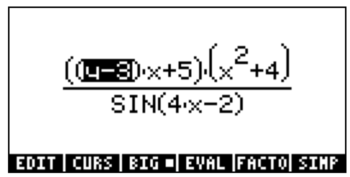


Der Einfügecursor (⬅) an dieser Stelle befindet sich rechts neben der 2 im Argument der SIN-Funktion im Nenner. Drücken Sie die Pfeiltaste ▽, um den reinen Bearbeitungscursor (☐) um die 2 herum, im Nenner zu erhalten. Anschließend drücken Sie die linke Pfeiltaste so lange, bis sich der reine Bearbeitungscursor um das y herum im ersten Faktor des Nenners befindet. Drücken Sie anschließend die Pfeiltaste nach oben, um den Auswahlcursor (■) um das y herum zu erhalten. Drücken Sie nun wiederholt die Pfeiltaste nach oben . Wir können dem Ausdrucksbaum folgen, welcher nun bei y beginnt und bis hin zur Vervollständigung des Ausdrucks folgt. Nachfolgend finden Sie die Abfolge der Operationen, welche von der Pfeiltaste nach oben hervorgehoben wurde:

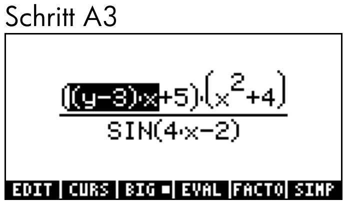
Schritt A1



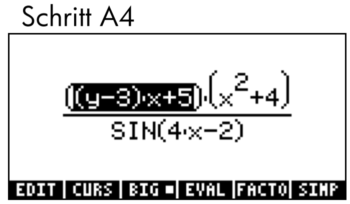
Schritt A2



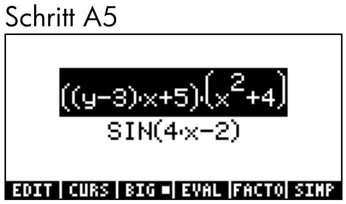
Schritt A3



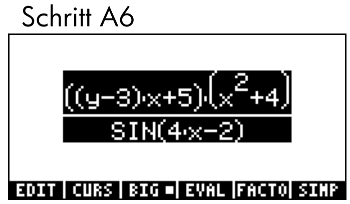
Schritt A4



Schritt A5



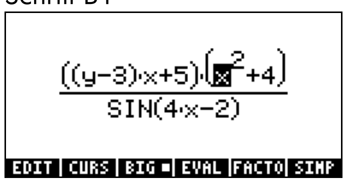
Schritt A6



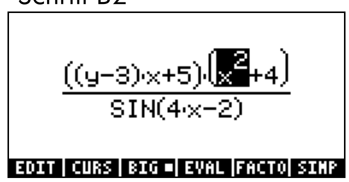
Beachten wir die Anwendung der Operationshierarchie in dieser Auswahl. Als Erstes das y (Schritt A1). Dann, y-3 (Schritt A2, Klammern). Danach (y-3)x (Schritt A3, Multiplikation). Anschließend (y-3)x+5, (Schritt A4, Addition). Gefolgt von, ((y-3)x+5)(x^2+4) (Schritt A5, Multiplikation) und schließlich ((y-3)x+5)(x^2+4)/SIN(4x-2) (Schritt A6, Division). Es ist von besonderer Bedeutung, hervorzuheben, dass die Multiplikation in Schritt A5 das erste

Glied ((y-3)x+5) mit einem zweiten Glied (x^2+4), welches bereits berechnet wurde, beinhaltet. Um die Schritte in der Kalkulation des zweiten Gliedes zu sehen, drücken Sie wiederholt die Taste Pfeil nach unten , bis sich der reine Bearbeitungscursor um das y herum befindet. Drücken Sie anschließend die Pfeiltaste, bis sich dieser Cursor über dem x im zweiten Glied des Nenners befindet. Drücken Sie dann die Taste Pfeil nach Oben, um dieses x auszuwählen. Die Schritte in der Berechnung dieses Ausdrucks, wenn wir von dieser Stelle aus starten, wird wie nachfolgend gezeigt aussehen:

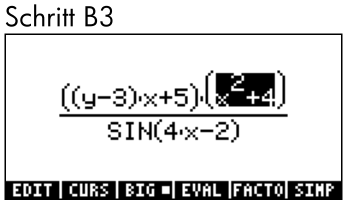
Schritt B1



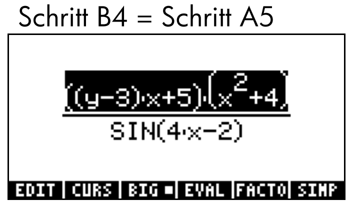
Schritt B2



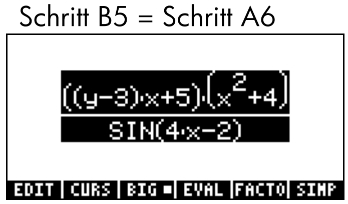
Schritt B3



Schritt B4 = Schritt A5

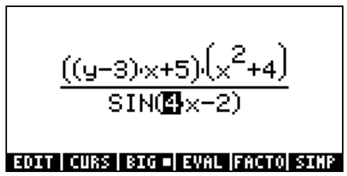


Schritt B5 = Schritt A6

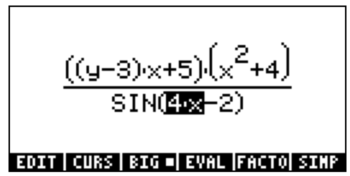


Wir können die Berechnung des Ausdrucks auch beginnend mit der 4 im Argument der SIN Funktion im Nenner starten. Drücken Sie wiederholt die Taste Pfeil nach unten , bis sich der reine Bearbeitungscursor über dem y, ein weiteres Mal befindet. Drücken Sie dann die rechte Pfeiltaste, bis sich der Cursor über der 4 im Nenner befindet. Drücken Sie dann die Taste Pfeil nach oben ▲, um die 4 auszuwählen. Die Schritte in der Berechnung dieses Ausdrucks, wenn wir von dieser Stelle aus starten, wird nachfolgend gezeigt:

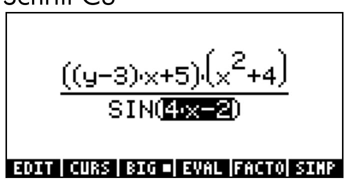
Schritt C1



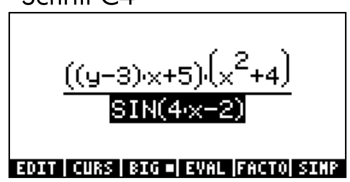
Schritt C2



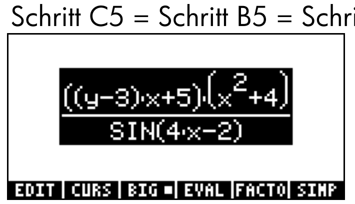
Schritt C3



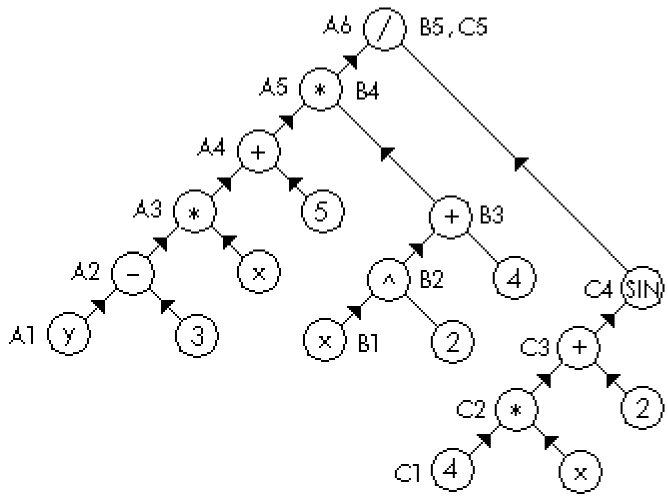
Schritt C4



Schritt C5 = Schritt B5 = Schritt A6



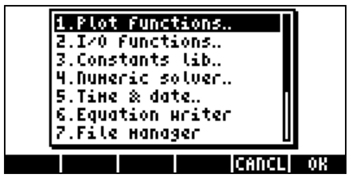
Der Ausdrucksbaum für den oben dargestellten Ausdruck ist nachfolgend dargestellt:

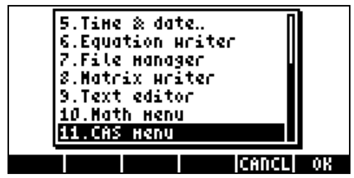


Die Schritte bei der Berechnung der Glieder des Baumes (A1 bis A6, B1 bis B5 und C1 bis C5) werden neben dem Kreis, der Zahlen, Variablen oder Operatoren enthält, angezeigt.

Anhang F
Das Menü (APPS) Anwendungen

Das Menü (APPS) kann über die Taste APPS erreicht werden (erste Taste in der zweiten Reihe von oben). Die Taste APPS zeigt die folgenden Anwendungen:

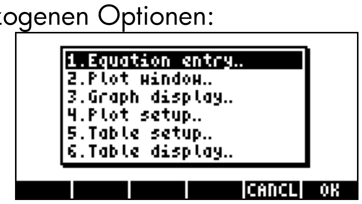




Die verschiedenen Anwendungen werden nachfolgend beschrieben.

Plot-Funktionen

Die Auswahl von Option *1. Plot functions..* im Menü APPS zeigt die folgende Menüliste mit grafikbezogenen Optionen:



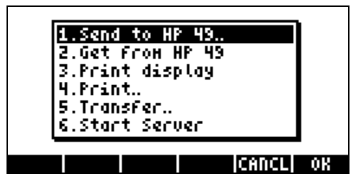
Die sechs gezeigten Optionen entsprechen den unten aufgeführten Tastenkombinationen:

Eingeben der Gleichung…	← Y=	Plot-Fenster…	← WIN
Grafik-Anzeige…	← GRAPH	Plot-Einstellungen…	← 2D/3D
Tabelleneinstellungen…	← TBLSET	Tabellenanzeige…	← TABLE

Diese Anwendungen werden in Kapitel 12 näher erläutert.

I/O functions.. (Ein-/Ausgabe-Funktionen)

Die Auswahl von Option *2. I/O functions..* im Menü APPS zeigt die folgende Menüliste von Ein-/Ausgabe-Funktionen:

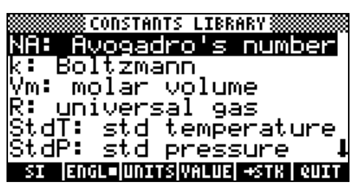


Nachfolgend eine Erläuterung dieser Anwendungen:

Send to HP 49..	Daten an einen anderen Taschenrechner senden
Get from HP 49	Daten von einem anderen Taschenrechner empfangen
Print display	Bildschirm ausdrucken
Print..	Ausgewähltes Objekt drucken
Transfer..	Daten an ein anderes Gerät übertragen
Start Server..	Den Taschenrechner als Server zur Kommunikation mit Computern einrichten

Constants lib.. (Konstantenbibliothek)

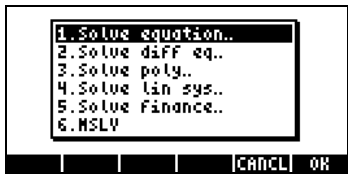
Die Auswahl von Option *3. Constants lib..* im Menü APPS öffnet die Constant Library Application (Konstantenbibliothek), die Werte für physikalische Standard-Konstanten liefert:



Eine genau Beschreibung der Konstantenbibliothek finden Sie in Kapitel 3.

Numeric solver.. (Numerischer Löser)

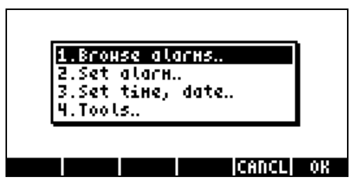
Die Auswahl von Option 3. *Numeric solver..* im Menü APPS liefert das Menü für den numerischen Löser:



Diese Operation entspricht der Tastenkombination NUM.SLV . Weitere Informationen zu dem Menü für den numerischen Löser finden Sie in den Kapiteln 6 und 7.

Time & date.. (Datum & Zeit)

Die Auswahl von Option *5.Time & date..* im Menü APPS startet das Menü für Zeit und Datum:



Diese Operation entspricht der Tastenkombination TIME . Weitere Informationen zu Datum und Zeit finden Sie in Kapitel 26.

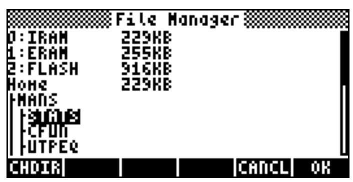
Equation Writer.. (EquationWriter)

Die Auswahl von Option *6.Equation Writer..* im Menü APPS öffnet den EquationWriter:

Diese Operation entspricht der Tastenkombination ↱ EQW . Weitere Informationen zum EquationWriter finden Sie in Kapitel 2. Beispiele zum Anwenden des EquationWriters finden Sie durchgehend in diesem Handbuch.

File manager.. (Dateimanager)

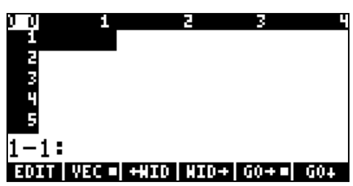
Option *7.File manager..* des Menüs APPS startet den Dateimanager:



Diese Operation entspricht der Tastenkombination ↰ FILES . Weitere Informationen zum Dateimanager finden Sie in Kapitel 2.

Matrix Writer.. (MatrixWriter)

Option *8.Matrix Writer..* des Menüs APPS startet den MatrixWriter:



Diese Operation entspricht der Tastenkombination ↰ MTRW . Weitere Informationen zum MatrixWriter finden Sie in Kapitel 10

Text editor.. (Texteditor)

Option *9.Text editor..* des Menüs APPS startet den Zeilen Texteditor:

In vielen Fällen kann der Texteditor durch Drücken der Taste gestartet werden. Ist jedoch das im Display angezeigte Objekt ein algebraisches Objekt, wird mit der Taste in der Regel der EquationWriter gestartet. Der Texteditor wird in Kapitel 2 vorgestellt und in Appendix L detailliert erläutert.

Math menu .. (Menü Mathematik)

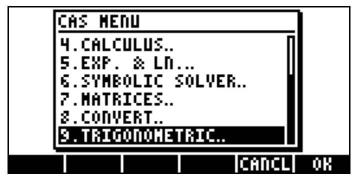
Die Option *10.Math menu..* im APPS öffnet das MTH (mathematische) Menü:

Diese Operation entspricht der Tastenkombination MTH . Weitere Informationen zum Menü MTH finden Sie in Kapitel 3 (reelle Zahlen) und in den Kapiteln 4 (komplexe Zahlen), 8 (Listen), 9 (Vektoren), 10 (Erzeugen von

Matrizen), 11 (Matrix Operationen), 16 (Schnelle Fourier-Transformationen), 17 (Wahrscheinlichkeitsrechnungen) und 19 (Zahlensysteme).

CAS menu.. (Menü CAS)

Option *11.CAS menu..* des Menüs APPS öffnet das Menü CAS oder SYMBOLIC:



Das Menü erscheint auch durch Drücken der Taste SYMB. Weitere Informationen zu dem Menü CAS bzw. SYMBOLIC finden Sie in den Kapiteln 5 (Algebraische und arithmetische Operationen), 4 (Berechnungen mit komplexen Zahlen), 6 (Lösung für Einzelgleichungen), 10 (Erstellen und Manipulieren von Matrizen), 11 (Matrix-Operationen und lineare Algebra), 13 (Infinitesimalrechnung), 14 (Anwendungen multivariater Infinitesimalrechnung) und 15 (Vektoranalyse).

Anhang G
Nützliche Tastaturkürzel

Hier finden Sie einige gebräuchliche Tastaturkürzel, die bei diesem Taschenrechner benutzt werden können:

- Einstellen des Display-Kontrasts: ON (festhalten) +, oder ON (festhalten) −

- Betätigen zwischen RPN und ALG Modi: MODE +/- OK oder MODE +/- ENTER .

- Setzen/Löschen des Systemflags 95 (ALG- vs. RPN-Modus)
  MODE FLAGS ▲ ← ▲ ← ▲ ← ▲ ✓CHK

  - Im ALG-Modus:
    CF(-95) wählt den RPN-Modus

  - Im RPN-Modus:
    95 +/- ENTER SF wählt den ALG-Modus

  Ein Tastaturkürzel zum Umschalten zwischen den Modi APPROX und EXACT besteht im Halten der rechten Shift-Taste und gleichzeitigem Drücken der Taste Enter, d. h.: ↱ (festhalten) ENTER .

- Setzen/Löschen des Systemflags 105 (EXACT vs. APPROX CAS-Modus)

  MODE FLAGS ▲ ← ▲ ← ▲ ▲ ▲ ✓CHK

  - Im ALG-Modus:
    SF(-105) wählt den APPROX CAS-Modus
    CF(-105) wählt den EXACT CAS-Modus

  - Im RPN-Modus:
    105 +/- ENTER SF wählt den APPROX CAS-Modus

105 +/- ENTER CF wählt den EXACT CAS-Modus

- Setzen/Löschen des Systemflags 117 (CHOOSE Boxes vs. SOFT-Menüs):

  MODE FLAGS ▲ ← ▲ ▼ ✓CHK

  - Im ALG-Modus:
    SF(-117) wählt die SOFT Menüs
    CF(-117) wählt die CHOOSE Boxes.

  - Im RPN-Modus:
    117 +/- ENTER SF wählt die SOFT Menüs
    117 +/- ENTER CF wählt die SOFT Menüs

- Winkelmaßänderung:
  - auf Grad: ALPHA ALPHA D E G ENTER
  - auf Radiant: ALPHA ALPHA R A D ENTER

- Sonderzeichen:
  - Symbol für Winkel (∠): ALPHA → 6
  - Symbol für Faktorielle (!): ALPHA → 2
  - Symbol für Grad (°): ALPHA → (festhalten) 6

- Sperren/Entsperren der Alphatastatur:
  - Sperren der Alphatastatur (Großbuchstaben): ALPHA ALPHA
  - Entsperren der Alphatastatur (Großbuchstaben): ALPHA
  - Sperren der Alphatastatur (Kleinbuchstaben): ALPHA ALPHA ← ALPHA
  - Entsperren der Alphatastatur (Kleinbuchstaben): ← ALPHA ALPHA

- Griechische Buchstaben:

Alpha (α):	ALPHA → A	Beta (β):	ALPHA → B
DELTA (Δ):	ALPHA → C	Delta (d):	ALPHA → D
Epsilon (ε):	ALPHA → E	Rho (ρ):	ALPHA → F
Mu (μ):	ALPHA → M	Lambda (λ):	ALPHA → N
PI (Π):	ALPHA → P	Sigma (σ):	ALPHA → S

Theta (θ): ALPHA → T   Tau (t): ALPHA → U
Omega (ω): ALPHA → V

- System-Level-Betrieb (Halten Sie ON gedrückt, lassen Sie die Taste los nachdem Sie die zweite oder dritte Taste gedrückt haben):
  - ON (festhalten) F1 F6 : "Kaltstart" – der gesamte Speicher wird gelöscht
  - ON (festhalten) F2 : macht den letzten Tastenanschlag rückgängig
  - ON (festhalten) F3 : "Warmstart" – Speicherinhalt bleibt erhalten
  - ON (festhalten) F4 : Startet interaktiven Selbsttest
  - ON (festhalten) F5 : Startet kontinuierlichen Selbsttest
  - ON (festhalten) SPC : Ruhemodus – Timer aus
  - ON (festhalten) F1 : Erzeugt eine Bildschirmmomentaufnahme
  - ON (festhalten) F4 : Storniert den nächsten zu wiederholenden Alarm

- Menüs, auf die nicht über die Tastatur zugegriffen werden kann: Im RPN-Modus tippen Sie MENÜ(Nummer des Menüs). Im ALG-Modus tippen Sie MENÜ(Nummer des Menüs). Nummer des Menüs kann einer der folgenden Werte sein:
  - Softmenü STAT: 96
  - Softmenü PLOT: 81
  - Softmenü SOLVE: 74, oder → (festhalten) 7
  - Softmenü UTILITY: 113

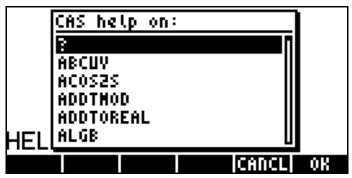
- Andere Menüs:
  - Menü MATHS: ALPHA ALPHA M A T H S ENTER
  - Menü MAIN: ALPHA ALPHA M A I N ENTER

- Andere Tastaturkürzel:
  - → (festhalten) 7 : Menü SOLVE (Menü 74)
  - ← (festhalten) MODE : Menü PRG/MODUS (Kapitel 21)

- ○ ⬑ (festhalten) ▼ : Startet den Texteditor (Anhang L)
- ○ ⬑ (festhalten) UPDIR : HOME(), springt ins HOME-Verzeichnis
- ○ ⬑ (festhalten) PREV : Wiederherstellen des letzten aktiven Menüs
- ○ ⬏ (festhalten) ▼ : Anzeigen von Variableninhalten
- ○ oder Menüeinträgen
- ○ ⬏ (festhalten) CHARS : Menü PRG/CHAR (Kapitel 21)

Anhang H
CAS-Hilfesystem

Auf das CAS-Hilfesystem können Sie mit der Tastenkombination TOOL NXT HELP ENTER zugreifen. Die folgende Abbildung stellt die erste Seite der Auflistung des CAS-Hilfesystems dar:



Die Befehle sind in alphabetischer Folge aufgelistet. Mithilfe der Pfeiltasten ▲▼ können Sie sich durch die Liste bewegen. Im Folgenden finden Sie einige nützliche Hinweise zur Steuerung des Hilfesystems.

- Sie können die Nach-Unten-Taste ▼ gedrückt halten und den Bildschirm so lange beobachten, bis der gesuchte Befehl erscheint. Lassen Sie jetzt die Nach-Unten-Taste los. Aller Wahrscheinlichkeit nach wird der gesuchte Befehl in diesem Moment nicht ausgewählt sein. (Sie werden möglicherweise über den Befehl hinausschießen oder davor stoppen). Sie können jedoch die Suche verlangsamen, indem Sie die Pfeiltasten ▲▼ einzeln drücken, um den gewünschten Befehl auszuwählen und dann OK drücken.
- Falls Sie bei gedrückter Nach-Unten-Taste ▼ über den gesuchten Befehl hinausschießen, können Sie die Nach-Oben-Taste ▲ gedrückt halten, um den Befehl in entgegengesetzter Richtung suchen. Verlangsamen Sie Ihre Auswahl durch einzelnes Drücken der Pfeiltasten ▲▼.
- Sie können jedoch auch den Anfangsbuchstaben des gesuchten Befehls tippen und dann die Nach-Unten-Taste ▼ zum Auffinden des jeweiligen Befehls verwenden. Angenommen, Sie suchen nach dem Befehl DERIV. Nach dem Aufrufen des Hilfesystems (TOOL NXT HELP ENTER) tippen Sie ALPHA D. Dadurch wird der erste Befehl der

mit D beginnt, ausgewählt, d. h. DEGREE. Um zu DERIV zu gelangen, drücken Sie die Pfeiltaste ▼ zweimal. Um den Befehl auszuwählen, drücken Sie OK.

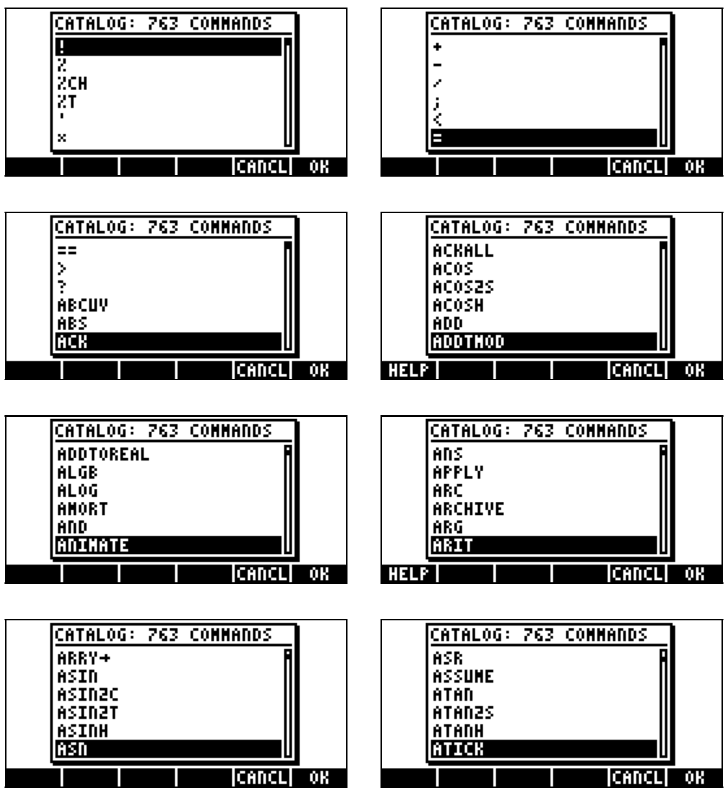
- Durch Sperren der alphabetischen Tastatur können Sie zwei oder mehr Anfangsbuchstaben des gesuchten Befehls eingeben. Auf diese Weise werden Sie direkt zum gesuchten Befehl oder zumindest in seine Nähe gebracht. Danach müssen Sie die Feststellung der alphabetischen Tastatur wieder lösen und, falls nötig, die Nach-Oben- und Nach-Unten-Tasten ▲▼ zum Auffinden des Befehls verwenden. Drücken Sie OK, um den Befehl auszuwählen. Um beispielsweise den Befehl PROPFRAC zu finden, können Sie eine der nachstehenden Tastenfolgen verwenden:

TOOL NXT HELP ENTER ALPHA ALPHA P R ALPHA ▼ ▼ OK
TOOL NXT HELP ENTER ALPHA ALPHA P R O ALPHA ▼ OK
TOOL NXT HELP ENTER ALPHA ALPHA P R O P ALPHA OK

In Anhang C finden Sie weitere Informationen zu CAS (Computer Algebraic System). Anhang C enthält auch weitere Anwendungsbeispiele des CAS-Hilfesystems.

Anhang I
Liste der Befehle im Befehlskatalog

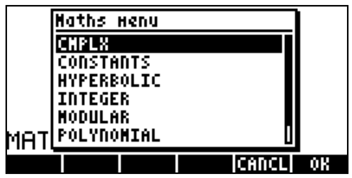
Dies ist eine Liste aller zur Verfügung stehenden Befehle des Befehlskatalogs (→ CAT). Befehle, die zum CAS-Modul (Computer Algebraic System) gehören, sind auch in Anhang H aufgeführt. Einträge des CAS-Hilfesystem sind für einen Befehl verfügbar, wenn die Funktionstaste HELP beim Hervorheben des gewünschten Befehls erscheint. Drücken Sie diese Funktionstaste, um die CAS-Hilfe für den Befehl zu erhalten. Einige der ersten Hilfsbildschirme des Katalogs werden nachfolgend abgebildet:



Die vom Anwender definierten Befehle erscheinen auch in der Liste des Befehls-Katalogs in Kursivschrift. Ist eine Hilfefunktion zur Bibliothek verfügbar, erscheint die Funktionstaste HELP , sobald Sie einen anwenderdefinierten Befehl anklicken, für den ein Hilfetext hinterlegt wurde.

Anhang J
Das Menü MATHS

Das Menü MATHS, auf welches mit dem Befehl MATHS (verfügbar im Befehlskatalog CAT) zugegriffen werden kann, enthält folgende Untermenüs:



Das Untermenü CMPLX

Das Untermenü CMPLX enthält Funktionen für Operationen mit komplexen Zahlen:

Eine Beschreibung dieser Funktionen finden Sie in Kapitel 4.

Das Untermenü CONSTANTS

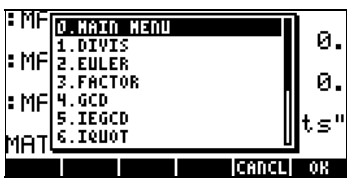
Das Untermenü CONSTANTS erlaubt den Zugriff auf die im Taschenrechner verfügbaren mathematischen Konstanten. Eine Beschreibung dieser Funktionen finden Sie in Kapitel 3:

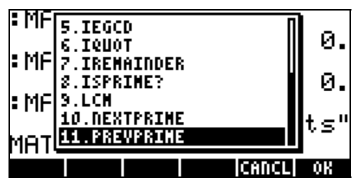
Das Untermenü HYPERBOLIC

Das Untermenü HYPERBOLIC enthält die hyperbolischen Funktionen und deren Umkehrfunktionen. Eine Beschreibung dieser Funktionen finden Sie in Kapitel 3:

Das Untermenü INTEGER

Das Untermenü INTEGER stellt Funktionen zur Manipulation von Integer-Zahlen sowie einiger Polynome zur Verfügung. Eine Beschreibung dieser Funktionen finden Sie in Kapitel 5:





Das Untermenü MODULAR

Das Untermenü MODULAR stellt Funktionen für modulare Arithmetik mit Zahlen und Polynomen zur Verfügung. Eine Beschreibung dieser Funktionen finden Sie in Kapitel 5:

Das Untermenü POLYNOMIAL

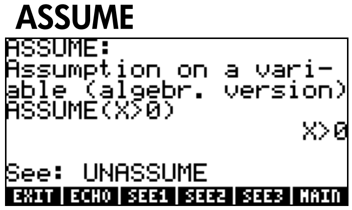
Das Untermenü POLYNOMIAL beinhaltet Funktionen zum Erstellen und Manipulieren von Polynomen. Eine Beschreibung dieser Funktionen finden Sie in Kapitel 5:

Das Untermenü TESTS

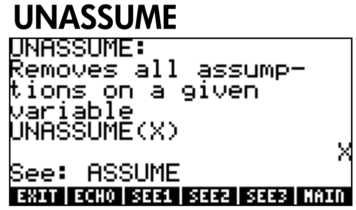
Das Untermenü TESTS beinhaltet relationale Operatoren (==, <, usw.), logische Operatoren (AND, OR, usw.), die Funktion IFTE und die Befehle ASSUME und UNASSUME.

Relationale und logische Operatoren werden im Kapitel 21 im Zusammenhang mit der Programmierung des Taschenrechners in der Programmiersprache UserRPL vorgestellt. Die Funktion IFTE wird in Kapitel 3 erläutert. Die Funktionen ASSUME und UNASSUME werden nachfolgend durch ihre Einträge im CAS-Hilfesystem dargestellt (siehe Anhang C).

ASSUME

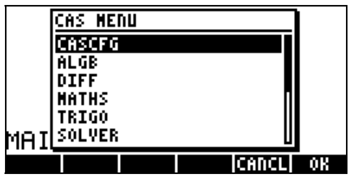


UNASSUME



Anhang K
Das Menü MAIN

Das Menü MAIN ist über den Befehls-Katalog verfügbar. Es enthält folgende Untermenüs:



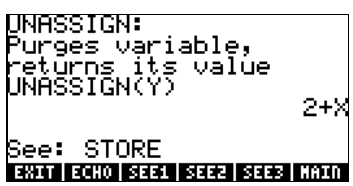
Der Befehl CASCFG

Dies ist der erste Eintrag im Menü MAIN. Dieser Befehl konfiguriert das CAS-Modul. Informationen zur CAS-Konfiguration erhalten Sie in Anhang C.

Das Untermenü ALGB

Das Untermenü ALGB enthält die folgenden Befehle:

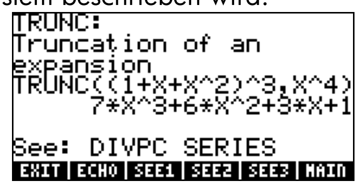
Diese Funktionen, mit Ausnahme von 0. MAIN MENU und 11.UNASSIGN sind im ALG-Tastaturmenü (ALG) verfügbar. Ausführliche Erklärungen dieser Funktionen finden Sie in Kapitel 5. Die Funktion UNASSIGN ist in folgendem Eintrag des Menüs CAS beschrieben:



Das Untermenü DIFF

Das Untermenü DIFF enthält die folgenden Funktionen:

Diese Funktionen sind auch im Untermenü CALC/DIFF (CALC) verfügbar. Eine Beschreibung dieser Funktionen finden Sie in den Kapiteln 13, 14 und 15, mit Ausnahme der Funktion TRUNC, die nachfolgend anhand ihres Eintrags im CAS-Hilfesystem beschrieben wird.



Das Untermenü MATHS

Eine genaue Beschreibung des Untermenüs MATHS finden Sie in Anhang J.

Das Untermenü TRIGO

Das Untermenü TRIGO enthält die folgenden Funktionen:

Diese Funktionen sind auch im TRIG Menü (→ TRIG) verfügbar. Eine Beschreibung dieser Funktionen finden Sie in Kapitel 5.

Das Untermenü SOLVER

Das Untermenü SOLVER enthält die folgenden Funktionen:

Diese Funktionen sind im Menü CALC/SOLVE (← CALC) verfügbar. Eine Beschreibung dieser Funktionen finden Sie in den Kapiteln 6, 11 und 16.

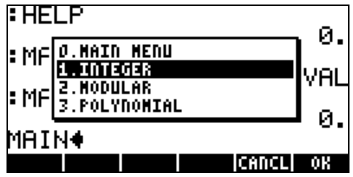
Das Untermenü CMPLX

Das Untermenü CMPLX enthält die folgenden Funktionen:

Das Untermenü CMPLX kann auch über die Tastatur erreicht (→ CMPLX) werden. Einige dieser Funktionen des Menüs CMPLX sind auch im Menü MTH/COMPLEX verfügbar (← MTH). Eine Beschreibung der Funktionen für komplexe Zahlen finden Sie Kapitel 4.

Das Untermenü ARIT

Das Menü ARIT enthält die folgenden Untermenüs:



Die Untermenüs INTEGER, MODULAR und POLYNOMIAL werden ausführlich in Anhang J beschrieben.

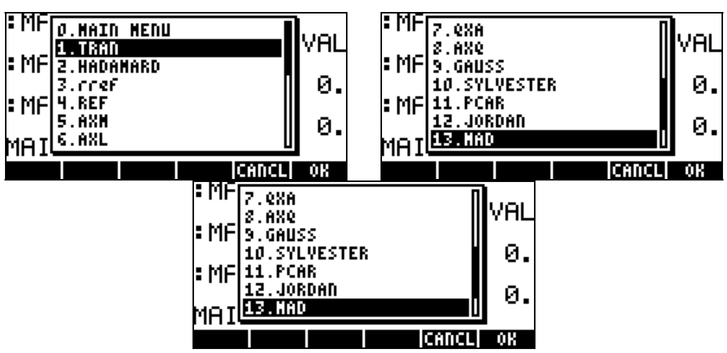
Das Untermenü EXP&LN

Das Menü EXP&LN enthält die folgenden Funktionen:

Dieses Menü kann auch über die Tastatur über ← EXP&LN erreicht werden. Eine Beschreibung der Funktionen dieses Menüs finden Sie in Kapitel 5.

Das Untermenü MATR

Das Menü MATR enthält die folgenden Funktionen:



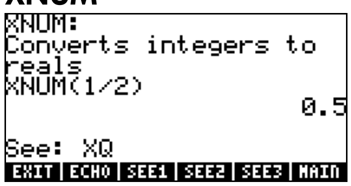
Diese Funktionen sind auch im Menü MATRICES der Tastatur (← MATRICES) verfügbar. Eine Beschreibung dieser finden Sie in den Kapiteln 10 und 11.

Das Untermenü REWRITE

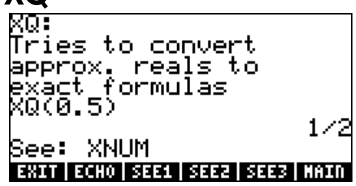
Das Menü REWRITE enthält die folgenden Funktionen:

Diese Funktionen sind verfügbar im Menü CONVERT/REWRITE (starten Sie dieses mit ← CONVERT). Eine Beschreibung dieser Funktionen finden Sie in Kapitel 5, ausgenommen die Funktionen XNUM und XQ, die nachfolgend anhand ihrer Einträge im CAS-Hilfsystem erläutert werden. (TOOL NXT HELP):

XNUM

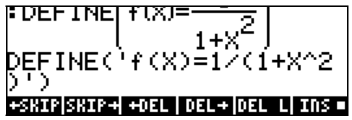


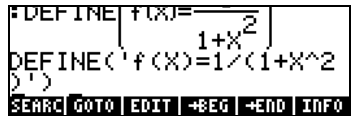
XQ

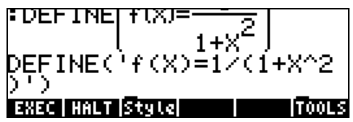


Anhang L
Befehle des Zeileneditors

Rufen Sie im RPN-Stack oder im ALG-Modus den Zeileneditor mit ↰ ▼ auf, werden Ihnen die folgenden Untermenüs zur Verfügung gestellt (drücken Sie NXT, um die verbleibenden Funktionen zu sehen):

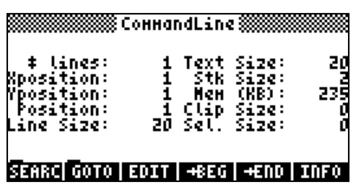






Die Funktionen werden wie folgt kurz beschrieben:

←SKIP: Überspringt alle Zeichen bis zum Wortanfang.
SKIP→: Überspringt alle Zeichen zum Wortende.
←DEL: Löscht alle Zeichen bis zum Wortanfang.
DEL→: Löscht alle Zeichen bis zum Wortende.
DEL L: Löschen alle Zeichen einer Zeile.
INS: Wenn ausgewählt, werden Zeichen an der Cursor-Position eingefügt. Wenn nicht ausgewählt, ersetzt der Cursor diese Zeichen (sie werden überschrieben).
EDIT: Bearbeitet den ausgewählten Bereich.
→BEG: Bewegt den Cursor an den Wortanfang.
→END: Markiert das Ende eines ausgewählten Bereichs.
INFO: Gibt Information zum Befehlszeileneditor, beispielsweise:



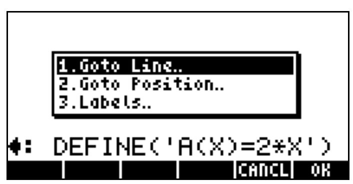
Die in dieser Abbildung angezeigten Daten sind selbsterklärend. Beispielsweise bezeichnen *X* und Y *positions* die Position auf einer Zeile (X) und die Zeilennummer (Y). *Stk Size* bezeichnet die Anzahl der Objekte in der ALG-Modus-History oder im RPN-Stack. *Mem(KB)* bezeichnet die Größe des freien Speichers. *Clip Size* ist die Anzahl der Zeichen im Clipboard. *Sel Size* ist die Anzahl der Zeichen im aktuell ausgewählten Bereich.

EXEC: führt den gewählten Befehl aus
HALT: stoppt die Befehlsausführung.

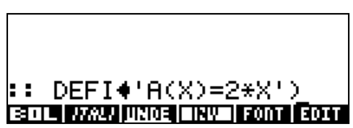
Der Zeileneditor stellt die folgenden Untermenüs zur Verfügung:

SEARCH: sucht Zeichen oder Worte in der Befehlszeile und umfasst folgende Funktionen:

GOTO: springt zu einer gewünschten Position in der Befehlszeile und enthält folgende Funktionen:



Style: Textstile, die in der Befehlszeile verwendet werden können:



Das Untermenü SEARCH

Die Funktionen des Untermenüs SEARCH sind:

Find : Benutzen Sie diese Funktion, um einen String in der Befehlszeile zu finden. Nachfolgend das Eingabeformular, dass bei diesem Befehl zur Verfügung steht:

Replace: Benutzen Sie diesen Befehl, um einen String zu finden und zu ersetzen. Nachfolgend das Eingabeformular, das bei diesem Befehl zur Verfügung steht:

Find next..: Findet das nächste Muster, das den im Find-Befehl definierten Kriterien entspricht

Replace Selection: Ersetzt den ausgewählten Bereich mit dem im Replace-Befehl definierten Muster.

Replace/Find Next: Ersetzt ein Muster und sucht nach dem nächsten Vorkommen des im Replace-Befehl definierten Musters.

Replace All: Ersetzt alle Vorkommen eines bestimmten Musters. Bei diesem Befehl wird eine Bestätigung des Anwenders vor dem Ersetzen des Musters erwartet.

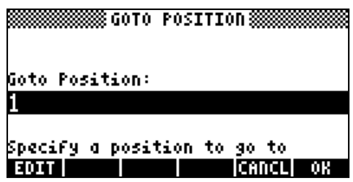
Fast Replace All: Ersetzt alle Vorkommen eines bestimmten Musters ohne Rückfrage an den Anwender.

Das Untermenü GOTO

Die Funktionen des Untermenüs GOTO sind:

Goto Line: springt zu einer angegebenen Zeile. Das Eingabeformular für diesen Befehl sieht wie folgt aus:

Goto Position: springt zu einer angegebenen Position in der Befehlszeile. Das Eingabeformular für diesen Befehl sieht wie folgt aus::



Labels: springt zu einer angegebenen Marke in der Befehlszeile.

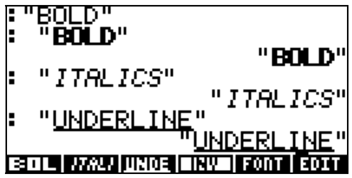
Das Untermenü Style

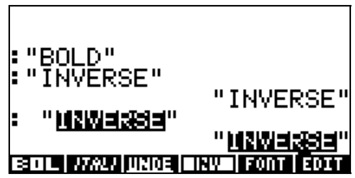
Das Untermenü Style enthält die folgenden Styles:

BOL: Bold (fett)
ITALI: Italics (kursiv)
UNDE: Underline (unterstrichen)
INV : Inverse (umgekehrt)

Mit dem Befehl FONT kann der Anwender die Schrift für den Befehlszeileneditor auswählen.

Nachfolgend einige Beispiele für verschiedene Styles:


:"BOLD"
: "BOLD"
"BOLD"
: "ITALICS"
"ITALICS"
: "UNDERLINE"
"UNDERLINE"
BOL ITAL UNDE INV FONT EDIT


:"BOLD"
:"INVERSE"
"INVERSE"
: "INVERSE"
"INVERSE"
BOL ITAL UNDE INV FONT EDIT

Anhang M
Index

A

B

C

D

E

F

J

K

N

O

P

Q

R

S

T

U

Beschränkte Garantie

Grafiktaschenrechner hp 48gII, Garantiezeitraum: 12 Monate

1. HP garantiert Ihnen, dem Endbenutzer, dass HP Hardware, Zubehör und Verbrauchsmaterialien frei von Material- und Verarbeitungsfehlern sind. Diese Garantie beginnt mit dem Kaufdatum und gilt für den oben angegebenen Zeitraum. Wenn HP innerhalb des Garantiezeitraums über einen derartigen Mangel informiert wird, übernimmt HP nach eigenem Ermessen entweder die Reparatur des nachweislich fehlerhaften Produkts oder tauscht dieses aus. Als Austauschprodukte können neue oder im Hinblick auf die Leistung neuwertige Produkte eingesetzt werden.
2. Bei ordnungsgemäßer Installation und Verwendung der HP Software übernimmt HP ab dem Kaufdatum und für den oben angegebenen Zeitraum die Garantie, dass keine Material- oder Verarbeitungsfehler bestehen, die dazu führen, dass die Programmierungsanweisungen nicht ausgeführt werden können. Wenn HP innerhalb des Garantiezeitraums über einen derartigen Mangel informiert wird, ersetzt HP die fehlerhafte Software, die die Programmierungsanweisungen nicht ausführt.
3. HP übernimmt keine Garantie für einen störungs- oder fehlerfreien Betrieb von HP Produkten. Sollte HP innerhalb eines angemessenen Zeitraums nicht in der Lage sein, ein Produkt gemäß den Garantiebestimmungen zu reparieren oder auszutauschen, sind Sie bei sofortiger Rücksendung des Produkts berechtigt, den Kaufpreis zurückzuverlangen.
4. HP Produkte können aus recyceltem Material hergestellte Teile enthalten, deren Leistung Teilen aus neuem Material entspricht oder die unbeabsichtigt verwendet wurden.
5. Die Garantie gilt nicht für Mängel, die auf Folgendes zurückzuführen sind: (a) unsachgemäße oder ungeeignete Wartung oder Kalibrierung; (b) Verwendung von Software, Schnittstellen, Teilen oder Verbrauchsmaterialien, die nicht von HP zur Verfügung gestellt wurden; (c) unbefugte Änderung oder falsche Verwendung; (d) Betrieb außerhalb des Rahmens der für das Produkt veröffentlichen technischen Daten für den Betrieb; oder (e) unsachgemäße Vorbereitung oder

Wartung des Standorts.

6. SOWEIT GEMÄSS ÖRTLICHEM RECHT ZULÄSSIG, STELLEN DIE OBEN GENANNTEN GARANTIEANSPRÜCHE DIE ALLEINIGEN ANSPRÜCHE DAR, UND ES GELTEN KEINE WEITEREN SCHRIFTLICHEN ODER MÜNDLICHEN GARANTIEN ODER GEWÄHRLEISTUNGEN, WEDER AUSDRÜCKLICH NOCH STILLSCHWEIGEND. HP WEIST INSBESONDERE ALLE STILLSCHWEIGENDEN GARANTIEN ODER GEWÄHRLEISTUNGEN BEZÜGLICH MARKTGÄNGIGKEIT; ZUFRIEDENSTELLENDER QUALITÄT UND EIGNUNG ZU EINEM BESTIMMTEN ZWECK ZURÜCK. Einige Staaten, Länder oder Provinzen lassen eine Einschränkung des Garantiezeitraums für eine konkludente Garantie nicht zu. In diesem Fall finden die oben genannten Einschränkungen oder Ausschlüsse keine Anwendung. Aus dieser Garantie ergeben sich für Sie bestimmte Rechte. Darüber hinaus haben Sie möglicherweise weitere Rechte, die von Staat zu Staat, von Land zu Land oder von Provinz zu Provinz unterschiedlich sein können.

7. SOWEIT GEMÄSS ÖRTLICHEM RECHT ZULÄSSIG, STELLEN DIE IN DIESER GARANTIEERKLÄRUNG GENANNTEN ANSPRÜCHE IHRE ALLEINIGEN UND AUSSCHLIESSLICHEN ANSPRÜCHE DAR. MIT AUSNAHME DER VORSTEHENDEN BESTIMMUNG ÜBERNEHMEN HP UND IHRE ANBIETER IN KEINEM FALL EINE HAFTUNG FÜR EINEN DATENVERLUST ODER FÜR DIREKTE, SPEZIELLE, ZUFÄLLIG ENTSTANDENE SCHÄDEN ODER FOLGESCHÄDEN (EINSCHLIESSLICH ENTGANGENEM GEWINN ODER DATENVERLUST), ODER SONSTIGE SCHÄDEN, UNABHÄNGIG DAVON, OB DIESE VERTRAGLICH FESTGEHALTEN SIND ODER AUS EINER UNERLAUBTEN HANDLUNG ODER ANDERWEITIG ENTSTEHEN. Einige Staaten, Länder oder Provinzen lassen den Ausschluss oder die Beschränkung von zufällig entstandenen Schäden oder Folgeschäden nicht zu. In diesem Fall finden die oben genannten Einschränkungen oder Ausschlüsse keine Anwendung.

8. Die Garantien für HP Produkte und HP Dienstleistungen werden ausschließlich in den ausdrücklichen Garantieerklärungen dargelegt, die im Lieferumfang dieser Produkte und Dienstleistungen enthalten sind. HP haftet nicht für technische oder redaktionelle Fehler oder Auslassungen in diesen Dokumenten.

FÜR ENDVERBRAUCHER-KAUFABSCHLÜSSE IN AUSTRALIEN UND NEUSEELAND: SOFERN GEMÄSS GELTENDEM RECHT ZULÄSSIG, SCHLIESSEN DIE IN DIESER ERKLÄRUNG ENTHALTENEN GARANTIEBESTIMMUNGEN DIE VERBINDLICHEN, GESETZLICH FESTGELEGTEN RECHTE FÜR DEN VERKAUF DIESES PRODUKTS AN SIE WEDER AUS NOCH SCHRÄNKEN SIE DIESE EIN ODER ÄNDERN DIESE, SONDERN ERWEITERN DIESE RECHTE.

Service

Europa

Land:	Telefonnummern
Österreich	+43-1-3602771203
Belgien	+32-2-7126219
Dänemark	+45-8-2332844
Osteuropäische Staaten	+420-5-41422523
Finnland	+35-89640009
Frankreich	+33-1-49939006
Deutschland	+49-69-95307103
Griechenland	+420-5-41422523
Holland	+31-2-06545301
Italien	+39-02-75419782
Norwegen	+47-63849309
Portugal	+351-229570200
Spanien	+34-915-642095
Schweden	+46-851992065
Schweiz	+41-1-4395358 (deutsch) +41-22-8278780 (französisch) +39-02-75419782 (italienisch)
Türkei	+420-5-41422523
Groß Britannien	+44-207-4580161
Tschechien	+420-5-41422523
Südafrika	+27-11-237 62 00
Luxemburg	+32-2-7126219
andere europäische Länder	+420-5-41422523

Asien Pazifik

Land:	Telefonnummern

Australien	+61-3-9841-5211
Singapur	+61-3-9841-5211

Lat. Amerika

Land:	Telefonnummern
Argentinien	0-810-555-5520
Brasilien	Sao Paulo 3747-7799; ROTC 0-800-157751
Mexiko	Mx City 5258-9922; ROTC 01-800-472-6684
Venezuela	0800-4746-8368
Chile	800-360999
Kolumbien	9-800-114726
Peru	0-800-10111
Mittelamerika & Karibik	1-800-711-2884
Guatemala	1-800-999-5105
Puerto Rico	1-877-232-0589
Costa Rica	0-800-011-0524

N. Amerika

Land:	Telefonnummern
USA	1800-HP INVENT
Kanada	(905) 206-4663 or 800- HP INVENT

ROTC = Rest des Landes

Unter http://www.hp.com finden Sie die neuesten Service- und Support-Informationen.

Regulierungsinformationen

Dieser Abschnitt enthält wichtige Informationen zur Konformität des Grafiktaschenrechners hp 48gII mit Bestimmungen in bestimmten Regionen. Änderungen am Taschenrechner, die nicht ausdrücklich von Hewlett-Packard genehmigt wurden, können zum Verlust der Betriebserlaubnis in diesen Regionen führen.

USA

This calculator generates, uses, and can radiate radio frequency energy and may interfere with radio and television reception. The calculator complies with the limits for a Class B digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation.

However, there is no guarantee that interference will not occur in a particular installation. In the unlikely event that there is interference to radio or television reception(which can be determined by turning the calculator off and on), the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Relocate the calculator, with respect to the receiver.

Connections to Peripheral Devices

To maintain compliance with FCC rules and regulations, use only the cable accessories provided.

Canada

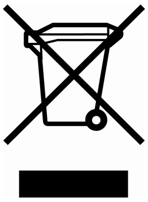
This Class B digital apparatus complies with Canadian ICES-003. Cet appareil numerique de la classe B est conforme a la norme NMB-003 du Canada.

Japan

この装置は、情報処理装置等電波障害自主規制協議会(VCCI)の基準に基づく第二情報技術装置です。この装置は、家庭環境で使用することを目的としていますが、この装置がラジオやテレビジョン受信機に近接して使用されると、受信障害を引き起こすことがあります。

取扱説明書に従って正しい取り扱いをしてください。

Entsorgung von Altgeräten aus privaten Haushalten in der EU



Das Symbol auf dem Produkt oder seiner Verpackung weist darauf hin, dass das Produkt nicht über den normalen Hausmüll entsorgt werden darf. Benutzer sind verpflichtet, die Altgeräte an einer Rücknahmestelle für Elektro- und Elektronik-Altgeräte abzugeben. Die getrennte Sammlung und ordnungsgemäße Entsorgung Ihrer Altgeräte trägt zur Erhaltung der natürlichen Ressourcen bei und garantiert eine Wiederverwertung, die die Gesundheit des Menschen und die Umwelt schützt. Informationen dazu, wo Sie Rücknahmestellen für Ihre Altgeräte finden, erhalten Sie bei Ihrer Stadtverwaltung, den örtlichen Müllentsorgungsbetrieben oder im Geschäft, in dem Sie das Gerät erworben haben.