

HP 35s 関数電卓

ユーザーズガイド



i n v e n t

第 1 版

HP 部品番号 F2215AA-90019

ご注意

次の URL で製品登録を行ってください。www.register.hp.com

本書に記載した内容（その例も含む）は、「現状のまま」で提供され、予告なしに変更することがあります。当社は、商品性、権利非侵害、および特定目的に対する適合性についての暗黙の保証を含め、またこれに限定されることなく、本書についての保証は一切行いません。

当社は、本書の誤りや本書の提供、内容の実施、また本書の利用によって偶発的あるいは必然的に生じる損害については責任を負わないものとします。

© 1988, 1990-1991, 2003, 2007 Hewlett-Packard Development Company, L.P.

当社の文書による事前の承諾なく、本書の内容の一部または全部を複製、改変、および翻訳することは、著作権法下での許可事項を除き、禁止されています。

Hewlett-Packard Company
16399 West Bernardo Drive
MS 8-600
San Diego, CA 92127-1899
USA

版歴

第 1 版

2007 年 2 月

目次

第 1 部． 基本操作

1. はじめに	1-1
重要な予備知識	1-1
電源のオン / オフ	1-1
ディスプレイのコントラストの調整	1-1
キーボードとディスプレイの主要部分	1-2
シフトキー	1-2
アルファベットキー	1-3
カーソルキー	1-3
バックスペースとクリア	1-4
メニューの 使用	1-6
メニューの終了	1-9
RPN モードと ALG モード	1-9
Undo キー	1-11
ディスプレイとインジケータ	1-12
数値の入力	1-15
数値を負の値に変更する	1-15
10 の指数	1-15
入力カーソルについて	1-17
数値の範囲とオーバーフロー	1-17
算術計算の実行	1-18
単一引数（単項）演算	1-18
2 つの引数をとる（2 項）演算	1-19
表示形式の制御	1-21
数値内のピリオドとカンマ（・）（,）	1-23

複素数の表示形式 ($x+yi$ 、 $x+yi$ 、 $r\theta a$)	1-24
フル 12 桁精度の表示	1-25
分数	1-26
分数の入力	1-26
メッセージ	1-28
電卓のメモリ	1-28
メモリの空き容量を調べる	1-28
すべてのメモリをクリアする	1-29

2. RPN : 自動メモリストック2-1

スタックとは	2-1
X レジスタと Y レジスタのディスプレイ表示	2-3
X レジスタのクリア	2-3
スタックの確認	2-3
スタックの X レジスタと Y レジスタの交換	2-4
スタックでの四則計算の方法	2-5
ENTER の動作	2-6
スタックのクリア方法	2-7
最後の X レジスタ	2-8
LAST X を使ったエラーの訂正	2-9
LAST X による数値の再使用	2-10
RPN モードでの連鎖計算	2-12
括弧無しでの作業	2-12
練習	2-14
計算の順番	2-15
追加練習	2-16

3. データの変数への格納3-1

数値の格納と呼び出し	3-2
変数の表示	3-4

MEM カタログの使用	3-5
VAR カタログ	3-5
格納した変数を使った計算	3-7
STO 四則演算	3-7
RCL 四則演算	3-8
ある変数と x を交換	3-10
変数 I と J	3-10
4. 実数の関数	4-1
指数関数と対数関数	4-2
除算の商と剰余	4-2
累乗関数	4-3
三角関数	4-4
π の入力	4-4
角度モードの設定	4-4
三角関数	4-5
双曲線関数	4-6
% 関数	4-6
物理定数	4-8
変換関数	4-10
直交座標／極座標変換	4-10
時間の変換	4-13
角度の変換	4-13
単位変換	4-14
確率関数	4-15
階乗	4-15
ガンマ	4-15
確率	4-15
数値の部分	4-17

5. 分数	5-1
分数の入力	5-1
ディスプレイでの分数	5-2
表示のルール	5-2
確度インジケータ	5-3
分数表示の変更	5-4
最大分母の設定	5-4
分数の表示形式の選択	5-6
分数表示の例	5-8
分数の丸め	5-9
方程式内の分数	5-10
プログラム内の分数	5-10
6. 方程式の入力と評価	6-1
方程式の使いかた	6-1
式の操作のまとめ	6-3
方程式リストへの方程式の入力	6-4
方程式内の変数	6-4
方程式内の数値	6-5
方程式内の関数	6-5
方程式内の括弧	6-6
方程式の表示と選択	6-6
方程式の編集とクリア	6-8
方程式の種類	6-9
方程式の評価	6-10
評価のための ENTER の使用	6-11
評価のための XEQ の使用	6-13
方程式のプロンプトへの応答	6-14

方程式の文法	6-15
演算子の優先順位	6-15
方程式で利用できる関数	6-16
文法エラー	6-19
方程式の確認	6-19
7. 方程式を解く	7-1
方程式を解く方法	7-2
組み込みの方程式を解く	7-7
SOLVE の理解と制御	7-8
結果の確認	7-8
SOLVE 計算の中断	7-9
SOLVE の初期推定値の選択	7-9
詳細	7-13
8. 方程式の積分	8-1
方程式の積分 (∫ FN)	8-2
積分の確度	8-6
確度の指定	8-6
確度の解釈	8-6
詳細	8-8
9. 複素数.....	9-1
複素数スタック	9-2
複素数の計算	9-3
複素数と極座標	9-5
方程式内の複素数	9-7
プログラムにおける複素数	9-8

10.ベクトルの演算.....	10-1
ベクトルの演算.....	10-1
ベクトルの絶対値.....	10-3
ドット積.....	10-4
ベクトル間の角度.....	10-5
方程式内のベクトル.....	10-6
プログラム内のベクトル.....	10-6
変数またはレジスタからのベクトルの作成.....	10-7
11.基数変換と四則および論理.....	11-1
基数 2、8、16 の四則演算.....	11-4
数値の表現.....	11-6
負の数値.....	11-6
数値の範囲.....	11-7
長い 2 進数のウィンドウ.....	11-8
プログラムと方程式での基数の使用.....	11-8
12.統計演算.....	12-1
統計データの入力.....	12-1
1 変数データの入力.....	12-2
2 変数データの入力.....	12-2
データ入力のエラーの修正.....	12-3
統計の計算.....	12-4
平均.....	12-5
標本標準偏差.....	12-6
母標準偏差.....	12-7
線形回帰.....	12-8
データの精度に関する制限.....	12-11
総和と統計レジスタ.....	12-12
総和統計.....	12-12
統計レジスタへのアクセス.....	12-13

第 2 部． プログラミング

13.プログラムの基礎13-1

プログラムの設計	13-3
モードの選択.....	13-3
プログラムの境界 (LBL と RTN)	13-4
プログラムでの RPN、ALG、方程式の使用	13-4
データの入出力	13-5
プログラムの入力	13-6
クリア関数とバックスペースキー	13-7
プログラム内の関数名.....	13-8
プログラムの実行	13-10
プログラムの実行 (XEQ)	13-10
プログラムのテスト	13-11
データの入力と表示.....	13-12
INPUT を使用したデータの入力	13-13
VIEW を使用したデータの表示	13-15
方程式を使用したメッセージの表示	13-16
停止なしの情報の表示.....	13-18
プログラムの中断と一時停止.....	13-19
中断 (STOP) または一時停止 (PSE) のプログラミング ...	13-19
実行中のプログラムの停止	13-19
エラーによる停止	13-19
プログラムの編集.....	13-20
プログラムメモリ	13-21
プログラムメモリの表示	13-21
メモリの使用.....	13-22
プログラムのカタログ (MEM)	13-22

プログラムのクリア	13-23
チェックサム	13-23
プログラムできない機能	13-24
基数変換 (BASE) を使ったプログラム	13-25
プログラムにおける基数モードの選択	13-25
プログラム行に入力された数値	13-25
多項式とホーナーの方法	13-26

14. プログラムのテクニック 14-1

プログラムのルーチン	14-1
サブルーチンの呼び出し (XEQ、RTN)	14-2
サブルーチンの入れ子	14-3
分岐 (GTO)	14-5
プログラムされた GTO 命令	14-5
キーボードからの GTO の使用	14-6
条件命令	14-6
比較テスト (x?y, x?0)	14-7
フラグ	14-9
ループ	14-15
条件ループ (GTO)	14-16
カウンタによるループ (DSE、ISG)	14-16
変数とラベルの間接アドレッシング	14-19
変数 "I" と "J"	14-19
間接アドレス (I) と (J)	14-20
(I)/(J) によるプログラムの制御	14-21
(I)/(J) を使った方程式	14-21
名前のない間接変数	14-21

15.プログラムの SOLVE と積分	15-1
プログラムの SOLVE	15-1
プログラムでの SOLVE の使用	15-6
プログラムの積分	15-8
プログラムでの積分の使用	15-10
SOLVE と積分の制約	15-12
16.統計のプログラム	16-1
カーブフィッティング	16-1
正規分布と逆正規分布	16-11
グループ化された標準偏差	16-18
17.その他のプログラムと方程式	17-1
貨幣の時間価値	17-1
素数生成器	17-6
ベクトルのクロス積	17-10

第 3 部． 付録

A. サポート、電池、およびサービス	A-1
サポート	A-1
一般的な質問への回答	A-1
環境限界	A-2
電池の交換	A-3
電卓の動作テスト :	A-4
セルフテスト	A-6
保証	A-7
カスタマサポート	A-8
Regulatory information	A-12
Federal Communications Commission Notice	A-12

B. メモリとスタック	B-1
電卓メモリの管理	B-1
電卓のリセット	B-2
メモリのクリア	B-3
スタック上昇の状態	B-4
無効になる操作	B-4
中立的操作	B-5
LAST X レジスタの状態	B-6
スタックレジスタの内容へのアクセス	B-7
C. ALG: まとめ	C-1
ALG について	C-1
ALG モードにおける 2 つの引数をとる演算	C-2
単純な四則計算	C-2
累乗関数	C-3
百分率計算	C-3
順列と組合せ	C-4
除算の商と剰余	C-4
括弧を含む計算	C-5
指数関数と対数関数	C-6
三角関数	C-6
双曲線関数	C-7
数値の部分	C-7
スタックの確認	C-8
方程式の積分	C-8
複素数の計算	C-9
基数 2、8、16 の四則演算	C-11
2 変数の統計データの入力	C-12

D. SOLVE の詳細	D-1
SOLVE が根を見つける方法	D-1
結果の解釈	D-3
SOLVE が根を発見できなかったときは	D-8
丸め誤差	D-13
E. 積分の詳細	E-1
積分の評価方法	E-1
不正確な結果を引き起こす条件	E-2
計算時間が長くなる条件	E-7
F. メッセージ	F-1
G. 操作インデックス	G-1

第 1 部

基本操作

はじめに



余白部分にあるこの記号に注意してください。この記号は、PRN モードでの例またはキー入力であることを示します。

ALG モードでは入力方法が異なります。

付録 C で、ALG モードでの電卓の使用方法を説明します。

重要な予備知識

電源のオン / オフ

電卓の電源をオンにするには、**[C]** を押します。**[C]** キーの下部に、ON と印刷されています。

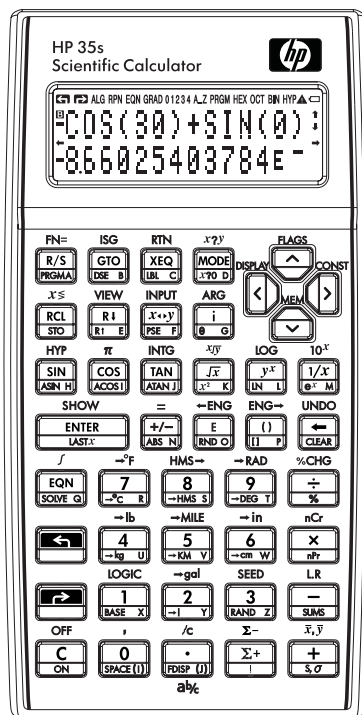
電卓の電源をオフにするには、**[⇐][C]** を押します。つまり、**[⇐]** Shift キーを押して離れた後、**[C]** を押します（キーの上に黄色で OFF と印刷されています）。電卓は**連続メモリ**を備えているため、電源をオフにしても格納した情報には影響がありません。

電池の節約のため、10 分間操作しないと自動的に電源が切れます。ディスプレイに電池容量低下記号（**[⏻]**）が表示された場合は、速やかに電池を交換してください。手順については、付録 A を参照してください。

ディスプレイのコントラストの調整

ディスプレイのコントラストは、照明、見る角度、コントラスト設定によって異なります。コントラストを調整するには、**[C]** キーを押しながら、**[+]** または **[-]** を押します。

キーボードとディスプレイの主要部分



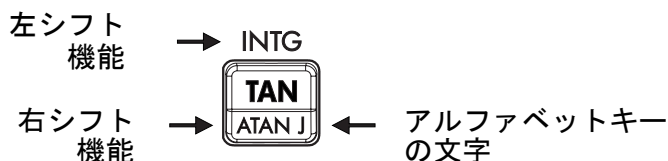
シフトキー

各キーには、3つの機能があります。キーの表面に印刷されている機能、左シフト機能（黄色）、右シフト機能（青色）です。シフト機能名は、それぞれのキーの上部に黄色で、下部に青色で印刷されています。適切なシフトキー（ または ）を押してから目的のキーを押すと、シフト機能が利用できます。例えば、電卓の電源をオフにするには、 シフトキーを押して離れた後、 を押します。

 または  を押すと、ディスプレイ上部の対応する  または  インジケータ記号がオンになります。このインジケータは、次のキーを押すまでオンのままになっています。シフトキーをキャンセルする（インジケータをオフにする）には、同じシフトキーをもう一度押します。

1-2 はじめに

アルファベットキー



上図のように、ほとんどのキーの右下に英字が表示されています。文字入力（例えば、変数やプログラムラベル）が必要になると、ディスプレイに **A..Z** インジケータが表示され、アルファベットキーが「アクティブ」であることを示します。

変数については第 3 章を、ラベルについては第 13 章を参照してください。

カーソルキー

4 つのカーソル方向キーにはそれぞれ矢印が付けられています。本テキストでは、カーソルキーは 、、、 で表わされます。

バックスペースとクリア

まず、入力のクリア方法、数値の訂正方法、ディスプレイの表示を全部クリアして最初からやり直す方法について説明します。

クリア用のキー

キー	説明
	バックスペース。 式の入力途中で  を押すと、入力カーソル (<code>_</code>) の左側の文字が消去されます。一方、完成した式や 2 行目に表示された計算結果では、 を押すと結果がゼロで置換されます。また、 はエラーメッセージのクリアやメニューの終了にも使用します。 は、電卓がプログラム入力モードと方程式入力モードのときには、以下のように機能します。 ■ 方程式入力モード： 方程式の入力途中か編集集中であれば、 は挿入カーソルの左側の文字を消去します。それ以外の場合、つまり、方程式の入力が完了して挿入カーソルが表示されていない場合、 は方程式全体を消去します。 ■ プログラム入力モード： プログラム行の入力途中か編集集中であれば、 は挿入カーソルの左側の文字を消去します。それ以外の場合、つまり、プログラム行の入力が完了している場合、 は行全体を消去します。
	クリアまたはキャンセル。 表示されている数値がゼロにクリアされるか、現在の状況（メニュー、メッセージ、プロンプト、カタログ、方程式入力モード、プログラム入力モードなど）がキャンセルされます。

クリア用のキー（続き）

キー	説明
 CLEAR	クリアメニュー (× VARS ALL Σ STK CLVAR×) には、x (X レジスタの数値)、すべての直接変数、すべてのメモリ、すべての統計データ、すべてのスタック、間接変数を削除するためのオプションがあります。 [3] (3ALL) を押した場合、新しいメニュー CLR ALL? Y N が表示されるので、メモリのすべての内容を消去する前に判断を確認できます。 プログラム入力の場合は、3ALL の代わりに 3PGM と表示されます。[3] (3PGM) を押した場合、新しいメニュー CLR PGMS? Y N が表示されるので、すべてのプログラムを消去する前に判断を確認できます。 方程式入力の場合は、3ALL の代わりに 3EQN と表示されます。[3] (3EQN) を押した場合、CLR EQN? Y N メニューが表示されるので、すべての方程式を消去する前に判断を確認できます。 [6] (CLVAR×) を選択すると、このコマンドが 3 個のプレースホルダを持つコマンド行にペーストされます。プレースホルダの空白に 3 桁の数字を入力する必要があります。アドレスが入力したアドレスより大きい間接変数は、すべて消去されます。 例：CLVAR056 は、アドレスが 56 より大きい間接変数をすべて消去します。

メニューの使用

HP 35s には、キーボードに表示された機能よりも多くの機能が用意されています。16 個のキーは、メニューキーであるからです。これら 16 のメニューから、さらに多くの機能、または多くの機能のオプションにアクセスできます。

HP 35s のメニュー

メニュー名	メニュー説明	章
	数値関数	
L.R.	$\hat{x} \hat{y} r m b$ 線形回帰：カーブフィッティングと線形推定。	12
$\bar{x}, \bar{y}$	$\bar{x} \bar{y} \bar{x}w$ 統計 x 値と y 値の相加平均、統計 x 値の加重平均。	12
s, σ	$sx sy \sigma x \sigma y$ 標本標準偏差と母標準偏差。	12
CONST	41 個の物理定数の値にアクセスするためのメニュー。 「物理定数」(4-8 ページ) を参照。	4
SUMS	$n \Sigma x \Sigma y \Sigma x^2 \Sigma y^2 \Sigma xy$ 統計データの合算。	12
BASE	DEC HEX OCT BIN $d h o b$ 基数変換 (10 進数、16 進数、8 進数、2 進数)。	12
INTG	SGN INT $\div$ Rmdr INTG FP IP 符号付きの値、整数の除算、剰余、最大整数、小数部、整数部。	4、C
LOGIC	AND XOR OR NOT NAND NOR 論理演算子	11

プログラミング手順		
FLAGS	SF CF FS? フラグを設定、クリア、テストするための機能。	14
x?y	$\neq \leq < > \geq =$ X レジスタと Y レジスタの比較テスト。	14
x?0	$\neq \leq < > \geq =$ X レジスタと 0 の比較テスト。	14
その他の機能		
MEM	VAR PGM メモリステータス（空きメモリのバイト数）、変数のカタログ、プログラム（プログラムラベル）のカタログ。	1, 3, 12
MODE	DEG RAD GRAD ALG RPN 角度モードと演算モード。	4, 1
DISPLAY	FIX SCI ENG ALL. , 1,000 1000 x <i>i</i> y x+y <i>i</i> rθa 固定、科学、技術、完全浮動小数点表記、基数記号オプション（. または ,）、複素数表記（RPN モードでは xiy と rθa のみ利用可能）。	1
R↓ R↑	X Y Z T ALG モードでスタック（X、Y、Z、T レジスタ）を確認する機能	C
CLEAR	メモリのさまざまな部分をクリアするための機能。 表の  CLEAR (1-5 ページ) を参照	1, 3, 6, 12

メニュー機能を使用するには、以下の操作を実行します。

1. メニューキーを押して、メニュー項目のセットを表示します。
2.     を押して、選択する項目にアンダーラインを移動します。
3. 項目にアンダーラインが付いたら、 を押します。

メニューの項目に番号が付いていれば、アンダーラインを付けて  を押す代わりに、項目の番号を入力することもできます。

CONST、SUMS など、一部のメニューは複数のページから構成されます。こうしたメニューを選択すると、▲ または ▼ インジケータがオンになります。これらのメニューでは、> および < カーソルキーを使用して、現在のメニューページ上の項目に移動し、▽ および △ キーを使用して、メニューページ間を移動します。

例：

この例では、DISPLAY メニューを使って数値表示を小数点以下 4 桁に固定し、 $6 \div 7$ を計算します。DISPLAY メニューを使用して例を閉じると、完全浮動小数点表記に戻ります。

キー：	ディスプレイ：	説明：
	0	初期表示
	0	
DISPLAY	1FIX 2SCI	DISPLAY メニューに入ります
	3ENG 4ALL	
1 または ENTER	FIX _____	FIX コマンドが第 2 行にペーストされます
4	0.0000	小数点以下 4 桁に固定します
	0.0000	
6 ENTER 7 ÷	0.0000	除算を実行します
	0.8571	
DISPLAY 4	0	フル精度に戻ります
	8.57142857143E-	

メニューを使うことで、さまざまな機能を簡単に選択して実行することができます。電卓に内蔵された関数の名前を覚えたり、キーボードから探す手間が省けます。

メニューの終了

メニュー機能を実行すると、上の例で示すように、メニュー表示は自動的に消えます。機能を実行しないでメニューを終了するには、次の 3 つの方法があります。

-  を 1 レベルずつ押して、2 レベルの CLEAR または MEM メニューから戻る。表の  CLEAR (1-5 ページ) を参照してください。
-  または  を押して、他のすべてのメニューをキャンセルする

キー:

ディスプレイ:

123.5678_



1FIX 2SCI

3ENG 4ALL



 または 

123.5678_

- その他のメニューキーを押して、今のメニューから新しいメニューに移動する

キー:

ディスプレイ:

123.5678_Σ



1FIX 2SCI

3ENG 4ALL



1X 2VARS

3ALL 4Σ





123.5678

RPN モードと ALG モード

電卓を、RPN（逆ポーランド記法）モードまたは ALG（代数）モードで算術演算を実行するよう設定できます。

逆ポーランド記法（RPN）モードの場合、計算の中間結果が自動的に格納されるため、括弧を使用する必要はありません。

代数モード（ALG）では、通常の演算順序で算術演算を実行します。

RPN モードを選択するには：

MODE **5** (5RPN) を押して、電卓を RPN モードに設定します。電卓が RPN モードにある場合、**RPN** インジケータがオンになります。

ALG モードを選択するには：

MODE **4** (4ALG) を押して、電卓を ALG モードに設定します。電卓が ALG モードにある場合、**ALG** インジケータがオンになります。

例：

1 + 2 = 3 を計算するとします。

RPN モードでは、最初の数値を入力して **ENTER** キーを押し、2 番目の数値を入力して、最後に算術計算演算子キーを押します：**+**。

ALG モードでは、最初の数値を入力して **+** を押し、2 番目の数値を入力して、最後に **ENTER** キーを押します。

RPN モード	ALG モード
1 ENTER 2 +	1 + 2 ENTER

ALG モードでは計算結果と計算が表示されます。RPN モードでは結果のみが表示され、計算は表示されません。

注



計算には、ALG（代数）モードまたは RPN（逆ポーランド記法）のいずれかを選択できます。このマニュアルでは、余白の“✓”は、例またはキーストロークが RPN モードのもので、ALG モードでは実行方法が異なることを示しています。付録 C で、ALG モードでの電卓の使用方法を説明します。

Undo キー

Undo キー

Undo キーの動作は電卓のコンテキストによって異なりますが、通常は、任意の操作を元に戻すのではなく、エントリの削除を復元します。数値関数が実行された後、ディスプレイの第 2 行のデータを復元する方法の詳細については、第 2 章 **Last X レジスタ** を参照してください。 **UNDO** を  や  の使用後にすぐに押すと、以下の項目が復元されます。

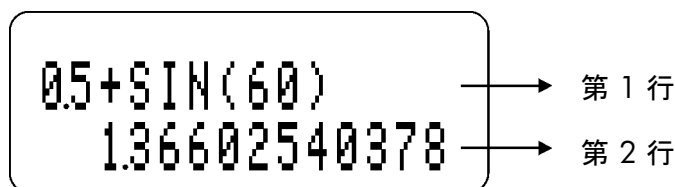
- 削除した入力
- 方程式モードで削除した方程式
- プログラムモードで削除したプログラム行

さらに、Undo は CLEAR メニューでクリアされたレジスタの値を復元させることもできます。Undo 操作は、削除操作の直後でなければなりません。他の操作が入ると、削除されたオブジェクトを Undo で復元できなくなります。Undo は、削除後に入力全体を復元する場合だけでなく、入力の編集集中にも使用できます。 **UNDO** を編集集中に押すと、以下の項目が復元されます。

-  で削除した、式の中の数値
-  でクリアした、編集途中の式
- 方程式モードやプログラムモードにおいて  で削除した、方程式やプログラムの中の文字

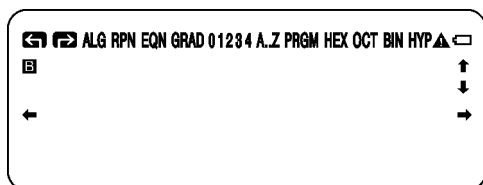
Undo 操作は、空きメモリ容量によっても制限されます。

ディスプレイとインジケータ



ディスプレイは、2 行の表示行と**インジケータ**から構成されています。

1 行に入力する文字が 14 文字を超えると、左にスクロールします。入力データは、ALG モードでは第 1 行、RPN モードでは第 2 行に表示されます。すべての計算は、E 符号（指数表記）を含めて 14 桁まで表示され、指数値は最大 3 桁です。



インジケータ

上図に示すディスプレイの記号を**インジケータ**と呼びます。それぞれの表示には、特定の意味があります。

HP 35s のインジケータ

インジケータ	意味	章
B	" B (Busy)" インジケータは、操作、方程式、プログラムの実行中に表示されます。	5
▲	分数表示モード ( FDISP を押す) の	
▼	場合、" ▲ " または " ▼ "、すなわち " ▲▼ "	
	インジケータの半分だけがオンになります。このインジケータは、表示された分子が、 真 の値よりもわずかに小さいか、大きいかを示しています。" ▲▼ " のどちらの部分もオンになっていない場合は、分数の 厳密 値が表示されています。	
	左シフトがアクティブになっています。	1
	右シフトがアクティブになっています。	1
RPN	逆ポーランド記法モードがアクティブになっています。	1, 2
ALG	代数モードがアクティブになっています。	1、C
PRGM	プログラム入力がアクティブになっています。	13
EQN	方程式入力モードがアクティブになっているか、電卓が、方程式を評価中か、または方程式を実行中です。	6
0 1 2 3 4	どのフラグがセットされているかを示します (フラグ 5 ~ 11 にはインジケータがありません)。	14
RAD または GRAD	ラジアンまたはグラジアン角度モードが設定されています。DEG モード (デフォルト) にはインジケータがありません。	4
HEX OCT BIN	アクティブな数値の基数を示します。DEC (基数 10、デフォルト) にはインジケータがありません。	11
HYP	双曲線関数がアクティブになっています。	4、C

HP 35s のインジケータ（続き）

インジケータ	意味	章
◀, ▶	ディスプレイの第 1 行または第 2 行の入力表示には、左または右にさらに文字があります。これらのインジケータが両方同時に表示される場合、入力表示の左右に表示されていない文字が存在します。第 1 行の入力に表示されていない文字があるときは、省略記号 (...) が表示されます。RPN モードでは、  キーと  キーを使って入力をスクロールし、先頭文字と終了文字を表示します。ALG モードでは、   と   を使って残りの文字を表示します。	1, 6
↑, ↓	方程式リスト、変数カタログ、プログラム行、メニューページ、プログラムカタログ内のプログラムを移動するための、  キーと  キーがアクティブになっています。	1, 6, 13
A..Z	アルファベットキーがアクティブになっています。	3
▲	注意！ 特別な状態またはエラーを示します。	1
◻	電池容量が低下しています。	A

数値の入力

電卓が処理可能な最小値／最大値は、 $\pm 9.999999999999^{499}$ です。計算結果がこの範囲を超えると、エラーメッセージ“OVERFLOW”が、▲インジケータと一緒に一瞬だけ表示されます。その後このメッセージは、電卓が表示可能な、オーバーフロー限界に一番近い値に置き換わります。電卓がゼロと区別できる最小値は、 $\pm 10^{-499}$ です。この範囲内の数値を入力すると、ディスプレイにゼロが表示されます。同様に、計算結果がこの範囲にある場合、結果はゼロと表示されます。上記の最大範囲を超える数値を入力すると、エラーメッセージ“INVALID DATA”が表示され、エラーメッセージをクリアしたときに前の入力に戻るので、値を修正できます。

数値を負の値に変更する



 キーで数値の符号を変更できます。

- 負の数値を入力するには、数値を入力して  を押します。
- ALG モードでは、数値の入力前または入力後に  キーを押します。
- 前に入力した数値の符号を変更するには、単に  キーを押します（数値に指数が含まれる場合、 は、**仮数**、すなわち数値の指数でない部分にのみ影響します）。

10 の指数

ディスプレイで表示される指数

10 の累乗で明示的に表される数値（ 4.2×10^{-5} など）は、指数 10 の前に E を付けて表示されます。 4.2×10^{-5} と入力すると、4.2E-5 と表示されます。

数値は、大きさが表示可能な範囲を超えると、指数表記に自動的に変更されます。

例えば、FIX 4（小数点以下 4 桁）の表示形式で、次のキー入力の効果を確認します。

キー:

0 **.** **0** **0**

0 **0** **6** **2**

ENTER

ディスプレイ:

0.000062_

0.0001

説明:

入力中の数値を示します。

数値を表示形式に適合するよう丸めます。

科学表記にしないと有効数字が表示されないため、自動的に科学表記を使用します。

0 **.** **0** **0**

4.2000E-5

0 **0** **4** **2**

ENTER

10 の指数の入力

10 の指数をすばやく入力するには、**[E]** キーを使用します。例えば、100 万を 1000000 と入力する代わりに、**[1][E][6]** を入力できます。以下の例で、プロセスと結果の表示方法を示します。

例:

プランク定数を入力するとします。 6.6261×10^{-34}

キー:

6 **.** **6** **2** **6** **1**

[1]

[E]

ディスプレイ:

0

6.6261_

0

6.621E_

6.621E-34

6.621E-34

説明

仮数を入力します

$\times 10^x$ と等価です

指数を入力します

[3][4][+/-][ENTER]

乗数のない 10 の累乗の場合、前述の「百万」のように、**[1][E]** キーの後に目的の指数を押します。

その他の指数関数

10 の指数 (10 を底とする真数) を計算するには、**[$\square$][10^x]** を使用します。

任意の数値の累乗 (指数) の結果を計算するには、**[y^x]** を使用します (第 4 章を参照)。

1-16 はじめに

入力カーソルについて

数値を入力すると、ディスプレイにカーソル (**_**) が現れて点滅します。カーソルは、次の数字がどこに入るかを示します。すなわち、数値が完了していないことを示しています。

キー：

ディスプレイ：

説明：

1 **2** **3**

123_

入力が完了していません。数値は未完成です。

関数を実行して結果を計算すると、数値が完成し、入力が完了するため、カーソルが消えます。

✓ **$\sqrt{x}$**

11.0905

入力が完了しています。

[ENTER] キーを押すと入力が完了します。2 つの数値を分離するには、最初の数値を入力し、**[ENTER]** を押して入力を完了してから、2 番目の数値を入力します。

✓ **1** **2** **3** **[ENTER]**

123.0000

完成した数値です。

✓ **4** **+**

127.0000

もう 1 つの完成した数値です。

入力が完了していない（カーソルが表示されている）ときは、バックスペース **[←]** で最後の数字を削除できます。入力が完了する（カーソルが消える）と、**[←]** は、**[C]** と同様に機能し、数値全体をクリアします。試してみてください。

数値の範囲とオーバーフロー

電卓で使用可能な最小値は $-9.9999999999 \times 10^{49}$ で、最大値は $9.9999999999 \times 10^{49}$ です。

- 計算結果がこの範囲を超えると、値は、 $-9.9999999999 \times 10^{49}$ または $9.9999999999 \times 10^{49}$ となり、警告メッセージ **OVERFLOW** が表示されます。

算術計算の実行

HP 35s は、RPN モードまたは代数モード (ALG) で動作します。それぞれのモードにより、式の入力方法が異なります。次のセクションでは、単一引数 (単項) 演算の場合と 2 つの引数の (2 項) 演算の場合の入力の違いについて説明します。

単一引数 (単項) 演算

$1/x$ 、 x^2 、 LN 、 SIN などの HP 35s の一部の数値演算は、単一の数値入力を必要とします。これらの単一引数演算は、RPN モードか ALG モードかによって入力方法が異なります。RPN モードでは、まず値を入力し、次に演算を適用します。数値を入力してから ENTER キーを押すと、ディスプレイの第 1 行に入力した数値が、第 2 行に計算結果が表示されます。そうでないと、第 2 行に計算結果が表示されるだけで、第 1 行が変化しません。ALG モードでは、最初に演算子を押すと、ディスプレイに関数と括弧が表示されます。括弧内に数値を入力し、 ENTER キーを押します。第 1 行に式が、第 2 行に計算結果が表示されます。次の例で相違点を説明します。

例：

3.4^2 を最初に RPN モードで、次に ALG モードで計算します。

キー：	ディスプレイ：	説明：
MODE 5 (5RPN)		RPN モードに入ります (必要な場合)
3 $\cdot$ 4	0 3.4	数値を入力します
R x^2	0 11.56	2 乗演算子を押します
MODE 4 (4ALG)		ALG モードに変更します
R x^2	SQ(<)	2 乗演算を入力します
3 $\cdot$ 4	SQ(3.4)	括弧の間に数値を挿入します
ENTER	SQ(3.4)	ENTER キーを押して結果を表示します
	11.56	

この例で、2乗演算子はキーに x^2 と表示されていますが、ディスプレイでは SQ() となります。ALG モードには、ディスプレイの表示とキーボードの表示（および RPN モードでの表示）が異なる単一引数演算子が複数あります。このような演算子を下の表にまとめます。

キー	RPN、RPN プログラム	ALG、方程式、ALG プログラム
x^2	x^2	SQ()
$\sqrt{x}$	$\sqrt{x}$	SQRT()
e^x	e^x	EXP()
10^x	10^x	ALOG()
$1/x$	$1/x$	INV()

2つの引数をとる（2項）演算

$+$ 、 $\div$ 、 y^x 、 nCr などの2つの引数をとる演算も、モードによって入力方法が異なりますが、違いは単一引数演算子の場合とほぼ同じです。RPN モードでは、最初の数値を入力してから2番目の数値を x レジスタに格納し、2つの引数をとる演算を呼び出します。ALG モードでは、伝統的な中置記法と、より関数指向の手法をとる別の方法の、2種類あります。相違点を次の例で説明します。

例

2+3 と ${}_6C_4$ を最初に RPN モードで、次に ALG モードで計算します。

キー：	ディスプレイ：	説明：
MODE 5 (5RPN)		RPN モードに変更します（必要な場合）
2 ENTER 3	2 3_	2 を入力し、3 を x レジスタに格納します。3 の後にカーソルが点滅しています。ENTER を押さないでください！
+	0 5 6	加算キーを押して結果を表示します。
6 ENTER 4	4_	6 を入力し、4 を x レジスタに格納します。
↵ nCr	5_ 15	組合せキーを押して結果を表示します。

MODE 4 (4ALG)
2 + 3 ENTER

2+3

ALG モードに変更します
式と結果が表示されます。

5

◀ nCr

nCr(,)

組合せ関数を入力します。

6 > 4

nCr(6,4)

6 を入力し、編集カーソルをカンマ
を超えて移動し、4 を入力します。

ENTER

nCr(6,4)

ENTER キーを押して結果を表示し
ます。

15

ALG モードで、中置記法演算子は $+$ 、 $-$ 、 $\times$ 、 $\div$ 、 y^x です。2 つの引数をとるその他の演算は、 $f(x,y)$ という形式の関数表記を使用します。x は第 1 オペランド、y は第 2 オペランドです。RPN モードでは、2 つの引数をとる演算のオペランドを、スタックで Y、X の順番で入力します。つまり、y は y レジスタの値、x は x レジスタの値となります。

y の x 乗根 ($\sqrt[x]{y}$) は、このルールの例外です。例えば RPN モードで $\sqrt[3]{8}$ を計算するには、8 ENTER 3 ◀ $\sqrt[x]{y}$ を押します。ALG モードでの同等の演算の入力は、◀ $\sqrt[x]{y}$ 3 > 8 ENTER となります。

単一引数演算の場合と同様に、2 つの引数をとる演算のいくつかは、RPN モードと ALG モードで表示が異なります。相違点を下の表にまとめます。

キー	RPN、RPN プログラム	ALG、方程式、ALG プログラム
y^x	y^x	$^$
$\sqrt[x]{y}$	$x \sqrt{} y$	XROOT(,)
INT $\div$	INT $\div$	IDIV(,)

$+$ 、 $\times$ などの可換的演算の場合、オペランドの順番は計算結果に影響しません。非可換的な 2 つの引数をとる演算で、オペランドの入力の順番を間違えた場合、RPN モードであれば $\leftrightarrow$ キーを押すだけで x レジスタと y レジスタの中身を交換できます。これについては、第 2 章で詳しく説明します（「スタックの X レジスタと Y レジスタを交換」を参照してください）。

1-20 はじめに

表示形式の制御

すべての数値は 12 桁の精度で格納されますが、数値の表示桁数は Display メニューのオプションで制御できます。  を押してこのメニューにアクセスします。最初の 4 つのオプション (FIX、SCI、ENG、ALL) が、数値表示の桁数を制御します。内部で複雑な計算が行われるときは、中間結果が 15 桁の精度になることもあります。表示する数値を表示形式に従って丸めます。

固定小数点表記 (FIX)

FIX 表記は、小数点以下最大 11 桁 ("." または " " 基数点の 右に 11 桁) の数値を表示します (適合する場合)。プロンプト FIX に続けて、表示する小数点以下の桁数を入力します。10 桁または 11 桁の場合はそれぞれ、  または   を押します。

例えば数値 123.4567889 では、"7"、"0"、"8"、"9" が、電卓を FIX 4 表示モードに設定したときに表示される小数点以下の桁です。

現在の小数点以下桁数の設定で表示するには大きすぎる (10^{11}) か小さすぎる (10^{-11}) 数値は、自動的に科学表記で表示されます。

科学表記 (SCI)

SCI 表記は、数値を科学的記数法 ("." または " " 基数点の前 1 桁) で表示します。小数点以下は最大 11 桁、指数は最大 3 桁です。プロンプト SCI に続けて、表示する小数点以下の桁数を入力します。10 桁または 11 桁の場合はそれぞれ、  または   を押します (数値の仮数部は常に 10 未満です)。

例えば数値 $1.2346E5$ では、"2"、"3"、"4"、"6" が、電卓を SCI 4 表示モードに設定したときに表示される小数点以下の桁です。"E" の次の "5" は 10 の指数を表わし、 1.2346×10^5 を意味します。

12 桁を超える数値を入力または計算する場合、12 桁を超える桁の精度は維持されません。

工学表記 (ENG)

ENG 表記は、数値を科学的記数法と同様の方法で表示しますが、指数が 3 の倍数になるという点が異なります ("." または "," 基数点の前に最大 3 桁存在します)。この表記は、 10^3 の倍数で指定されている単位 (マイクロ単位、ミリ単位、キロ単位など) を使用する、科学計算や工学計算に有効です。

プロンプト ENG に続けて、最初の有効数字に続く桁数を入力します。10 桁または 11 桁の場合はそれぞれ、 または  を押します。

例えば数値 $123.46E3$ では、"2"、"3"、"4"、"6" が、電卓を ENG 4 表示モードに設定したときに表示される、最初の有効数字の後の有効数字です。"E" の次の "3" は 10 の指数 (3 の倍数) を表わし、 123.46×10^3 を意味します。

  ENG または   ENG  を押すと、表示されている数値の指数表示が 3 の倍数に変化し、仮数もそれに従って調整されます。

例：

この例では、数値 $12.346E4$ を使って工学表記の動作を示します。   ENG と   ENG  機能の使用法も示します。例では RPN モードを使用します。

キー：	ディスプレイ：	説明：
  	ENG_	工学表記を選択します
(3ENG)		
	0.0000E0 0.0000E0	4 を入力します (第 1 桁に続く有効桁が 4 の場合)
     	123.46E3	12.346E4 を入力します
  	123.46E3	
  ENG or	123.46E3	
  ENG 	123.46E3	
  ENG	123.46E3	指数を 3 の倍数で増加します
  ENG 	0.12346E6 123.46E3	指数を 3 の倍数で減少します
	123.46E3	

全表記 (ALL)

全表記はデフォルト設定です。数値を最大 12 桁の精度で表示します。全桁がディスプレイに収まらないときは、数値が自動的に科学表記で表示されます。

数値内のピリオドとカンマ (・) (,)

HP 35s では、数値を読みやすくするためピリオドとカンマを使用します。小数点 (基数) としてピリオドまたはカンマを選択できます。位取り記号を使用して、個々の数字を 3 個単位のグループに分けるかどうか也可以选择できます。次の例でオプションを示します。

例

数値 12,345,678.90 を入力し、小数点をカンマに変更します。次に位取り記号を消してみます。最後に、デフォルト設定に戻します。例では RPN モードを使用します。

キー:	ディスプレイ:	説明:
 DISPLAY 4		完全浮動小数点精度 (全表記) を選択します
(4ALL)		
1 2 3 4 5	12,345,678.9	・ デフォルト表記は、カンマを位取り記号、ピリオドを基数として使用します。
6 7 8 . 9	12,345,678.9	
ENTER		
 DISPLAY 6 (6,)	12,345,678,9 12,345,678,9	基数をカンマに変更します。位取りは自動的にピリオドになります。
 DISPLAY 8	12345678,9 12345678,9	カンマの位取り記号を消します。
(81000)		
 DISPLAY 5 (5.)	12,345,678.9	デフォルト形式に戻します。
 DISPLAY 7	12,345,678.9	
(71,000)		

複素数の表示形式 ($x+iy$ 、 $x+yi$ 、 $r\theta a$)

複素数は、さまざまな形式で表示できます。 $x+iy$ 、 $x+yi$ 、 $r\theta a$ 。ただし、 $x+yi$ は ALG モードでのみ利用できます。次の例では、複素数 $3+4i$ を 3 つの方法で表示します。

例

複素数 $3+4i$ をそれぞれの形式で表示します。

キー:	ディスプレイ:	説明:
MODE 4 (4ALG)		ALG モードを有効にします
3 i 4 ENTER	$3.i.4$	複素数を入力します。デフォルト形式の $3i4$ で表示されます。
↵ DISPLAY .	$3.i.4$	$x+yi$ 形式に変更します。
1 ($11x+yi$)	$3+4.i$	
↵ DISPLAY .	$3.i.4$	$r\theta a$ 形式に変更します。半径は
0 ($1\theta r\theta a$) or	$5\theta 53.13\theta 1\theta 23542$	5 で、角度は約 53.13° です。
↵ DISPLAY ↵		
↵ > ENTER		

フル 12 桁精度の表示

小数点以下の表示桁数を変更すると、ディスプレイの表示に影響しますが、数値の内部表現には影響しません。内部的に格納されているすべての数値は、常に 12 桁です。

例えば数値 14.8745632019 を入力したとき、表示モードが FIX 4 に設定されていると "14.8746" しか表示されませんが、残り 6 桁 ("632019") は電卓の内部に存在しています。

一時的に数値を全精度で表示するには、 **SHOW** を押します。これにより、**SHOW** を押しているあいだ、数値の **仮数**（指数以外）が表示されます。

キー：	ディスプレイ：	説明：
✓   ENTER  .	58.5000	小数点以下 4 桁が表示されます。
 		
 DISPLAY  (2SCI)	5.85E1	科学表記：小数点以下 2 桁と指数。
		
 DISPLAY  (3ENG)	58.5E0	工学表記。
		
 DISPLAY  (4ALL)	58.5	すべての有効数字。有効数字の後のゼロは省略されます。
 DISPLAY  (1FIX)	58.5000	小数点以下 4 桁が表示されます。
		指数はありません。
✓ 	0.0171	58.5 の逆数です。
 SHOW (hold)	170940170940	SHOW を離すまで全精度で表示します

分数

HP 35s を使って分数の入力や演算を行い、小数または分数で表示できます。分数は $a\ b/c$ の形式で表示されます。 a は整数、 b と c は自然数です。 b は $0 \leq b < c$ という条件を、 c は $1 < c \leq 4095$ を満たすものとします。

分数の入力

分数は、次の方法でいつでもスタックに入力できます。

1. 整数部を入力し、 を押します（最初の  は、数値の整数部と分数部を分離します）。
2. 分子を入力し、再度  を押します。2 番目の  は、分子と分母を分離します。
3. 分母を入力し、 または関数キーを押して、数字の入力を完了します。数値または計算結果が、現在の表示形式に従ってフォーマットされます。

 キーの下に $a\ b/c$ 記号は、 キーが分数の入力に 2 回使用されることに対する注意喚起です。

次の例で分数の入力と表示を示します。

例

帯分数 $12\frac{3}{8}$ を入力し、分数形式と小数形式で表示します。 $\frac{3}{4}$ を入力し、 $12\frac{3}{8}$ に加算します。例では RPN モードを使用します。

キー:	ディスプレイ:	説明:
   	0 12.3	小数点が、通常通り解釈されます。
 	0.0000 $12\frac{3}{8}$ _	 を 2 度目に押すと、ディスプレイが分数モードに戻ります。
	12.3750 12.3750	入力された数値が、現在の表示形式を使って表示されます。
	$12\frac{3}{8}$ $12\frac{3}{8}$	分数表示モードに変更します。
   	$12\frac{3}{8}$ $0\frac{3}{4}$ _	$\frac{3}{4}$ を入力します。整数部が無い場合、最初の入力は  です ($0\frac{3}{4}$ と入力することもできます)。
	0 $13\frac{1}{8}$	$12\frac{3}{8}$ に $\frac{3}{4}$ を加算します。
	0 13.1250	現在の表示モードに戻します。

分数の使用方法の詳細については、第 5 章「分数」を参照してください。

メッセージ

電卓は、エラー状態が発生すると **▲** インジケータを表示します。通常、エラーインジケータと一緒にメッセージも表示されます。

- メッセージをクリアするには、**[C]** または **[←]** を押します。RPN モードではエラーの発生前のスタックに戻ります。ALG モードでは最後の式に戻り、エラーを修正できるよう、エラーの位置に編集カーソルが移動します。
- その他のキーでもメッセージはクリアされますが、キーの機能は動作しません。

メッセージが表示されず、**▲** インジケータだけが現れる場合は、非アクティブまたは無効なキーを押しています。例えば、**[.]** **[.]** を押すと、2 番目の小数点はこの状況では意味がないため、**▲** が表示されます。

すべての表示メッセージについては、付録 F「メッセージ」を参照してください。

電卓のメモリ

HP 35s には 30KB のメモリが搭載されており、データの任意の組合せ（変数、方程式、プログラム行）を格納できます。

メモリの空き容量を調べる

[◀] [MEM] を押すと、次のメニューが表示されます。

1VAR 2 PGM
nnn mm,mmm

ここで、

nnn は、使用される間接変数の量です。

mm,mmm は、利用できるメモリのバイト数です。

1 (1VAR) を押すと、直接変数のカタログが表示されます（第 3 章「VAR カタログによる変数のレビュー」を参照してください）。**2** (2PGM) を押すと、プログラムのカタログが表示されます。

1. 変数のカタログを入力するには、**1** (1VAR) を押します。プログラムのカタログを入力するには、**2** (2PGM) を押します。
2. カタログを確認するには、**✓** または **△** を押します。
3. 変数またはプログラムを削除するには、カタログで表示中に **✖** **CLEAR** を押します。
4. カタログを終了するには、**C** を押します。

すべてのメモリをクリアする

すべてのメモリをクリアすると、格納した数値、方程式、プログラムがすべて消去されます。モードと表示形式の設定には影響しません（設定とデータをクリアするには、付録 B「メモリのクリア」を参照してください）。

すべてのメモリをクリアするには：

1. **4** (4ALL) を押します。確認のプロンプト CLR ALL? Y N が表示されます。これにより、不注意によるメモリのクリアを防止しています。
2. **<** (Y) **ENTER** を押します。

RPN：自動メモリスタック

この章では、RPN モードの場合、計算が自動メモリスタックでどのように行われているかを説明します。**電卓を使用するだけであれば、この章の内容を理解する必要はありません**が、内容を理解することで、特にプログラミングの際に電卓の利用方法の幅が大きく広がります。

第 2 部「プログラミング」で、プログラムのデータの処理と整理にスタックが果たす役割について説明しています。

スタックとは

HP 35s が括弧を使用することなく、複雑な計算を容易に処理できるのは、**中間結果の自動格納**のおかげです。自動格納の鍵となるのが、**自動 RPN メモリスタック**です。

HP の演算論理は、ポーランドの論理学者 Jan Łukasiewicz（1878 – 1956）により開発された、明快で**括弧を必要としない**数学的論理「ポーランド記法」を基にしています。

従来の代数記法では、関連する数値や変数の**間**に演算子を配置するのに対し、Łukasiewicz の記法では、数値や変数の**前**に演算子を配置します。スタックの効率を最善にするため、記法を変更して、数値の**後**に演算子を指定するようにしました。このため、**逆ポーランド記法**、RPN と呼ばれます。

スタックは、相互に「積み重ね」られた、**レジスタ**と呼ばれる 4 つの格納場所から構成されます。これらのレジスタは、X、Y、Z、T と名前が付けられ、現在の 4 つの数値を格納し、操作します。「最も古い」数値は、T（**最上位**）レジスタに格納されています。スタックが、計算の作業領域です。

T	第 3 部	「最 も 古 い」 数 値
	第 2 部	
	第 1 部 0.0000	
Z	第 3 部	
	第 2 部	
	第 1 部 0.0000	
Y	第 3 部	表 示
	第 2 部	
	第 1 部 0.0000	
X	第 3 部	表 示
	第 2 部	
	第 1 部 0.0000	

最も「最近の」数値は、X レジスタに格納されています。**これが、ディスプレイの第 2 行に表示される数値です。**

それぞれのレジスタは、次の 3 つの部分に分かれています。

- 実数または 1 次元ベクトルは、第 1 部を占有します。この場合、第 2 部と第 3 部は null 値になります。
- 複素数または 2 次元ベクトルは、第 1 部と第 2 部を占有します。この場合、第 3 部は null 値になります。
- 3 次元ベクトルは、第 1 部、第 2 部、第 3 部を占有します。

プログラミングでは、計算の実行、中間結果の一時的な格納、プログラムやサブルーチン間での格納データ（変数）の受け渡し、入力の受け入れ、出力の供給にスタックが用いられます。

2-2 RPN : 自動メモリスタック

X レジスタと Y レジスタのディスプレイ表示

メニュー、メッセージ、方程式行、またはプログラム行が表示されている場合を除いて、X レジスタと Y レジスタが表示されます。一部の機能名には、x または y が含まれます。

これは偶然ではありません。これらの文字は、X レジスタと Y レジスタを表わしています。例えば、  を押すと、10 を X レジスタの数値で乗じた値が得られます。

X レジスタのクリア

  (×) を押すと、常に X レジスタがゼロにクリアされます。命令をプログラムする際にも使用します。 キーの動作はこれと対照的に、状況に依存します。状況に応じて、現在の表示をクリアまたはキャンセルします。X レジスタが表示されているときにのみ、  (×) と同様に動作します。

 も、X レジスタが表示されており、かつ数字入力完了していれば（カーソルがない状態であれば）、  (×) と同様に動作します。

スタックの確認

R↓（ロールダウン）

（ロールダウン）キーを使用すると、1 度に 1 レジスタずつ、内容を下向きに「ロール」することで、スタック全体の内容を確認できます。x レジスタと y レジスタをロールするときに数値を表示できます。

1、2、3、4（       と押す）が入力されているスタックを考えます。 を 4 回押すことで、スタックが一巡し、元の位置に戻ります。

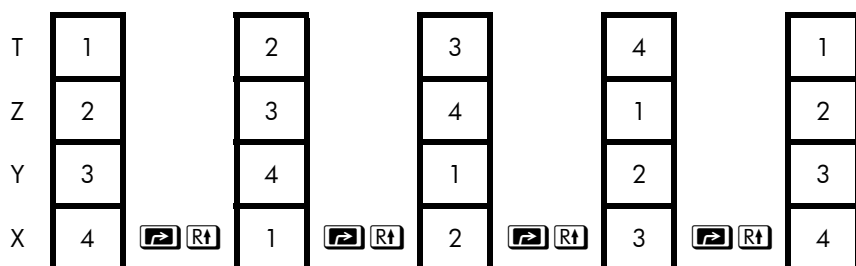
T	1		4		3		2		1
Z	2		1		4		3		2
Y	3		2		1		4		3
X	4		3		2		1		4

Xレジスタの内容がTレジスタへ、Tレジスタの内容がZレジスタへというように循環します。レジスタ自体の位置は保持され、レジスタの内容だけがロールする点、およびXレジスタとYレジスタの内容のみが表示される点に注意してください。

R↑ (ロールアップ)

 **R↑** (ロールアップ) キーの機能は **R↓** とほぼ同じですが、1度に1レジスタずつ、スタックの内容を上向きに「ロール」する点が異なります。

Xレジスタの内容がYレジスタへ、Tレジスタの内容がXレジスタへというように循環します。



スタックのXレジスタとYレジスタの交換

スタック内容を操作するその他のキーとして、**X↔Y** (xとyの交換) があります。このキーは、他のスタックに影響を与えることなくXレジスタとYレジスタの内容を入れ替えます。**X↔Y** を2回押すと、XレジスタとYレジスタの内容が元の順番に戻ります。

X↔Y 機能は主に、計算中に数値の順番を入れ替えるのに使用されます。

例えば、 $9 \div (13 \times 8)$ を計算する1つの方法として、

1 **3** **ENTER** **8** **×** **9** **X↔Y** **÷** を押します。

この式を左から右に計算するには、キーを次のように押します。

9 **ENTER** **1** **3** **ENTER** **8** **×** **÷** .

注



スタックに格納されている数値は、常に4個である点に注意してください。Tレジスタ（最上位レジスタ）の内容は、5番目の数値が入力されると失われます。

スタックでの四則計算の方法

新しい数値が X レジスタに入力されたり（**スタックの積み上げ**）、演算子が 2 つの数値を結合して X レジスタに新しい数値を生成する（**スタックの破棄**）と、スタックの内容が自動的に上下に移動します。

スタックに数値 1、2、3、4 が格納されている場合を考えます。計算中にスタックがその内容を破棄して、積み上げる様子を確認します。

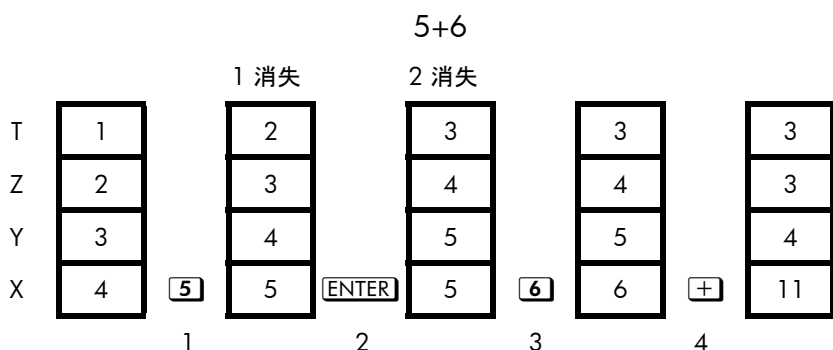
$$3+4-9$$

T	1	1	1	1
Z	2	1	2	1
Y	3	2	7	2
X	4	7	9	-2
	+	9	-	
	1	2	3	

1. スタックが、内容を「破棄」します。T（最上位）レジスタが、その内容を複製します。
 2. スタックが、その内容を「積み上げ」ます。T レジスタの内容が失われます。
 3. スタックが、破棄されます。
- スタックが積み上がる時、T（最上位）レジスタの内容が Z レジスタの内容に置き換わり、T レジスタの**以前**の内容が失われる点に注意してください。スタックのメモリ限界が 4 個の数値であることがわかります。
 - スタックは自動的に移動することから、新しい計算を実行する前に X レジスタをクリアする必要は**ありません**。
 - ほとんどの機能は、X レジスタに次の数値が入力されると、その内容を**積み上げるようスタックを準備しています**。スタックの上昇が発生しない演算の一覧については、「付録 B」を参照してください。

ENTER の動作

ENTER は、順々に入力された 2 つの数値を分離します。スタックの観点からは、どのようにこれが実現されるのでしょうか。再度、スタックに数値 1、2、3、4 が格納されている場合を考えます。2 個の新しい数値を入力して、加算します。



1. スタックを積み上げます。
2. スタックを積み上げ、X レジスタを複製します。
3. スタックを積み上げ**ません**。
4. スタックを破棄し、T レジスタを複製します。

ENTER は X レジスタの内容を複製して Y レジスタに上書きします。次に入力する（または呼び出す）数値は、X レジスタにある最初の数値のコピーを**上書き**します。これにより、順々に入力された 2 つの数値を分離します。

ENTER の複製効果を使用して、スタックをすばやくクリアできます。0

ENTER **ENTER** **ENTER** を押します。すべてのスタックレジスタがゼロになります。ただし、必ずしも計算の前にスタックをクリアする**必要**はありません。

同じ数値を 2 回使用する

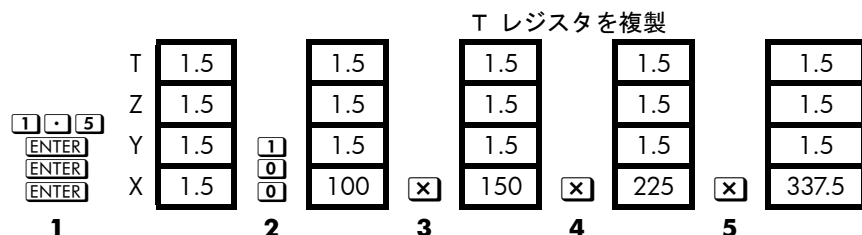
ENTER の複製機能を別の用途に使用できます。数値にその数値を加算するには、**ENTER** **+** を押します。

スタックに定数を入力する

[ENTER] の複製効果と、スタック破棄の (T から Z への) 複製効果を併用することで、計算のための定数をスタックに入力できます。

例：

あるバクテリア培養菌の増殖率を 1 日あたり 50% とすると、個体数 100 の場合、3 日目の終わりには個体数がいくらになっていますか？



1. スタックに増殖率を入力します。
2. 最初の個体数を入力します。
3. 1 日後の個体数を計算します。
4. 2 日後の個体数を計算します。
5. 3 日後の個体数を計算します。

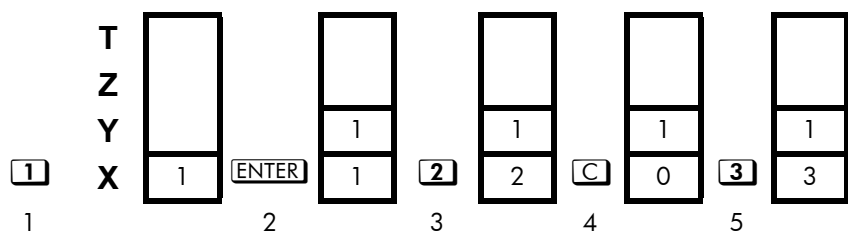
スタックのクリア方法

X レジスタをクリアすると、X レジスタにゼロが入ります。入力した（または呼び出した）次の数値が、このゼロを上書きします。

X レジスタの内容、すなわち **x** をクリアするには、4 つの方法があります。

1. **[C]** を押します。
2. **[←]** を押します。
3. **[ CLEAR]** **[1]** (1×) を押します（主にプログラムの入力中に使用されます）。
4. **[ CLEAR]** **[5]** (5STK) を押して、X レジスタ、Y レジスタ、Z レジスタ、T レジスタをゼロにクリアします。

例として、1 と 3 を入力しようとして誤って 1 と 2 を入力した場合の、エラーの修正方法を示します。



1. スタックを積み上げます。
2. スタックを積み上げ、X レジスタを複製します。
3. X レジスタを上書きします。
4. ゼロで上書きすることにより、x をクリアします。
5. x を上書きします（ゼロを置き換えます）。

最後の X レジスタ

LAST X レジスタはスタックの付随レジスタであり、最後の数値機能の実行前に X レジスタに記録されていた数値を保持します。（数値機能とは、 $\sqrt{x}$ のように、別の数値から結果を生成する演算です。） **LAST x** を押すと、この値が X レジスタに戻ります。

この「最後の X」値を取得する機能には、主に 2 つの用途があります。

1. エラーの訂正
2. 計算での数値の再使用

LAST X レジスタに x を格納する機能の一覧については、付録 B を参照してください。

LAST X を使ったエラーの訂正

誤った単一引数関数

誤った単一引数関数を実行した場合、 **LAST X** を使用して数値を取得し、正しい関数を実行できます。（不正確な結果をスタックからクリアする場合、 を最初に押します。）

  と  **%CHG** はスタックの破棄が発生しないため、これらの関数からも単一引数関数と同じ方法で回復できます。

例：

$4.7839 \times (3.879 \times 10^5)$ を計算し、その平方根を求めようとして、誤って  を押してしまった場合を考えます。この場合でも最初からやり直す必要はなく、 **LAST X**  を押すと正しい結果が得られます。

2 つの引数をとる関数を使用した場合のエラー

2 つの引数をとる演算 (、、 など) を使用したときの誤りは、 **LAST X** とその演算の逆の演算で訂正できます。

1.  **LAST X** を押して 2 番目の数値（操作の直前の x ）を回復します。
2. 逆の演算を行います。これにより元の 1 番目の数値が戻ります。2 番目の数値は、まだ LAST X レジスタ内にあります。その後：
 - **間違った関数**を使用していた場合は、再度  **LAST X** を押して元のスタック定数を復元します。正しい関数を実行します。
 - **間違った 2 番目の数値**を使用していた場合は、正しい数値を入力し、関数を実行します。

間違った 1 番目の数値を使用していた場合は、正しい 1 番目の数値を入力し、 **LAST X** を押して 2 番目の数値を回復してから、関数を再度実行します。（不正確な結果をスタックからクリアする場合、 を最初に押します。）

例：

計算中にミスを犯したと想定します。

$$16 \times 19 = 304$$

考えられるエラーには、次の 3 種類があります。

間違っただ計算：	誤り：	訂正：
1 6 ENTER 1	間違っただ関数	 LAST X +
9 =		 LAST X ×
1 5 ENTER 1	間違っただ 1 番目の数値	1 6  LAST X ×
9 ×		
1 6 ENTER 1	間違っただ 2 番目の数値	 LAST X ÷ 1 9 ×
8 ×		

LAST X による数値の再使用

 **LAST X** を使用して、計算で数値（定数など）を再使用できます。四則計算演算の実行直前に定数を入力します。これにより、その定数が X レジスタの最後の数値となるため、格納して、 **LAST X** で取得できます。

例：

$$\frac{96.704 + 52.3947}{52.3947} \text{ を計算します。}$$

	T	t		t		t
9 6	Z	z	5 2	z		t
• 7	Y	96.7040	• 3	96.7040		z
0 4	X	96.7040	9 4	52.3947	+	149.0987
ENTER			7			

LAST	X	/	/	+	52.3947
------	---	---	---	----------	---------

	T	t		t
	Z	z		t
	Y	149.0987		z
↶ LASTx	X	52.3947	÷	2.8457

LAST	X	52.3947	52.3947
------	---	---------	---------

キー:	ディスプレイ:	説明:
9 6 • 7 0 4	96.7040	1 番目の数値を入力します。
ENTER		
5 2 • 3 9 4	149.0987	中間結果。
7 +		
↶ LASTx	52.3947	+ 以前の表示に戻します。
÷	2.8457	最終結果。

例：

地球から近い距離にある 2 つの恒星は、リゲル・ケンタウルス（距離は 4.3 光年）とシリウス（距離は 8.7 光年）です。光の速度 c (9.5×10^{15} m/年) を使用して、これらの距離をメートルに変換してください。

リゲル・ケンタウルスまでの距離： $4.3 \text{ yr} \times (9.5 \times 10^{15} \text{ m/yr})$.

シリウスまでの距離： $8.7 \text{ yr} \times (9.5 \times 10^{15} \text{ m/yr})$.

キー：	ディスプレイ：	説明：
[4] [.] [3] [ENTER]	4.3000	リゲル・ケンタウルスまでの光年。
[9] [.] [5] [E] [1] [5]	9.5E15_	光速、 c 。
[x]	4.0850E16	リゲル・ケンタウルスまでのメートル。
[8] [.] [7] [↵] [LASTx]	9.5000E15	c を取得します。
[x]	8.2650E16	シリウスまでのメートル。

RPN モードでの連鎖計算

RPN モードでは、スタックの内容が自動的に積み上げられて破棄されるため、格納したり再入力することなく、また括弧を使用することなく、中間結果を保持できます。

括弧無しでの作業

例えば、 $(12 + 3) \times 7$ を評価します。

紙に書いて計算するときは、最初に $(12 + 3)$ の中間結果を計算します ...

$$(12 + 3) = 15$$

次に、この中間結果に 7 を乗算します。

$$(15) \times 7 = 105$$

2-12 RPN：自動メモリスタック

HP 35s での式の評価も同様で、括弧の内側から開始します。

キー:	ディスプレイ:	説明:
1 2 ENTER 3 +	15.0000	最初に中間結果を計算します。

計算を続行する前に中間結果を格納するため **ENTER** を押す必要はありません。計算された結果であるため、自動的に格納されるからです。

キー:	ディスプレイ:	説明:
7 x	105.0000	関数キーを押して、答えを生成します。この結果をのちの計算で利用できます。

以下の例を検討します。**ENTER** を押す必要があるのは、例えば式の開始時などに連続して入力された数値を分離する場合だけです。演算自体 (**+**、**-**、など) が後続の数値を分離し、中間結果を格納します。格納されている最後の結果が、計算に必要となると取得される最初の値です。

$2 \div (3 + 10)$ を計算します。

キー:	ディスプレイ:	説明:
3 ENTER 1 0 +	13.0000	$(3 + 10)$ を最初に計算します。
2 x$\leftrightarrow$y $\div$	0.1538	2 を 13 の前に置いて、除算が正しく行われるようにします。 $2 \div 13$ 。

$4 \div [14 + (7 \times 3) - 2]$ を計算します。

キー:	ディスプレイ:	説明:
7 ENTER 3 x	21.0000	(7×3) を計算します。
1 4 + 2 -	33.0000	分母を計算します。
4 x$\leftrightarrow$y	33.0000	除算のため、4 を 33 の前に置きます。
$\div$	0.1212	$4 \div 33$ の答えを計算します。

複数の括弧がある問題でも、中間結果の自動格納を使って同じ方法で解くことができます。例えば $(3 + 4) \times (5 + 6)$ を紙で解く場合、最初に $(3 + 4)$ を計算します。次に $(5 + 6)$ を計算します。最後に、2 つの中間結果を乗算して答えを得ます。

HP 35s でも問題の解き方は同じですが、電卓が中間結果を記憶しているので書き留める必要がありません。

キー：	ディスプレイ：	説明：
3 ENTER 4 +	7.0000	最初に加算 (3+4) を実行します
5 ENTER 6 +	11.0000	次に加算 (5+6) を実行します
×	77.0000	中間結果同士を乗算して最終結果を得ます。

練習

計算：

$$\frac{\sqrt{(16.3805 \times 5)}}{0.05} = 181.0000$$

解法：

1 **6** **×** **3** **8** **0** **5** **ENTER** **5** **×** **√x** **×** **0** **0** **5** **÷**

計算：

$$\sqrt{[(2+3) \times (4+5)]} + \sqrt{[(6+7) \times (8+9)]} = 21.5743$$

解法：

2 **ENTER** **3** **+** **4** **ENTER** **5** **+** **×** **√x** **6** **ENTER** **7** **+** **8** **ENTER** **9** **+** **×** **√x** **+**

計算：

$$(10 - 5) \div [(17 - 12) \times 4] = 0.2500$$

解法：

1 **7** **ENTER** **1** **2** **-** **4** **×** **1** **0** **ENTER** **5** **-** **x↔y** **÷**

または

1 **0** **ENTER** **5** **-** **1** **7** **ENTER** **1** **2** **-** **4** **×** **÷**

計算の順番

連鎖計算の推奨される解法は、括弧の内側から外側へ向かって作業を行う方法です。しかし、左から右の順番で問題进行处理する方法もあります。

例えば、次の式をすでに計算しました。

$$4 \div [14 + (7 \times 3) - 2]$$

その際、鉛筆と紙で計算する場合と同様、一番内側の括弧 (7×3) から開始し、外側に向かって作業を進めました。キー入力は、**[7] [ENTER] [3] [×] [1] [4] [+] [2] [-] [4] [x↔y] [÷]** です。

左から右に問題を解く場合、次のように押します。

[4] [ENTER] [1] [4] [ENTER] [7] [ENTER] [3] [×] [+] [2] [-] [÷]。

この方法では、キー入力が 1 つ増えます。最初の間中間結果は、やはり一番内側の括弧 (7×3) です。左から右に問題を解く場合の長所は、**[x↔y]** を使用して**非可換関数** (**[−]**、**[÷]**) の演算対象の位置を変更する必要がない点です。

ただし通常、次の理由から、一番内側の括弧から開始する最初の方法を推奨します。

- キー入力が少ない。
- スタックで必要なレジスタの数が少ない。



注 左から右への方法を使用する場合、一度に 4 個を超える中間数値（または結果）が必要にならないことを確認してください（スタックが保持できる数値は 4 個までです）。

上記の例では、**左から右へ** 解く場合、ある時点でスタック内のすべてのレジスタが必要になります。

キー：	ディスプレイ：	説明：
4 ENTER 1 4	14.0000	4 と 14 を中間数値としてスタックに格納します。
ENTER		
7 ENTER 3	3_	この時点で、スタックが計算用の数値で埋まります。
×	21.0000	中間結果。
+	35.0000	中間結果。
2 -	33.0000	中間結果。
÷	0.1212	最終結果。

追加練習

次の問題を通して RPN の使い方を練習します。

計算：

$$(14 + 12) \times (18 - 12) \div (9 - 7) = 78.0000$$

解法：

1 **4** **ENTER** **1** **2** **+** **1** **8** **ENTER** **1** **2** **-** **×** **9** **ENTER** **7** **-** **÷**

計算：

$$23^2 - (13 \times 9) + 1/7 = 412.1429$$

解法：

2 **3** **$\Rightarrow$** **x^2** **1** **3** **ENTER** **9** **×** **-** **7** **$1/x$** **+**

計算：

$$\sqrt{(5.4 \times 0.8) \div (12.5 - 0.7^3)} = 0.5961$$

解法：

5 **•** **4** **ENTER** **•** **8** **×** **•** **7** **ENTER** **3** **y^x** **1** **2** **•** **5** **x[↔]y** **—**
÷ **√x**

または

5 **•** **4** **ENTER** **•** **8** **×** **1** **2** **•** **5** **ENTER** **•** **7** **ENTER** **3** **y^x** **—**
÷ **√x**

計算：

$$\sqrt{\frac{8.33 \times (4 - 5.2) \div [(8.33 - 7.46) \times 0.32]}{4.3 \times (3.15 - 2.75) - (1.71 \times 2.01)}} = 4.5728$$

解法：

4 **ENTER** **5** **•** **2** **—** **8** **•** **3** **3** **×** **↵** **LASTx** **7** **•** **4** **6** **—**
0 **•** **3** **2** **×** **÷** **3** **•** **1** **5** **ENTER** **2** **•** **7** **5** **—** **4** **•** **3** **×**
1 **•** **7** **1** **ENTER** **2** **•** **0** **1** **×** **—** **÷** **√x**

データの変数への格納

HP 35s は、数値、方程式、プログラムを格納できる 30KB のメモリを備えています。数値は、A から Z までの名前が付いた、**変数**と呼ばれる場所に格納されます。(銀行残高には B、光速には C など、格納した内容に関連した文字を選択できます)。

例：

この例では、最初に RPN モード、次に ALG モードで、変数 A に値 3 を格納する方法を示します。

キー：	ディスプレイ：	説明：
[MODE] [5] (5 RPN)		RPN モードに変更します (必要な場合)
[3]	0.0000 3_	値 (3) を入力します
[⇨] [STO]	STO_	Store コマンドが文字の入力を求め、A...Z インジケータが表示されます。
[A]	0.0000 3.0000	値 3 が A に格納され、スタックに戻されます。
[MODE] [4] (4 ALG)		ALG モードに変更します (必要な場合)
	3.0000	
[3] [⇨] [STO] [A]	3▶A_	再度、Store コマンドが文字の入力を求め、A...Z インジケータが表示されます。
[ENTER]	3▶A 3.0000	値 3 が A に格納され、結果が第 2 行に表示されます。

ALG モードでは、式を変数に格納できます。この場合、式の値が、式自体よりも変数に格納されます。

例：

キー：	ディスプレイ：	説明：
    	1+3÷4 	式を入力し、前の例と同様に
   	1.7500	進めます。

ピンクの文字はそれぞれ、1 つのキーと固有の変数に関連付けられています。
(ディスプレイの **A..Z** インジケータでこれを確認します)。

変数 X、Y、Z、T は、スタックの X レジスタ、Y レジスタ、Z レジスタ、T レジスタと別の格納場所です。

数値の格納と呼び出し

数値とベクトルは、Store ( ) コマンドと Recall () コマンドによって、文字名を付けた変数へ格納され、変数から呼び出されます。数値は、HP 35s によってサポートされる、実数または複素数、10 進数または分数、基数 10 などです。

表示されている数値 (X レジスタ) のコピーを直接変数に格納するには：

  文字キー  を押します。

数値のコピーを直接変数からディスプレイに呼び出すには：

 文字キー  を押します。

例：数値の格納

アボガドロ数（約 6.0221×10^{23} ）を A に格納します。

キー：	ディスプレイ：	説明：
6 . 0 2 2 1	6.0221E23_	アボガドロ数
E 2 3	6.0221E23▶R_	“▶” は変数の入力进行を求めます。
▶ STO A	6.0221E23▶R	アボガドロ数のコピーを A に格納
ENTER	6.0221E23	します。これにより数値入力も完了
		します。
C	-	ディスプレイの数値をクリアします。
RCL	A..Z	A..Z インジケータがオンになります
A ENTER	R=	アボガドロ数を A からディスプレ
	6.0221E23	イにコピーします。

変数に格納されている値を呼び出すには、Recall コマンドを使用します。次の例で示すように、このコマンドの表示は、RPN モードと ALG モードで少し異なります。

例：

この例では、最後の例で変数 G に格納した値 1.75 を呼び出します。例では、HP 35s は最初、ALG モードであるとします。

キー：	ディスプレイ：	説明：
RCL G ENTER	G	RCL を押すと A...Z モードがアク
	1.7500	ティブになります。コマンドが
		第 1 行にペーストされることはあ
		りません。

ALG モードでは、Recall を使用して変数をコマンド行の式にペーストできます。
15-2×G を評価するとします。上記から G=1.75 です。

キー:	ディスプレイ:	説明:
1 5 - 2 ×	15-2×G	
RCL G ENTER	11.5000	

RPN モードに変更し、G の値を呼び出します。

キー:	ディスプレイ:	説明:
MODE 5 (5RPN)		RPN モードに変更します

RCL		RPN モードでは、 RCL は編集行
	RCL _	に コマンドをペーストします。
G	1.7500	ENTER を押す必要はありません
	1.7500	

変数の表示

VIEW コマンド (**VIEW**) は、X レジスタに値を呼び出すことなく、変数の値を表示します。表示は「変数=値」の形式です。数値の桁が多すぎて画面に納まらないときは、**→** または **←** を使用して表示されていない数字を表示します。VIEW 表示をキャンセルするには、**←** または **C** を押します。VIEW コマンドはプログラミングでよく使われますが、スタックを変更せずに変数を表示したいときはいつでも使用できます。

MEM カタログの使用

MEMORY カタログ ( **MEM**) は、利用可能なメモリの量に関する情報を表示します。カタログ表示の形式は、次のとおりです。

1. VAR 2. PGM

nnnn mmm・mmmm

ここで、**mmm, mmm** は、利用可能なメモリのバイト数、**nnn** は、使用されている間接変数の量です。

間接変数の詳細については、第 14 章を参照してください。

VAR カタログ

デフォルトで、A から Z のすべての直接変数には値ゼロが入っています。直接変数にゼロ以外の数値を格納する場合、変数の値を VAR カタログで表示できません ( **MEM** **1** (1VAR))。

例：

この例では、C に 3 を、D に 4 を、E に 5 を格納します。これらの変数を VAR カタログで表示したあと、クリアします。例では RPN モードを使用します。

キー：	ディスプレイ：	説明：
 CLEAR 2		すべての直接変数をクリアします
(2VARS)		
3  STO C	4	C に 3 を、D に 4 を、E に 5 を格
4  STO D	5	納します。
5  STO E		
 MEM 1 (1VAR)	C=	VAR カタログを入力します。
	3	

▲ インジケータと ▼ インジケータにより、カタログのスクロールに便利な

▼ キーと ▲ キーがアクティブであることが示されます。ただし、分数表示モードがアクティブの場合、カタログ内の変数が 1 個だけでないと、確度を示す ▲ インジケータと ▼ インジケータはアクティブになりません。VAR カタログのナビゲート方法を示す例に戻ります。

▼ D= ゼロではない値を持つ次の直接変数 (D=4) にスクロールします。
4
▼ E= もう一度スクロールして E=5 を表示します。
5

VAR カタログを表示しているあいだに、例を使って変数の値をクリアしてゼロに戻す方法を示します。E を削除します。

▶ CLEAR C= E の値はゼロなので、E は VAR カタログに表示されなくなります。
3 ここに示すように、次の変数は C です。

変数 C の値をスタックにコピーするとします。

ENTER 5 C=3 の値が x レジスタにコピーされ、5（前に E=5 と定義）が y レジスタに移動します。
3

いつでも VAR カタログを終了するには、◀ または C を押します。変数をクリアする別の方法として、変数に値ゼロを格納する方法があります。すべての直接変数をクリアするには、▶ CLEAR 2 (2VARS) を押します。すべての直接変数の値がゼロの場合、VAR カタログを入力しようとすると、エラーメッセージ "ALL VARS = 0" が表示されます。

変数の値の桁が多すぎて画面に納まらないときは、> と < を使用して表示されていない数字を表示できます。

3-6 データの変数への格納

✓ 格納した変数を使った計算

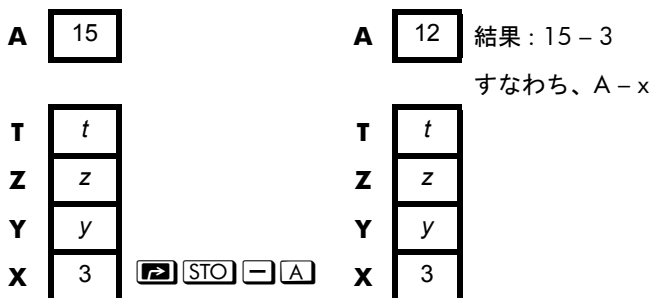
STO 四則演算と**RCL 四則演算**を使用すると、変数をスタックに呼び出さずに、変数に格納された数値を使って計算を実行できます。計算では X レジスタの値と、指定された変数内の数値を使用します。

✓ STO 四則演算

STO 四則演算では  **STO** 、 **STO** 、 **STO** 、 **STO**   を使用して、変数自体内で計算を実行し、結果をそこに格納します。X レジスタの値を使用し、スタックに影響しません。

変数の新しい値 = 変数の前の値 {+, −, ×, ÷} ×

例えば、A 内の値 (15) を X レジスタ内の値 (3、表示) だけ減算するとします。 **STO**  **A** を押します。現在、A = 12 です。3 はディスプレイに表示したままとなります。

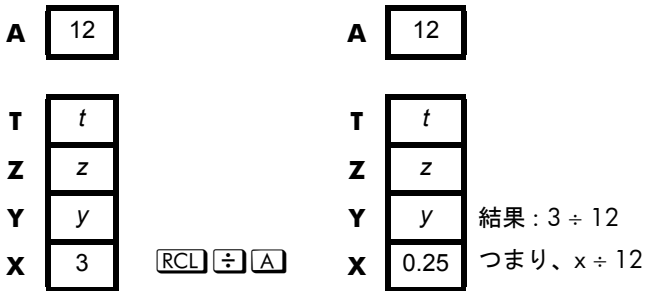


✓ RCL 四則演算

RCL 四則演算では、**[RCL] [+]**、**[RCL] [-]**、**[RCL] [×]**、**[RCL] [÷]** を使用し、呼び出された数値を使って X レジスタ内で計算を実行し、結果をディスプレイに表示したままにします。X レジスタのみに影響します。変数内の値は同じままで、結果が X レジスタ内の値と置き換わります。

新しい $x = \text{前の } x \{+, -, \times, \div\} \text{ 変数}$

例えば、X レジスタ内の数値（3、表示）を A 内の値（12）で除算するとします。**[RCL] [÷] A** を押します。現在、 $x = 0.25$ です。12 は A に格納されたままとなります。RCL 四則演算により、プログラムのメモリを節約できます。**[RCL] [+] A**（1 個の命令）を使用すると、**[RCL] A, [+]**（2 個の命令）の場合のメモリの半分になります。



✓ 例：

変数 D 、 E 、 F に値 1、2、3 が含まれるとします。STO 四則演算を使用して、それぞれの変数に 1 を加算します。

キー：	ディスプレイ：	説明：
$\boxed{1} \boxed{\rightarrow} \boxed{\text{STO}} \boxed{D}$	1.0000	仮定された値を変数に格納します。
$\boxed{2} \boxed{\rightarrow} \boxed{\text{STO}} \boxed{E}$	2.0000	
$\boxed{3} \boxed{\rightarrow} \boxed{\text{STO}} \boxed{F}$	3.0000	
$\boxed{1} \boxed{\rightarrow} \boxed{\text{STO}}$		1 を D 、 E 、 F に加算します。
$\boxed{+} \boxed{D} \boxed{\rightarrow} \boxed{\text{STO}}$	1.0000	
$\boxed{+} \boxed{E} \boxed{\rightarrow} \boxed{\text{STO}}$		
$\boxed{+} \boxed{F}$		
$\boxed{\leftarrow} \boxed{\text{VIEW}} \boxed{D}$	D= 2.0000	D の現在の値を表示します。
$\boxed{\leftarrow} \boxed{\text{VIEW}} \boxed{E}$	E= 3.0000	
$\boxed{\leftarrow} \boxed{\text{VIEW}} \boxed{F}$	F= 4.0000	
$\boxed{\leftarrow}$	1.0000	VIEW 表示をクリアし、X レジスタを再度表示します。

変数 D 、 E 、 F に最後の例からの値 2、3、4 が含まれているとします。3 を D で除算し、 E で乗算し、結果に F を加算します。

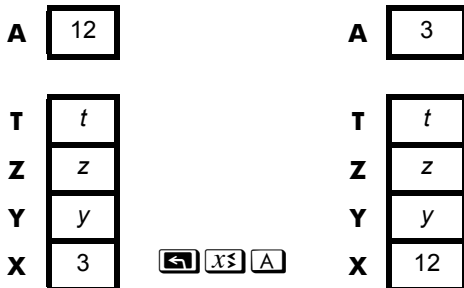
キー：	ディスプレイ：	説明：
$\boxed{3} \boxed{\text{RCL}} \boxed{\div} \boxed{D}$	1.5000	$3 \div D$ を計算します。
$\boxed{\text{RCL}} \boxed{\times} \boxed{E}$	4.5000	$3 \div D \times E$
$\boxed{\text{RCL}} \boxed{+} \boxed{F}$	8.5000	$3 \div D \times E + F$

ある変数と x を交換

  キーを使用すると、x（表示された X レジスタ）の内容を任意の変数の内容と交換できます。この関数を実行しても Y レジスタ、Z レジスタ、T レジスタには影響しません。

例：

キー：	ディスプレイ：	説明：
  	12.0000	12 を変数 A に格納します。
 		
	3_	x を表示します。
  	12.0000	X レジスタと変数 A の内容を交換します。
  	3.0000	X レジスタと変数 A の内容を交換します。



変数 I と J

直接アクセスできる 2 個の変数、I と J があります。I と J は、他の変数と同様に数値を格納しますが、(I) コマンドと (J) コマンドを使って他の変数（統計レジスタを含む）を参照するために使用できる点で特別です。(I) は  キーにあり、(J) は  キーにあります。これは間接アドレッシングと呼ばれるプログラミング技法です。詳細については、第 14 章の「変数とラベルの間接アドレッシング」を参照してください。

3-10 データの変数への格納

実数の関数

この章では、絶対値を求める ABS など、プログラムで使われる数値関数も含め、実数を演算する関数のほとんどをカバーします。実数関数には次のようなグループがあります。

- 指数関数と対数関数
- 除算の商と剰余
- 累乗関数 (y^x と $\frac{x}{y}$)
- 三角関数
- 双曲線関数
- % 関数
- 物理定数
- 座標、角度、単位の変換機能
- 確率関数
- 数値の部分 (数値変換機能)

四則演算は、第 1 章と第 2 章でカバーされています。これ以降の章では、高度な算術演算 (根の決定、積分、複素数、基数変換、統計) について説明します。この章の例では、HP 35s は RPN モードであるとします。

✓ 指数関数と対数関数

ディスプレイに数値を表示させ、関数を実行します。**[ENTER]** を押す必要はありません。

計算内容：	以下のキーを押します。
自然対数（底 e ）	[2nd] [LN]
常用対数（底 10）	[2nd] [LOG]
e を底とする指数	[2nd] [e^x]
10 を底とする指数（真数）	[2nd] [10^x]

✓ 除算の商と剰余

[5] **[INTG]** **[2]** (**2INT÷**) と **[5]** **[INTG]** **[3]** (**3Rmdr**) を使用して、2 個の整数の除算から、それぞれ整数の商と整数の剰余を求めることができます。

1. 最初の整数を入力します。
2. **[ENTER]** を押して、最初の数値を 2 番目の数値から区切ります。
3. 2 番目の数値を入力します。（**[ENTER]** を押さないでください）。
4. 関数キーを押します。

例：

58 ÷ 9 の商と剰余を表示します

キー：	ディスプレイ：	説明：
[5] [8] [ENTER] [9] [2nd]	6.0000	商を表示します。
[INTG] [2] (2INT÷)		
[5] [8] [ENTER] [9] [2nd]	4.0000	剰余を表示します。
[INTG] [3] (3Rmdr)		

✓ 累乗関数

RPN モードで数値 y の x 乗を計算するには、 y **[ENTER]** x を入力し、**[y^x]** を押します。 $(y > 0)$ の場合、 x は任意の数値ですが、 $y < 0$ の場合、 x は正でなければなりません。

計算内容：	以下のキーを押します。	結果：
152	[1] [5] [$\rightarrow$] [x^2]	225.0000
106	[6] [$\leftarrow$] [10^x]	1,000,000.0000
54	[5] [ENTER] [4] [y^x]	625.0000
2-1.4	[2] [ENTER] [1] [.] [4] [+/-] [y^x]	0.3789
$(-1.4)^3$	[1] [.] [4] [+/-] [ENTER] [3] [y^x]	-2.7440

RPN モードで数値 y の根 x (y の x 乗根) を計算するには、 y **[ENTER]** x を入力し、**[$\leftarrow$]** **[$\sqrt[y]{x}$]** を押します。 $y < 0$ の場合、 x は、整数でなければなりません。

計算内容：	以下のキーを押します。	結果：
$\sqrt{196}$	[1] [9] [6] [$\sqrt{x}$]	14.0000
$\sqrt[3]{-125}$	[1] [2] [5] [+/-] [ENTER] [3] [$\leftarrow$] [$\sqrt[y]{x}$]	-5.0000
$\sqrt[4]{625}$	[6] [2] [5] [ENTER] [4] [$\sqrt[y]{x}$]	5.0000
$-1.4\sqrt[3]{.37893}$	[.] [3] [7] [8] [9] [3] [ENTER] [1] [.] [4] [+/-] [$\leftarrow$] [$\sqrt[y]{x}$]	2.0000

三角関数

π の入力

  を押すと、 π の最初の 12 桁が X レジスタに格納されます。

(表示される数値は、表示設定に依存します。)   は π の近似値をスタックに戻す関数であるため、 を押す必要はありません。

π は超越数であるため、電卓は π を正確に表示できません。

角度モードの設定

角度モードは、三角関数で使用する角度の単位を指定します。このモードは、すでに入力された数値を変換しません (この章で後述する「変換関数」を参照してください)。

$$360 \text{ 度} = 2\pi \text{ ラジアン} = 400 \text{ グラジアン}$$

角度モードを設定するには、 を押します。表示されるメニューから、次のオプションを選択できます。

オプション	説明	インジケータ
DEG	度 (degree) に設定します。度を 10 進数で入力します。60 進数 (度、分、秒) ではありません。	なし
RAD	ラジアンモードに設定します。	RAD
GRAD	グラジアンモードに設定します。	GRAD

✓ 三角関数

x を使って表示されている場合

計算内容：	以下のキーを押します。
x のサイン	SIN
x のコサイン	COS
x のタンジェント	TAN
x のアークサイン	⇐ ASIN
x のアークコサイン	⇐ ACOS
x のアークタンジェント	⇐ ATAN

注



無理数 π を持つ計算は、電卓の内部精度である 15 桁では**正確に**表現できません。これは、三角関数の演算で特に顕著です。例えば、 $\sin \pi$ (ラジアン) の計算値は、ゼロではなく、ゼロに近い非常に小さい数値、 -2.0676×10^{-13} になります。

例：

cosine $(5/7)\pi$ ラジアンと cosine 128.57° が有効桁 4 桁まで等しいことを確認します。

キー：

ディスプレイ：

説明：

MODE **2** (2RAD)

ラジアンモードに設定します。

• **5** **•** **7** **ENTER**

0.7143

RAD インジケータがオンになります。

$5/7$ が、小数点表記で表示されます。

⇐ **π** **×** **COS**

-0.6235

$\cos (5/7)\pi$.

MODE **1** (1DEG)

-0.6235

度モードに変更します (インジケータはありません)。

1 **2** **8** **•** **5** **7**

-0.6235

$\cos 128.57^\circ$ を計算します。

COS

$\cos (5/7)\pi$ と同じになります。

プログラミングの注意：

逆三角関数を使用して角度 θ を求める方程式は、通常、次のような形式になります。

$$\theta = \arctan (y/x).$$

$x = 0$ の場合、 y/x が未定義になり、エラー：DIVIDE BY 0 が発生します。

双曲線関数

x を使って表示されている場合

計算内容：	以下のキーを押します。
x の双曲サイン (SINH)	 HYP 
x の双曲コサイン (COSH)	 HYP 
x の双曲タンジェント (TANH)	 HYP 
x の双曲アークサイン (ASINH)	 HYP  
x の双曲アークコサイン (ACOSH)	 HYP  
x の双曲アークタンジェント (ATANH)	 HYP  

✓ % 関数

% 関数は、 や  と異なり、パーセンテージ計算の結果を (X レジスタに) 戻すときに、元となる数値 (Y レジスタ内) がそのまま保持されます。元となる数値を再入力することなく、元の数値と計算結果の両方を使って後続の計算を実行できます。

計算内容：	以下のキーを押します。
y の x%	y  x  
y から x へのパーセンテージ変更 (y≠ 0)	y  x  

例：

6% の消費税と、\$15.76 の商品の総額を求めます。

価格を適切に丸めて表示するため、FIX 2 表示設定を使用します。

キー：	ディスプレイ：	説明：
DISPLAY 1 (1FIX)		表示を小数点以下 2 桁に丸めます。
2		
1 5 . 7 6 ENTER	15.76	
6 %	0.95	6% の税金を計算します。
+	16.71	総額（税抜き価格 + 6% の税金）

昨年、価格が \$16.12 だった商品が今年は \$15.76 となりました。昨年から今年の価格の変化率はいくらかでしょうか？

キー：	ディスプレイ：	説明：
1 6 . 1 2 ENTER	16.12	
1 5 . 7 6	-2.23	今年の価格は昨年の価格から、約 2.2% 下落しました。
%CHG		
DISPLAY 1 (1FIX)	-2.2333	表示形式を FIX 4 に戻します。
4		

注



%CHG 関数では 2 つの数値の順番が重要です。順番は、プラスとマイナスのどちらのパーセンテージ変化を考えるかに影響します。

物理定数

CONST メニューには 41 個の物理定数があります。 **CONST** を押すと次の項目を表示できます。

CONST メニュー

項目	説明	値
C	真空中の光速	$299792458\text{ m s}^{-1}$
g	標準重力加速度	9.80665 m s^{-2}
G	万有引力定数	$6.673\times 10^{-11}\text{ m}^3\text{ kg}^{-1}\text{ s}^{-2}$
Vm	理想気体のモル体積	$0.022413996\text{ m}^3\text{ mol}^{-1}$
NA	アボガドロ定数	$6.02214199\times 10^{23}\text{ mol}^{-1}$
R∞	リュードベリ定数	$10973731.5685\text{ m}^{-1}$
eV	電気素量	$1.602176462\times 10^{-19}\text{ C}$
me	電子素量	$9.10938188\times 10^{-31}\text{ kg}$
mP	陽子の質量	$1.67262158\times 10^{-27}\text{ kg}$
mn	中性子の質量	$1.67492716\times 10^{-27}\text{ kg}$
mH	ミュー粒子の質量	$1.88353109\times 10^{-28}\text{ kg}$
k	ボルツマン定数	$1.3806503\times 10^{-23}\text{ J K}^{-1}$
h	プランク定数	$6.62606876\times 10^{-34}\text{ J s}$
ħ	プランク定数 /2 pi	$1.054571596\times 10^{-34}\text{ J s}$
Φ0	磁束量子	$2.067833636\times 10^{-15}\text{ Wb}$
a0	ボーア半径	$5.291772083\times 10^{-11}\text{ m}$
ε0	真空の誘電率	$8.854187817\times 10^{-12}\text{ F m}^{-1}$
R	モル気体定数	$8.314472\text{ J mol}^{-1}\text{ K}^{-1}$
F	ファラデー定数	$96485.3415\text{ C mol}^{-1}$
u	原子質量定数	$1.66053873\times 10^{-27}\text{ kg}$
μ0	磁気定数	$1.2566370614\times 10^{-6}\text{ NA}^{-2}$
μB	ボーア磁子	$9.27400899\times 10^{-24}\text{ J T}^{-1}$
μN	核磁子	$5.05078317\times 10^{-27}\text{ J T}^{-1}$
μP	陽子の磁気モーメント	$1.410606633\times 10^{-26}\text{ J T}^{-1}$
μe	電子の磁気モーメント	$-9.28476362\times 10^{-24}\text{ J T}^{-1}$
μn	中性子の磁気モーメント	$-9.662364\times 10^{-27}\text{ J T}^{-1}$

項目	説明	値
μ_B	ミュー粒子の磁気モーメント	$-4.49044813 \times 10^{-26} \text{ J T}^{-1}$
r_e	古典電子半径	$2.817940285 \times 10^{-15} \text{ m}$
Z_0	真空の特性インピーダンス	$376.730313461 \text{ } \Omega$
λ_C	コンプトン波長	$2.426310215 \times 10^{-12} \text{ m}$
λ_{Cn}	中性子のコンプトン波長	$1.319590898 \times 10^{-15} \text{ m}$
λ_{Cp}	陽子のコンプトン波長	$1.321409847 \times 10^{-15} \text{ m}$
α	微細構造定数	$7.297352533 \times 10^{-3}$
σ	ステファン・ボルツマン定数	$5.6704 \times 10^{-8} \text{ W m}^{-2} \text{ K}^{-4}$
t	セルシウス温度（摂氏）	273.15
p_{atm}	標準気圧	101325 Pa
γ_P	陽子の磁気回転比	$267522212 \text{ s}^{-1} \text{ T}^{-1}$
C_1	第一放射定数	$374177107 \times 10^{-16} \text{ W m}^2$
C_2	第二放射定数	0.014387752 m K
G_0	コンダクタンス量子	$7.748091696 \times 10^{-5} \text{ S}$
e	自然対数の基数（自然定数）	2.71828182846

参照：Peter J.Mohr and Barry N.Taylor, CODATA Recommended Values of the Fundamental Physical Constants: 1998, Journal of Physical and Chemical Reference Data, Vol.28, No.6, 1999 and Reviews of Modern Physics, Vol.72, No.2, 2000.

定数を挿入するには：

1. 定数を挿入する場所にカーソルを置きます。
2.  **CONST** を押して、物理定数メニューを表示します。
3.  を押して（または  **CONST** を押して次のページに1ページずつアクセスできます）、目的の定数にアンダーラインが引かれるまでスクロールし、**ENTER** を押してその定数を挿入します。

式、方程式、プログラム内で使用される定数は、数値ではなく名前で参照する必要があります。

変換関数

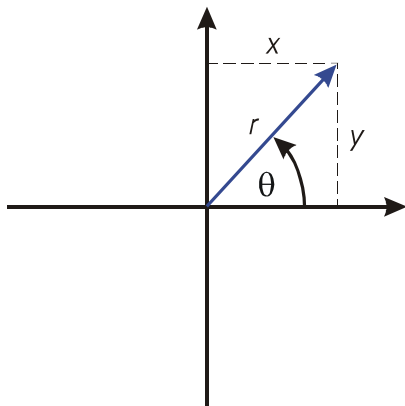
HP 35s には 4 種類の変換があります。次の数値を相互に変換できます。

- 複素数の直交形式と極形式
- 角度の単位：度、ラジアン、グラジアン
- 時間と角度の 10 進数表記と 60 進数表記
- サポートされるさまざまな単位 (cm/in、kg/lb など)

直交座標変換と極座標変換を除いて、それぞれの変換には特定のキーが割り当てられています。キーの左 (黄色) シフトに割り当てられた変換と、右 (青色) シフトに割り当てられた変換は、逆方向になります。この種の変換では、入力した数値が、別の単位を使って測定されているものと考えます。例えば、 $\boxed{\rightarrow^{\circ}\text{F}}$ を使ってある数値を華氏に変換するとき、入力する数値は、摂氏で測定された温度であると見なします。この章の例では、RPN モードを使用します。ALG モードでは、最初に関数を入力してから、数値を変換します。

直交座標／極座標変換

極座標 (r, θ) と直交座標 (x, y) は、次の図に示すように測定します。角度 θ は、現在の角度モードで設定された単位を使用します。 θ の計算結果は、 $-180^{\circ} \sim 180^{\circ}$ の度、 $-\pi \sim \pi$ のラジアン、または $-200 \sim 200$ のグラジアン範囲になります。



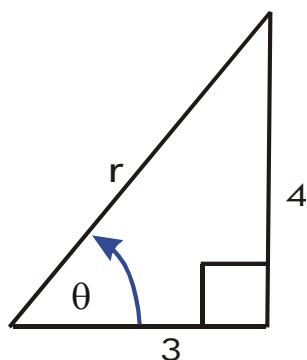
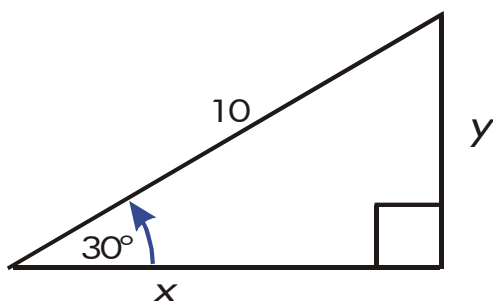
直交座標と極座標を変換するには：

複素数を表示するための形式は、モード設定です。複素数は、任意の形式で入力できます。複素数は、入力後、モード設定によって決定された表示形式に変換されます。複素数の表示形式を設定するには、次の手順を実行します。

1. RPN モードでは、**[]** **[DISPLAY]** を押し、**[9]** ($9 \times i$) または **[0]** ($10r\theta a$) を選択します (ALG モードでは、**[]** **[1]** ($11 \times + i$) も選択できます)
2. 目的の座標値を入力します (x **[]** y 、 x **[]** y **[]**、 r **[]** **[]** a)
- 3 **[ENTER]** を押します

例：極座標から直交座標への変換

次の直角三角形で、左の三角形の辺 x と y 、および右の三角形の斜辺 r と角度 θ を求めます。



キー：

ディスプレイ：

説明：

[MODE] **[1]** (1DEG)

[] **[DISPLAY]** **[9]** ($9 \times i$)

[1] **[0]** **[]** **[]** **[3]** **[0]** 8.6603 i 5.0000

[ENTER]

Degree と複素数座標モードを設定します。

$r\theta a$ (極座標) を xiy (直交座標) に変換します。

← **DISPLAY** **·** **0**

(10rθa)

3 **i** **4** **ENTER**

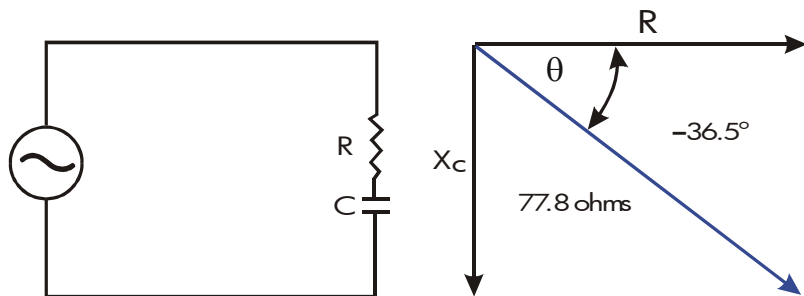
10.0000030.0000 複素数座標モードを設定します。

5.0000053.1301 xiy (直交座標) を $r\theta a$ (極座標) に変換します。

例：ベクトルの変換

次に示す RC 回路は、総インピーダンスが 77.8Ω 、電圧の電流に対する遅れが 36.5° であることがわかりました。この回路の抵抗 R と容量性リアクタンス X_C はどれだけのしょうか？

次のベクトル図を使います。インピーダンスは極座標の大きさ r と等しく、電圧の遅れは角度 θ 、(単位度) と等しくなります。これらの値を直交座標に変換すれば、 x - の値から R (単位 Ω)、 y の値から X_C (単位 Ω) が得られます。



キー：

MODE **1** (1DEG)

← **DISPLAY** **9** ($9 \times i.y$)

7 **7** **·** **8** **→** **0** 77.80-36.5

3 **6** **·** **5** **↵**

ENTER

ディスプレイ：

62.5401 i -46.2772

説明：

Degree と複素数座標モードを設定します。

θ を電圧の遅れの角度として入力します。 r を総インピーダンスのオームとして入力します。 x をオーム 抵抗、 R として計算します。
 y をオーム リアクタンス X_C として計算します。

時間の変換

HP 35s は、数値の 10 進数表記と 60 進数表記を変換できます。この機能は特に、時間と角度を度で測定する場合に便利です。例えば、10 進数では度で測定された角度が D.ddd... と表記されます。60 進数では、同じ角度が D.MMSSss と表記されます。ここで、D は度の整数部、ddd... は度の小数部、MM は分を表わす整数、SS は秒を表わす整数部、ss は秒の小数部です。

✓ 10 進数と時間、分、秒の変換

1. 変換する数値を入力します
2.   を押して、時間／角度、分、秒に変換します。または   を押して、10 進数に変換し直します。

例：時刻形式の変換

1 時間の 1/7 は何分何秒でしょうか？ FIX 6 表示形式を使用します。

キー：	ディスプレイ：	説明：
   (1FIX)		FIX 6 表示形式を設定します。
	0.000000	
   	0 1/7	1/7 時間を 10 進分数で入力します。
 	0.000000 0.083429	8 分 34.29 秒に等しくなります。
   (1FIX)	0.000000	表示形式を FIX 4 に戻します。
	0.0834	

✓ 角度の変換

ラジアンへの変更のとき、X レジスタ内の数値は、度単位であるとして計算します。度への変更のとき、X レジスタ内の数値は、ラジアン単位であるとして計算します。

角度の度とラジアンを変換するには：

例

この例では、角度を 30° から $\pi/6$ ラジアンに変換します。

キー：	ディスプレイ：	説明：
3 0	0.0000	角度を度で入力します。
	30	
↵ →RAD	0.0000	ラジアンに変換します。計算結果
	0.5236	は、0.5236 で、 $\pi/6$ の 10 進数
		近似です。

単位変換

HP 35s のキーボードには、→kg、→lb、→°C、→°F、→cm、→in、→l、→gal、→MILE、→KM の 10 個の単位変換機能があります。

変換元：	変換先：	以下のキーを 押します。	表示結果：
1 lb	kg	1 ↵ →kg	0.4536 (kg)
1 kg	lb	1 ↵ →lb	2.2046 (ポンド)
32 °F	°C	3 2 ↵ →°C	0.0000 (°C)
100 °C	°F	1 0 0 ↵ →°F	212.0000 (°F)
1 in	cm	1 ↵ →cm	2.5400 (cm)
100 cm	in	1 0 0 ↵ →in	39.3701 (インチ)
1 gal	l	1 ↵ →l	3.7854 (リットル)
1 l	gal	1 ↵ →gal	0.2642 (ガロン)
1 MILE	KM	1 ↵ →KM	1.6093 (KMS)
1 KM	MILE	1 ↵ →MILE	0.6214 (マイル)

確率関数

✓ 階乗

表示されている負でない整数 x ($0 \leq x \leq 253$) の**階乗**を計算するには、 
(右シフト  キー) を押します。

✓ ガンマ

整数ではない x の**ガンマ関数** $\Gamma(x)$ を計算するには、 $(x-1)$ を入力し、 
を押します。 $x!$ 関数は、 $\Gamma(x+1)$ を計算します。 x の値を負の整数にすることはできません。

確率

✓ 組合せ

n 個から一度に r 個を選ぶときの可能な**セット数**を計算するには、最初に n 、 、次に r (負でない整数のみ) を入力します。セット内では項目が重複して発生せず、同じ r 項目でも順番が異なる場合、別のものとしてカウントされません。

✓ 順列

n 個から一度に r 個を選ぶときの可能な**並べ方**の数を計算するには、最初に n 、 、次に r (負でない整数のみ) を入力します。並べ方内では項目が重複して発生せず、同じ r 項目でも順番が異なる場合、別のものとしてカウントされます。

✓ シード

数値を x に乱数生成器の新しいシードとして格納するには、  を押します。

✓ 乱数生成器

$0 < x < 1$ の範囲で乱数を生成するには、 **RAND** を押します。(この数値は、一様分布する疑似乱数シーケンスの一部です。次のスペクトルテストを通しています : D. Knuth, *The Art of Computer Programming*, vol. 2, *Seminumerical Algorithms*, London: Addison Wesley, 1981)

RANDAM 関数は、乱数の生成時にシードを使用しています。生成された各乱数は、次の乱数のシードになります。このため、同じシードを使って開始することで、乱数のシーケンスが繰り返される可能性があります。SEED 関数を使って新しいシードを格納できます。メモリがクリアされたとき、シードがゼロにリセットされます。シードがゼロのときは、電卓が固有のシードを生成します。

例：従業員の組合せ

女性従業員 14 名、男性従業員 10 名の会社で、6 名から成る安全委員会を設立中です。異なる組合せがいくつ可能でしょうか？

キー：	ディスプレイ：	説明：
   	24	24 名を一度に 6 つにグループ分
	6_	けしました。
	134,596.0000	可能な組合せの総数。

従業員をランダムに選択する場合、委員会が女性 6 名になる確率はどれだけでしょうか？ イベントの**確率**を計算するには、**そのイベントの組合せ数**を組合せの総数によって除算します。

キー：	ディスプレイ：	説明：
   	14	14 名の女性を一度に 6 つにグ
	6_	ループ分けしました。
	3,003.0000	委員会に 6 名の女性が選ばれる
		組合せ数
	134,596.0000	組合せの総数を X レジスタに戻
		します。
	0.0223	女性の組合せを組合せ総数に
		よって除算し、女性のみとなる
		組合せの確率を求めます。

4-16 実数の関数

数値の部分

これらの機能は、主にプログラミングで使用されます。

整数部

- ✓ x の小数部を除去し、ゼロに置換するには、 **INTG** **6** (6IP) を押します。
(例えば、14.2300 の整数部は、14.0000 です)。

小数部

- ✓ x の整数部を除去し、ゼロに置換するには、 **INTG** **5** (5FP) を押します。
(例えば、14.2300 の小数部は、0.2300 です)。

絶対値

X レジスタの数値をその絶対値に置換するには、 **ABS** を押します。複素数とベクトルでは、絶対値は次のようになります。

1. $r\theta a$ 形式の複素数は r です
2. xiy 形式の複素数は $\sqrt{x^2 + y^2}$ です
3. ベクトル $[A_1, A_2, A_3, \dots, A_n]$ は $|A| = \sqrt{A_1^2 + A_2^2 + \dots + A_n^2}$ です

引数値

複素数の引数を抽出するには、 **ARG** を使用します。複素数の引数：

1. in $r\theta a$ 形式では a
2. xiy 形式では $\text{Atan}(y/x)$

符号値

- ✓ x の符号を示すには、 **INTG** **1** (1SGN) を押します。 x 値が負の場合は -1.0000 が、ゼロの場合は 0.0000 が、正の場合は 1.0000 が表示されます。

最大の整数



与えられた数値以下の最大整数を取得するには、 (4INTG) を押します。

例：

この例では、数値の一部を抽出するさまざまな操作をまとめます。

計算内容：	以下のキーを押します。	ディスプレイ：
2.47 の整数部	(6IP)	2.0000
2.47 の小数部	(5FP)	0.4700
-7 の絶対値	ABS	7.0000
9 の符号値	(1SGN)	1.0000
-5.3 以下の最大整数	(4INTG)	-6.0000

RND 関数 () は、 x を表示形式で指定された桁数に内部的に丸めます。(内部数値は 12 桁で表示されます。) 分数表示モードにおける RND 関数のふるまいについては、第 5 章を参照してください。

分数

第 1 章の**分数**では、分数の入力、表示、計算の基本について説明しました。この章では、これらのトピックの詳細を示します。分数の入力と表示をまとめると、次のようになります。

- 分数を入力するには、**[.]** を 2 回押します。混在する数値の整数部の後に 1 回、数値の分数部の分子と分母のあいだでもう 1 回押します。2 3/8 を入力するには、**[2][.][3][.][8]** を押します。5/8 を入力するには、**[.][5][.][8]** または **[0][.][5][.][8]** を押します。
- 分数表示モードのオンとオフを切り替えるには、**[ FDISP]** を押します。分数表示モードがオフになると、表示が、表示メニューによって設定された前の表示形式に戻ります。このメニューで別の形式を選択すると、分数表示モードがアクティブの場合、分数表示モードもオフになります。
- 関数が分数を処理する方法は、RND の場合を除いて 10 進数のときと同じです。詳細については、この章で後から説明します。

この章の例では、他に記載がない限り、すべて RPN モードを使用します。

分数の入力

ほとんどすべての数値をキーボードで分数として入力できます。仮分数、つまり分子が分母より大きい分数でも問題ありません。

例：

キー：	ディスプレイ：	説明：
[ FDISP]		分数表示モードをオンにします。
[1][.][5][ENTER]	1 1 1/2	1.5 を入力します。分数として表示されます。
[1][.][3][.][4][ENTER]	1 3/4	1 3/4 を入力します。
[ FDISP]	1.7500	x を小数として表示します。
[ FDISP]	1 3/4	x を分数として表示します。

もし結果が例と同じにならない場合は、分数の表示方法を誤って変更した可能性があります。(この章の「分数表示の変更」を参照してください)。

次のトピックには、さらに有効な入力分数と無効な入力分数の例が含まれています。

ディスプレイでの分数

分数表示モードでは、数値は内部で小数として評価されたあと、最も近い分数として表示されます。さらに、精度インジケータにより、12 桁の小数と比較したときの分数の不正確さの方向が示されます。(ほとんどの統計レジスタは例外で、常に小数として表示されます)。

表示のルール

表示される分数が、入力した分数と異なる場合があります。デフォルトの状態では、電卓は、次のルールに従って分数を表示します。(ルールを変更するには、この章の「分数表示の変更」を参照してください)。

- 数値には、整数部と、必要に応じて真分数 (分子が分母より小さい分数) があります。
- 分母は 4095 以下とします。
- 分数は可能な限り約分されます。

例:

入力された値と表示される結果の例です。比較のため、内部 12 桁値も表示されます。最後の列の ▲ インジケータと ▼ インジケータについて説明します。

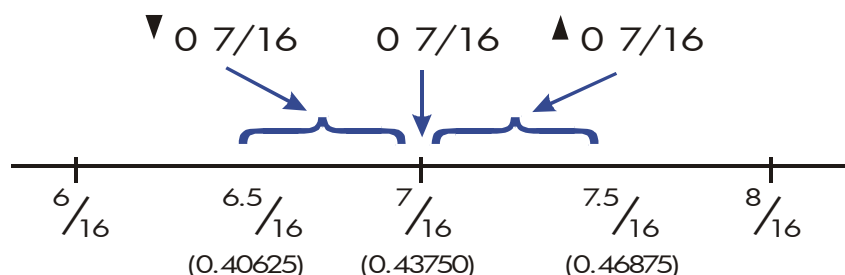
入力値	内部値	表示される分数
2 3/8	2.375000000000	2 3/8
14 15/32	14.4687500000	14 15/32
54/12	4.500000000000	4 1/2
6 18/5	9.600000000000	9 3/5
34/12	2.833333333333	2 5/6 ▼
15/8192	0.00183105469	0 7/3823 ▲
12345678 12345/3	12349793.0000	12349793
16 3/16384	16.0001831055	16 1/4095

確度インジケータ

表示中の分数の確度は、ディスプレイ右側の ▲ および ▼ インジケータで確認できます。電卓は、内部の 12 桁精度数値の分数部の値と、表示中の分数の値を比較します。

- インジケータが点灯していない場合、内部の 12 桁精度値の分数部は、表示中の分数の値と完全に一致しています。
- ▼ が点灯している場合、内部の 12 桁精度値の分数部は表示中の分数よりわずかに小さくなり、**正確な分子は**、表示中の分子より 0.5 未満だけ下になります。
- ▲ が点灯している場合、内部の 12 桁精度値の分数部は表示中の分数よりわずかに大きくなり、**正確な分子は**、表示中の分子より 0.5 未満だけ上になります。

次の図に、表示中の分数と近接値との関係を示します。▲ は、正確な分子が表示中の分子よりも「少し上に」あることを、▼ は「少し下に」あることを示します。



これは、分数の表示方法に関するルールを変更する場合に特に重要です。(次の「分数表示の変更」を参照してください)。例えば、すべての分数の分母を 5 にすると、 $2/3$ が $0\ 3/5\blacktriangle$ と表示されます。正確な分数は約 $3.3333/5$ であり、 $3/5$ より「少し上に」あるからです。同様に、 $-2/3$ は、 $-0\ 3/5\blacktriangle$ と表示されます。真の分子は 3 より「少し上に」あるからです。

インジケータが予期しないときに点灯する場合があります。例えば、 $2\ 2/3$ を入力すると、入力した値が正確であっても、 $2\ 2/3\blacktriangle$ と表示されます。電卓は、常に内部値の分数部と分数のみの 12 桁精度値を比較します。内部の数値が整数部を持つ場合、その分数部に含まれる桁は 12 桁より少なくなり、12 桁すべてを使用する分数とは、正確に一致しません。

分数表示の変更

デフォルトの状態では、電卓は、特定のルールに従って分数を表示します。ただしルールは、目的の分数の表示に応じて変更できます。

- 使用する最大分母を設定できます。
- 3 つの分数表示形式から選択できます。

次に、分数表示の変更方法について説明します。

最大分母の設定

どの分数でも、分母は、電卓に格納された値に基づいて選択されます。分数を $a\ b/c$ とすると、 $/c$ が、分母を制御する値に対応します。

$/c$ は、分数表示モードで使用される**最大** 分母のみを定義します。使用される特定の分母は、分数の表示設定に依存します (次項で説明します)。

- 最大分母を設定するには、値を入力後、  を押します。分数表示モードが自動的にオンになります。4095 を超える値は入力できません。
- $/c$ 値を X レジスタに呼び出すには、   を押します。

- デフォルト値を 4095 に復元するには、**0** **↩** **/c** を押すか、最大分母として 4095 より大きな値を入力します。やはり、分数表示モードが自動的にオンになります。

/c 機能は、X レジスタに格納されている数値の整数部の絶対値を使用します。LAST X レジスタの数値は変更しません。

表示中の分数が長すぎて画面に納まらないときは、➡ インジケータが現れます。**↩** **◀** と **↩** **▶** を使用して 1 ページずつスクロールし、分数の残りを表示します。数値の小数表示を表示するには、**↩** を押してから **SHOW** を押し続けます。

例：

この例では、最大分母を 3125 に設定し、画面に納まりきらない分数を表示するために必要な手順を示します。

キー：	ディスプレイ：	説明：
3 1 2 5 ↩		最大分母を 3125 に設定します。
/c		
1 4 ↩ e^x	0	分母内に表示されていない数値
	1202604 888/31	があります。
▶	0	右にスクロールして分母の残り
	25	を表示します。

注記：

1. ALG モードでは、第 1 行に式を入力してから、**↩** **/c** を押すことができます。この場合、最大分母を求めるため、式が評価され、計算結果が使用されます。
2. ALG モードでは、計算結果を /c 機能の引数として使用できます。第 2 行に値が表示された状態で、**↩** **/c** を押します。第 2 行の値が分数形式で表示され、最大分母を求めるため整数部が使用されます。
3. 複素数およびベクトルは /c コマンドの引数として使用できません。エラーメッセージ "INVALID DATA" が表示されます。

分数の表示形式の選択

電卓には、3つの分数表示形式があります。選択した表示形式のルールに沿って、最も精度の高い分数が表示されます。

- **最も高精度の分数。**分数は、 $/c$ 値までの分母を持ち、可能な限り約分されます。例えば、分子を使って数学的概念を調査している場合、**任意**の分母を可能にできます ($/c$ 値は 4095)。これがデフォルトの分数表示形式です。
- **分母の約数。**分数は、 $/c$ 値の約数である分母のみを持ち、可能な限り約分されます。例えば、株価を計算している場合、 $53 \frac{1}{4}$ と $37 \frac{7}{8}$ を表示できます ($/c$ 値は 8)。 $/c$ 値が 12 の場合、可能な分母は 2、3、4、6、12 です。
- **固定分母。**分数は常に、分母として $/c$ 値を使用します。分数は約分されません。例えば、時間測定を処理している場合、 $1 \frac{25}{60}$ を表示できます ($/c$ 値は 60)。

分数表示形式を制御する 3つのフラグがあります。フラグにはそれぞれ 7、8、9 という番号が割り振られています。各フラグは、クリアされた状態かセットされた状態かを保持しています。フラグの目的は次の通りです。

- フラグ 7 は、分数表示モードのオンとオフを切り替えます。クリア＝オフ、セット＝オンです。
- フラグ 8 は、 $/c$ 値以下の任意の値の使用と、 $/c$ 値の約数のみの使用を切り替えます。クリア＝任意の値の使用、セット＝ $/c$ 値の約数のみの使用です
- フラグ 9 は、フラグ 8 がセットされている状態でのみ有効で、分数を約分すると約分しないを切り替えます。クリア＝約分する、セット＝約分しないです。

フラグ 8 と 9 が適切にクリアまたはセットされている状態で、次の表に示す 3つの分数表示形式を取得できます。

この分数表示形式を取得するには：	これらのフラグを次のように変更：	
	8	9
最も高精度	クリア	—
分母の約数	セット	クリア
固定分母	セット	セット

ここに示す手順を使用して分数表示形式を設定するため、フラグ 8 と 9 を変更できます。(フラグはプログラムで特に便利です。使用方法の詳細については、第 14 章を参照してください)。

1. **FLAGS** を押して、フラグメニューを表示します。
2. フラグをセットするには、 **(1SF)** を押し、8 などのフラグ番号を入力します。
フラグをクリアするには、 **(2CF)** を押し、フラグ番号を入力します。
フラグがセットされているか確認するには、 **(3FS?)** を押し、フラグ番号を入力します。 または を押して、YES または NO 応答をクリアします。

例：

この例では、数値 π を使って分数を 3 つの表示形式で示します。例では、分数表示形式がオンで、フラグ 8 がデフォルト状態（クリア）であるとしします。

キー：	ディスプレイ：	説明：
0 5		最大 /c 値をデフォルトに戻します。
/c		最も高精度の形式
π	0	フラグ 8 = クリア
FLAGS (1SF)	0	フラグ 8 = セット
8	3 116/819	分母の約数表示形式。 819*5=4095
FLAGS (1SF)	0 0/4095	フラグ 9 = セット
9	3 580/4095	固定分母表示形式
FLAGS (2CF)	0	デフォルト形式（最も高精度）
FLAGS 2	3 16/113	に戻します。
9		

分数表示の例

次の表に、2 個の $\angle$ 値に対して、数値 2.77 が 3 つの分数表示形式でどのように表示されるかを示します。

分数形式	2.77 の表示方法	
	$\angle = 4095$	$\angle = 16$
最も高精度	$2\ 77/100^{(2.7700)}$	$2\ 10/13\blacktriangle^{(2.7692)}$
分母の約数	$2\ 1051/1365\blacktriangle^{(2.7699)}$	$2\ 3/4\blacktriangle^{(2.7500)}$
固定分母	$2\ 3153/4095\blacktriangle^{(2.7699)}$	$2\ 12/16\blacktriangle^{(2.7500)}$

次の表に、 $\angle$ 値 16 に対して、さまざまな数値が 3 つの分数表示形式でどのように表示されるかを示します。

分数表示形式 *	入力された数値と表示される分数				
	2	2.5	$2\ 2/3$	2.9999	$2\ 16/25$
最も高精度	2	$2\ 1/2$	$2\ 2/3\blacktriangle$	$3\blacktriangledown$	$2\ 9/14\blacktriangledown$
分母の約数	2	$2\ 1/2$	$2\ 11/16\blacktriangledown$	$3\blacktriangledown$	$2\ 5/8\blacktriangle$
固定分母	$2\ 0/16$	$2\ 8/16$	$2\ 11/16\blacktriangledown$	$3\ 0/16\blacktriangledown$	$2\ 10/16\blacktriangle$
* $\angle$ 値 16 の場合					

分数の丸め

分数表示モードがオンであれば、RND 関数は、X レジスタの数値を分数に対して最も近い小数に変換します。丸めは、現在の / ϵ 値とフラグ 8 と 9 の状態に従って実行されます。分数と変換後の小数が一致していれば、精度インジケータが消えます。そうでなければ、精度インジケータは点灯したままです（この章で前述の「確度インジケータ」を参照してください）。

方程式やプログラムでは、RND 関数は、分数表示モードがオンの場合に分数の丸めを実行します。

例：

56 $\frac{3}{4}$ インチの長さを 6 等分したいとします。 $\frac{1}{16}$ インチ増分で測定できるとして、各セクションの幅はいくらでしょうか？ そのときの累積丸め誤差はいくらでしょうか？

キー：	ディスプレイ：	説明：
 FLAGS  8		フラグ 8 をセットします
1 6  		$\frac{1}{16}$ インチ増分の分数表示形式を設定します。（フラグ 8 と 9 は、前の例と同じ状態であるとします）
5 6 . 3 . 4	56 $\frac{3}{4}$	距離を D に格納します。
 STO D		
6 	9 $\frac{7}{16}$ ▲	セクションは、9 $\frac{7}{16}$ インチより少し長くなります。
 RND	9 $\frac{7}{16}$	幅をこの値に丸めます。
6 x	56 $\frac{5}{8}$	6 分割の幅。
RCL D -	-0 $\frac{1}{8}$	累積丸め誤差。
 FLAGS 2 (2CF) 8	-0 $\frac{1}{8}$	フラグ 8 をクリアします。
 FDISP	-0.1250	分数表示モードをオフにします。

方程式内の分数

方程式の中で分数を使用できます。方程式が表示されるときは、式内のすべての数値が入力時の形式で表示されます。また、分数表示モードは、方程式を呼び出す演算にも有効です。

方程式を評価しているときや、変数の値の入力が求められたとき、分数を入力できます。値は、現在の表示形式を使って表示されます。

方程式の処理方法の詳細については、第 6 章を参照してください。

プログラム内の分数

式の場合と同様に、プログラム内でも分数を使用できます。数値が入力時の形式で表示されます。

プログラムの実行時は、分数表示モードがオンであれば、分数表示モードを使って分数が表示されます。INPUT 命令によって値の入力を求められた場合、分数を入力できます。プログラムの結果が、現在の表示形式を使って表示されます。

/c 関数の使用と、フラグ 7、8、9 のセットとクリアにより、分数表示形式を制御できます。第 14 章の「フラグ」を参照してください。

プログラムの処理方法については、第 13 章と 14 章を参照してください。

方程式の入力と評価

方程式の使いかた

HP 35s では方程式をさまざまな方法で使用できます。

- 評価する方程式の指定（この章）
- 未知の値を求める方程式の指定（第 7 章）
- 積分する関数の指定（第 8 章）

例：方程式による計算

まっすぐな断面を持つパイプの体積を頻繁に計算する必要があるとします。方程式を表示します。

$$V = .25 \pi d^2 l$$

ここで、 d はパイプの内径、 l は長さです。

- ✓ 何度も同じ計算を入力することは可能です。例えば、
 $\boxed{\cdot} \boxed{2} \boxed{5} \boxed{\text{ENTER}} \boxed{\pi} \boxed{\times} \boxed{2} \boxed{\cdot} \boxed{5} \boxed{x^2} \boxed{\times} \boxed{1} \boxed{6} \boxed{\times}$ を押すと、長さ 16 インチ、直径 $2\frac{1}{2}$ インチのパイプの体積を計算できます（78.5398 立方インチ）。しかし、**方程式** を格納すると、直径、長さ、体積の関係を HP 35s に記憶させることができるため、方程式を何度も使用できます。

電卓を方程式モードに設定し、次のキー入力を使って方程式を入力します。

キー:	ディスプレイ:	説明:
[EQN]	EQN LIST TOP または第 2 行に表示されている現在の方程式	方程式モードを選択します。 EQN インジケータによって示されます。
[RCL]		新しい式を開始します。 [RCL] により A..Z インジケータがオンになるため、変数名を入力できます。
[V] [←] [=] [.] [2] [5]	V=_ V=0.25_	[RCL] [V] で、V を入力します 数値入力は "_" 入力カーソルを使用します。
[×] [←] [π] [×]	V=0.25×π×_	[×] を押すと、その数値の入力が完了します。
[RCL] [D] [y^x] [2] [×] [RCL] [L] [ENTER]	V=0.25×π×D^2_ V=0.25×π×D^2×L_ V=0.25×π×D^2×L	[y^x] で、^ を入力します。
[←] [SHOW]	CK=49C8 LN=14	方程式が完了し、式が表示されます。 方程式のチェックサムと長さが表示されるため、キー入力を確認できます。

方程式のチェックサムと長さを例の方程式と比較することで、方程式が正しく入力されたことを確認できます。(詳細については、この章の最後の「方程式の確認」を参照してください)。

方程式の評価 (V の計算):

キー:	ディスプレイ:	説明:
[ENTER]	D? 値	方程式の右辺の変数を入力するプロンプトが出ます。最初に D のプロンプトが出ます。値は、D の現在の値です。
[2] [.] [1] [.] [2]	D? 2 1/2_	2 1/2 インチを分数として入力します。
[R/S]	L? 値	D を格納します。L の入力を求められます。値は、L の現在の値です。
[1] [6] [R/S]	V= 78.5398	L を格納し、体積 V を計算し、結果を V に格納します。

6-2 方程式の入力と評価

式の操作のまとめ

作成したすべての方程式は、**方程式リスト**に保存されます。このリストは、方程式モードをオンにすると表示されます。

方程式に関係する演算を実行するには、特定のキーを使用します。キーについては、後から詳細に説明します。

方程式リストの方程式を表示する場合、一度に 2 つの式が表示されます。現在アクティブな方程式は、第 2 行に表示されています。

キー	操作
	方程式モードを開始／終了します。
	表示中の方程式を評価します。方程式が 代入式 であれば、右辺を評価して結果を左辺の変数に格納します。方程式が 等価式 または 式 であれば、  キーと同様に値を計算します。（この章の「方程式の種類」を参照してください）。
	表示中の方程式を評価します。方程式中に "=" があれば "-" に変換して値を計算します。
	表示中の方程式から指定した未知の変数を求めます。（第 7 章を参照）。
 	表示中の方程式を、指定した変数で積分します。（第 8 章を参照）。
	表示中の方程式を削除するか、カーソルの左側の要素を削除します。
 または 	表示中の方程式の編集を開始します。カーソルを移動するだけで、内容の削除は行われません。
  または  	現在の方程式の表示画面をスクロールします。
 または 	方程式リスト内を上下に移動します。
  または  	方程式リストの一番上または下に移動します。
 	
 	表示中の方程式のチェックサム（確認のための値）と長さ（メモリのバイト数）を表示します。
 	最後に削除された要素または方程式を復元します。
	方程式モードを終了します。

方程式はプログラムでも利用できます。詳細については、第 13 章を参照してください。

方程式リストへの方程式の入力

方程式リストは、入力済みの方程式の集合です。リストは、電卓のメモリに保存されています。入力した各方程式は、方程式リストに自動的に保存されます。

方程式を入力するには：

方程式は、必要に応じて作成でき、個数は、空きメモリ容量によってのみ制限されます。

1. 電卓が通常の操作モードにあることを確認します。通常の操作モードでは、ディスプレイに数値が表示されています。例えば、変数またはプログラムのカタログを表示しておくことはできません。
2. **[EQN]** を押します。**EQN** インジケータによって方程式モードがアクティブであることが示され、方程式リストのエントリが表示されます。
3. 方程式の入力を開始します。これまで表示されていた方程式が移動して、入力中の方程式に置き換わります。これまでの方程式に影響はありません。入力ミスをした場合、必要に応じて **[←]** または **[↩] [UNDO]** を押します。
4. **[ENTER]** を押して方程式を終了し、ディスプレイに表示します。方程式は、方程式リスト（入力開始時に表示されていたエントリの直後）に自動的に保存されます。（代わりに **[C]** を押すと、方程式が保存されてから方程式モードがオフになります）。

方程式には変数、数値、ベクトル、関数、括弧を含めることができます。詳細については、以降のトピックで説明します。次の例でこれらの要素を示します。

方程式内の変数

方程式に電卓のすべての変数、A から Z、および **(I)** と **(J)** が使用できます。それぞれの変数に使用回数の制限はありません（**(I)** と **(J)** の詳細については、第 14 章「変数とラベルの間接アドレッシング」を参照してください）。

方程式に変数を入力するには、**[RCL] 変数** を押します。**[RCL]** を押すと、**A..Z** インジケータが表示され、変数キーを押して方程式に名前を入力できる状態になります。

方程式内の数値

式の中には有効な数値（2 進数、8 進数、16 進数、実数、複素数、分数）であればすべて入力できます。数値は常に表示形式 ALL で 12 文字まで表示されます。

方程式に数値を入力するには、、、 などの標準の数値入力キーを使用できます。減算には  を使用できません。

方程式内の関数

HP 35s の多数の関数を方程式に入力できます。関数の一覧は、この章の「方程式で利用できる関数」に示されています。この情報は、「付録 B 操作もくじ」にもあります。

方程式を入力する際、方程式に関数を入力する方法は、通常の代数方程式の場合とほぼ同じです。

- 方程式では、"+"、"÷" などの特定の関数は通常、**それらの引数間**に表示されます。こうした**中置**演算子の場合、演算子を方程式に同じ順番で入力します。
- その他の関数は通常、"COS"、"LN" などの関数名の**後**に 1 つ以上の引数を持ちます。こうした**前置**関数の場合、関数を、関数が発生する方程式に入力します。キーを押すと、関数名の後に左括弧が配置されるため、引数を入力できます。
- 関数に複数の引数がある場合、  を押して分離します。

方程式内の括弧

方程式に括弧を含めて、演算が実行される順序を制御できます。 $\boxed{()}$ を押して括弧を挿入します。(詳細については、この章の最後の「演算子の優先順位」を参照してください)。

例：方程式を入力する

方程式 $r = 2 \times c \times (t - a) + 25$ を入力します

キー：	ディスプレイ：	説明：
$\boxed{\text{EQN}}$	$V=0.25 \times \pi \times D^2 \times L$	方程式リストで最後に使用された式を表示します。
$\boxed{\text{RCL}} \boxed{R} \boxed{\text{CL}} \boxed{=}$	$R=_$	変数 R で新しい方程式を開始します。
$\boxed{2}$	$R= 2_$	数値を入力します
$\boxed{\times} \boxed{\text{RCL}} \boxed{C} \boxed{\times}$	$R=2 \times C \times _$	中置演算子を入力します。
$\boxed{()}$	$R=2 \times C \times (_$	前置演算子と左括弧を入力します。
$\boxed{\text{RCL}} \boxed{T} \boxed{-} \boxed{\text{RCL}}$		引数と右括弧を入力します。
$\boxed{A} \boxed{>} \boxed{+} \boxed{2} \boxed{5}$	$=2 \times C \times (T - A) + 25 _$	
$\boxed{\text{ENTER}}$	$R=2 \times C \times (T - A) + 25$	方程式を完了し、式を表示します。
$\boxed{\text{CL}} \boxed{\text{SHOW}}$	$CK=9E5F$ $LN=14$	チェックサムと長さを表示します。
$\boxed{C}$		方程式モードを終了します。

方程式の表示と選択

方程式リストには、 2×2 lin. solve と 3×3 lin. solve という 2 つの組み込み方程式、および入力した方程式があります。方程式を表示し、使用する方程式を選択できます。

方程式を表示するには：

1. **[EQN]** を押します。これにより方程式モードがオンになり、**EQN** インジケータが表示されます。ディスプレイには、方程式リストのエントリが表示されます。
 - **EQN LIST TOP**、方程式ポインタがリストのトップにある場合。
 - 現在の方程式（最後に表示していた方程式）。
2. **[^]** または **[v]** を押して、方程式リスト内を 1 ステップずつ移動し、各方程式を表示します。リストの最後に達すると、最初に戻ります。**EQN LIST TOP** は、リストのトップを示します。

長い式を表示するには：

1. 前述の手順に従って、方程式リストの方程式を表示します。14 文字を超える場合、14 文字だけ表示されます。➡ インジケータは、右にさらに文字があることを示します。
2. **[>]** を押して方程式の先頭から編集を開始するか、**[<]** を押して方程式の最後から編集を開始します。**[<]** または **[>]** を繰り返し押して、方程式内で 1 文字ずつカーソルを移動します。**◀** と **▶** は、左または右にさらに文字がある場合に表示されます。
3. **[F2][<]** または **[F2][>]** を押し、画面によって第 2 行に長い方程式をスクロールします。

方程式を選択するには：

前述の手順に従って、方程式リストの方程式を表示します。第 2 行に表示されている方程式が、すべての方程式演算に使用される方程式です。

例：方程式を表示する

入力した最後の方程式を表示します。

キー：	ディスプレイ：	説明：
[EQN]	$R=2 \times C \times (T-A) + 25$	方程式リストで現在の方程式を表示します。
[>]	$R=2 \times C \times (T-A) + 25$	方程式の左端のカーソルがオンになります
[ENTER][<]	$= 2 \times C \times (T-A) + 25$	方程式の右端のカーソルがオンになります
[C]		方程式モードを終了します。

方程式の編集とクリア

入力中の方程式を編集またはクリアできます。方程式リストに保存されている方程式も、編集またはクリアできます。ただし、 2×2 lin. solve と 3×3 lin. solve という 2 つの組み込み方程式は、編集／クリアできません。2 つの組み込み方程式間に方程式を挿入しようとする、新しい方程式が 3×3 lin. solve の後に挿入されます。

入力中の方程式を編集するには：

1. $\leftarrow$ または $\rightarrow$ を押して、カーソルを移動します。カーソルの前に文字を挿入できます。
2. カーソルを移動し、 $\leftarrow$ を繰り返し押して、不要な数値または関数を削除します。方程式の編集行に何も入っていないときは $\leftarrow$ を押しても何も変化はありませんが、空の方程式行で ENTER を押すと、空の方程式行が削除されます。ディスプレイには、方程式リストの前のエントリが表示されます。
3. ENTER (または C) を押して、方程式を方程式リストに保存します。

保存済みの方程式を編集するには：

1. 目的の方程式を表示し、 $\rightarrow$ を押して方程式の先頭でカーソルをアクティブにするか、 $\leftarrow$ を押して方程式の最後でカーソルをアクティブにします（前述の「方程式の表示と選択」を参照してください）。
2. 方程式でカーソルがアクティブのとき、新しい方程式の入力時と同様に方程式を編集できます。
3. ENTER (または C) を押して、編集済みの方程式を方程式リストに保存し、前のバージョンと置き換えます。

方程式の編集中にメニューを使用する

1. 方程式の編集時に設定メニュー (MODE 、 DISPLAY 、 CLEAR など) を選択すると、方程式編集状態が終了します。
2. 方程式の編集時に挿入メニューまたは閲覧メニュー (L.R. 、 X,Y 、 S.O. 、 SUMS 、 BASE 、 LOGIC 、 R 、 MEM 、 CONST など) を選択すると、方程式は、アイテムの挿入後に編集モードのままになります。
3. 方程式モードでは、メニュー X?Y 、 FLAGS 、 X?0 は利用できません。

6-8 方程式の入力と評価

保存済みの方程式をクリアするには：

方程式リストをスクロールして目的の方程式をディスプレイの第 2 行に表示させ、 を押します。

すべての保存済みの方程式をクリアするには：

EQN モードで、 **CLEAR** を押します。**3** (3EQN) を選択します。CLR EQN? Y N メニューが表示されます。 (Y) **ENTER** を選択します。

例：方程式を編集する

前の例の方程式で 25 を削除します。

キー：	ディスプレイ：	説明：
	$R=2 \times C \times (T-A) + 25$	方程式リストで現在の方程式を表示します。
	$=2 \times C \times (T-A) + 25 _$	方程式の最後でカーソルがオンになります
  	$=2 \times C \times \cos(T-A) _$	数値 25 を削除します。
	$R=2 \times C \times (T-A)$	方程式リストで編集済みの方程式の最後を表示します。
		方程式モードを終了します。

方程式の種類

HP 35s では 3 種類の方程式が利用できます。

- **等式**。方程式には 1 個の "=" が含まれ、左辺には 1 つ以上の変数が含まれます。例えば、 $x^2 + y^2 = r^2$ は**等式**です。
- **代入式**。方程式には 1 個の "=" が含まれ、左辺には変数が 1 つだけ含まれます。例えば、 $A = 0.5 \times b \times h$ は**代入式**です。
- **式**。方程式には "=" が含まれません。例えば、 $x^3 + 1$ は**式**です。

方程式により計算するときは、すべての種類の方程式が使用できますが、種類によって評価方法が異なります。未知の変数に対する問題を解くときは、通常、等式か代入式を使用します。関数を積分するときは、通常、式を使用します。

方程式の評価

方程式の最も便利な特性の 1 つが、数値を生成する**評価機能**です。これにより、方程式から結果を計算できます。(方程式を解いたり、積分することもできます。第 7 章と 8 章を参照してください)。

多くの方程式では左辺と右辺が "=" で分離されているため、方程式の基本値は、左辺と右辺の値の**差**になります。この計算では、方程式内の "=" が、基本的に "-" として処理されます。この値は、方程式の左右のバランスの評価基準です。

HP 35s には方程式を評価するための **ENTER** と **XEQ** の 2 つのキーがあります。キーの動作に違いがあるのは、**代入方程式**の評価方法の場合のみです。

- **XEQ** は、方程式の種類に関わらず、その方程式の値を返します。
- **ENTER** は、**代入タイプ**の方程式でない**限り**、方程式の値を返します。代入方程式の場合、**ENTER** は、右辺の値のみを返し、その値を左辺の変数にも「入力」します。値を変数に格納します。

次の表に、方程式を評価する 2 つの方法を示します。

方程式の種類	[ENTER] の結果		[XEQ] の結果
等式 : $g(x) = f(x)$ 例 : $x^2 + y^2 = r^2$	$g(x) - f(x)$ $x^2 + y^2 - r^2$		
代入式 : $y = f(x)$ 例 : $A = 0.5 \times b \times h$	$f(x)^*$ $0.5 \times b \times h^*$	$y - f(x)$ $A - 0.5 \times b \times h$	
式 : $f(x)$ 例 : $x^3 + 1$	$f(x)$ $x^3 + 1$		
* 結果が左辺の変数（この例では A）にも格納されます。			

方程式を評価するには：

1. 目的の方程式を表示します。(前述の「方程式の表示と選択」を参照)。
2. **[ENTER]** または **[XEQ]** を押します。方程式が、必要な各変数の値を入力するよう求めます。(方程式の数値の基数が現在の基数と異なる場合、電卓が結果を現在の基数に自動的に変更します)。
3. それぞれのプロンプトで、求められる値を入力します。
 - 表示された値が正しければ、**[R/S]** を押します。
 - 別の値を希望する場合、値を入力し、**[R/S]** を押します。(この章の「方程式のプロンプトへの応答」も参照してください)。

計算を中断するには、**[C]** または **[R/S]** を押します。第 2 行にメッセージ INTERRUPTED が表示されます。

方程式の評価では、スタックから値を取り込みません。方程式内の数値と変数値のみを使用します。方程式の値は、X レジスタに戻されます。

評価のための ENTER の使用

方程式リストに方程式が表示されている場合、**[ENTER]** を押して方程式を評価できます。(方程式の入力中に **[ENTER]** を押しても、方程式を終了するだけで、式を評価しません)。

- 方程式が **代入式**であれば、右辺のみが評価されます。結果は X レジスタに返され、左辺変数に格納された後、変数がディスプレイに表示されません。基本的に、**[ENTER]** は、左辺変数の値を検出します。
- 方程式が**等価式**または**式**であれば、**[XEQ]** の場合と同様に、方程式全体が評価されます。結果は、X レジスタに戻されます。

例：ENTER で方程式を評価する

この章の最初の方にある、長さ 20 m、直径 35-mm のパイプの体積を求める方程式を使用します。

キー：	ディスプレイ：	説明：
EQN (必要であれば ↵) ENTER	$V = 0.25 \times \pi \times D^2 \times L$ D? 2.5	目的の方程式を表示します。
3 5 R/S	L? 16	代入式の評価を開始し、値が V に格納されるようにします。方程式の右辺の変数を入力するプロンプトが出ます。D の現在の値は 2.5 です。
2 0 × 1 0 0 0 ENTER R/S	V= 19,242,255.0033	D を格納します。L の入力 が促されます。現在の値は 16 です。
÷ 1 E 6 ENTER	19.2423	L (単位ミリメートル) を格納し、V (単位ミリ立方メートル) を計算し、結果を V に格納して、V を表示します。ミリ立方メートルをリットルに変換します (✓ は変更しません)

評価のための XEQ の使用

方程式リストに方程式が表示されている場合、**[XEQ]** を押して方程式を評価できます。方程式の種類に関わらず、方程式全体が評価されます。結果は、X レジスタに戻されます。

例：XEQ で方程式を評価する

前の例の結果を使用して、パイプの直径を 35.5 mm に変更したときにパイプの体積がどれだけ変化するかを求めます。

キー：	ディスプレイ：	説明：
[EQN]	$V = 0.25 \times \pi \times D^2 \times L$	目的の方程式を表示します。
[XEQ]	V?	値を求めるため方程式の評価を開始します。 すべての 変数の入力を求めます。
[R/S]	D?	V はそのままにします。D の入力を求めます。
[3] [5] [.] [5]	L?	新しい D を格納します。L の入力を求めます。
[R/S]	20.000	
[R/S]	-553.705.7051	L はそのままにします。方程式の値（左辺と右辺間の不均衡）を計算します。
[÷] [1] [E] [6]	-0.5537	ミリ立方メートルをリットルに変換します。
[ENTER]		

方程式の値は、(V からの) 古い体積引く (新しい D 値を使用して計算された) 新しい体積で、古い体積は、示された大きさだけ小さくなります。

方程式のプロンプトへの応答

方程式を評価するとき、必要な各変数の値を入力するよう求められます。プロンプトは、 $X \div 2.5000$ など、変数名と現在の値を示します。名前のない間接変数 (I) または (J) が方程式にある場合、名前のない間接変数に格納された現在の値が自動的に使用されるため、値に対するプロンプトが表示されません。(第 14 章を参照)。

- 数値をそのままにするには、**[R/S]** を押します。
- 数値を変更するには、新しい数値を入力し、**[R/S]** を押します。新しい数値は X レジスタの古い値に上書きされます。必要に応じて数値を分数としても入力できます。数値を計算する必要がある場合、通常のキーボード計算を使用してから、**[R/S]** を押します。例えば、RPN モードでは **2 [Y^x] 5 [ENTER] [R/S]**、ALG モードでは **2 [Y^x] 5 [ENTER] [R/S]** を押すことができます。**[ENTER]** を押す前には、式が第 2 行に表示され、**[ENTER]** を押した後には、式の結果が第 2 行に表示されます。
- プロンプトをキャンセルするには、**[C]** を押します。変数の現在の値は X レジスタ内で同じままで、第 2 行の右辺に表示されます。数値の入力中に **[C]** を押すと、数値がゼロにクリアされます。**[C]** を再度押して、プロンプトをキャンセルします。
- プロンプトによって隠された数字を表示するには、**[☞] [SHOW]** を押します。

RPN モードでは、プロンプトが出るたびに変数の現在値が X レジスタにコピーされますが、スタックの移動は発生しません。X プロンプトに従って数値を入力すると、その値は X レジスタに入ります。**[R/S]** を押すとスタックの移動が発生し、値はスタックに保存されます。

方程式の文法

方程式は、評価方法を決定する特定の規約に従います。

- 演算子の間の相互作用。
- 方程式内で有効な関数
- 方程式に対する文法エラーのチェック

演算子の優先順位

方程式内の演算子は特定の順序で評価され、論理的で予測可能な評価が行われます。

順番	操作	例
1	括弧	$(X+1)$
2	関数	$SIN(X+1)$
3	指数 (y^x)	X^3
4	単項マイナス ($+/-$)	$-A$
5	乗算と除算	$X \times Y, A \div B$
6	加算と減算	$P+Q, A-B$
7	等式	$B=C$

例えば、括弧の中のすべての演算が実行された後に、括弧の外の演算が実行されます。

例：

方程式	意味
$A \times B^3 = C$	$a \times (b^3) = c$
$(A \times B)^3 = C$	$(a \times b)^3 = c$
$A + B \div C = 12$	$a + (b/c) = 12$
$(A + B) \div C = 12$	$(a + b) / c = 12$
$\%CHG(T+12, A-6)^2$	$[\%CHG((t + 12), (a - 6))]^2$

方程式で使用できる関数

次の表に、方程式で有効な関数を示します。この情報は、付録 G「操作もくじ」にもあります。

LN	LOG	EXP	ALOG	SQ	SQRT
INV	IP	FP	RND	ABS	!
SGN	INTG	IDIV	RMDR		
SIN	COS	TAN	ASIN	ACOS	ATAN
SINH	COSH	TANH	ASINH	ACOSH	ATANH
→DEG	→RAD	HMS→	→HMS	%CHG	XROOT
→L	→GAL	→MILE	→KM	nCr	nPr
→KG	→LB	→°C	→°F	→CM	→IN
SEED	ARG	RAND	π	$\wedge$	
+	-	$\times$	$\div$	$\bar{x}$	$\bar{y}$
s_x	s_y	σx	σy	$\bar{x}$	$\bar{y}$
$\bar{x}_w$	$\hat{x}$	$\hat{y}$	r	m	b
n	Σx	Σy	Σx^2	Σy^2	Σxy

1 つまたは 2 つの引数をとる前置関数を入力すると、自動的に左側の括弧が表示されます。

2 つの引数をとる前置関数は、%CHG、XROOT、IDIV、RMDR、nCr、nPr です。引数の間はカンマで区切ります。

✓ 方程式では、XROOT 関数の引数の順番が RPN モードと反対になります。例えば、 -8 **[ENTER]** 3 **[$\sqrt[3]{}$]** は、 $XROOT(3, -8)$ と等価です。

✓ 2 つの引数をとるその他の関数もすべて、RPN で使用する場合は引数の順番は Y, X になります。例えば、 28 **[ENTER]** 4 **[$\leftarrow$]** **[nCr]** は、 $nCr(28, 4)$ と等価です。

2 つの引数をとる関数では、第 2 の引数が負の値の場合は注意してください。これらは有効な式です。

%CHG($-X$, -2)

%CHG(X , $(-Y)$)

6-16 方程式の入力と評価

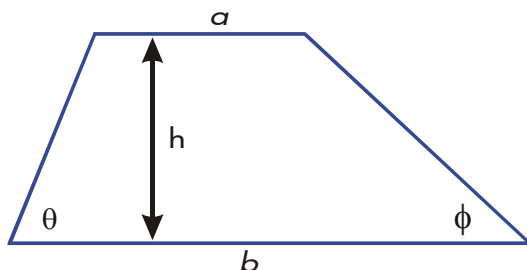
次の 8 個の関数には、等価の操作とは異なる名前が付いています。

RPN 操作	方程式での関数
x^2	SQ
$\sqrt{x}$	SQRT
e^x	EXP
10^x	ALOG
$1/x$	INV
$\sqrt[x]{y}$	XROOT
y^x	^
INT÷	IDIV

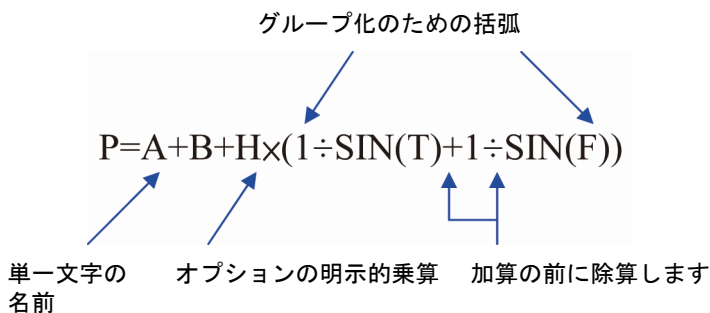
例：台形の辺の長さ

次の方程式で台形の辺の長さを計算できます。教科書には次のように記載されています。

$$\text{辺の長さ} = a + b + h \left(\frac{1}{\sin \theta} + \frac{1}{\sin \phi} \right)$$



次の方程式は、HP 35s の方程式の文法ルールに従います。



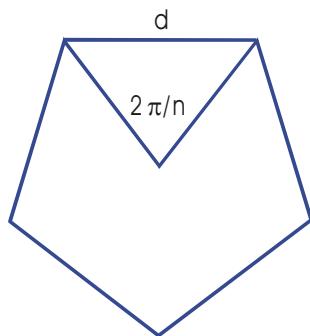
次の方程式も、文法ルールに従っています。この方程式は、分数形 $1 \div \text{SIN}(T)$ の代わりに逆関数 $\text{INV}(\text{SIN}(T))$ を使用しています。SIN 関数は INV 関数の中に「入れ子（ネスト）」になっている点に注意してください。（INV は $\boxed{1/x}$ で入力します）。

$$P=A+B+H \times (\text{INV}(\text{SIN}(T)) + \text{INV}(\text{SIN}(F)))$$

例：多角形の面積

一辺の長さが d である正 n 角形の面積は、次のようになります。

$$\text{面積} = \frac{1}{4} n d^2 \frac{\cos(\pi/n)}{\sin(\pi/n)}$$



この方程式を次のように指定できます。

$$A=0.25 \times N \times D^2 \times \cos(\pi \div N) \div \sin(\pi \div N)$$

目的の方程式を入力するとき、演算子と関数をどう組み合わせるか確認してください。

次のキー入力を使って方程式を方程式リストに入力できます。

EQN **RCL** **A** **↵** **=** **·** **2** **5** **×** **RCL** **N** **×** **RCL** **D** **y^x** **2** **×**
COS **↵** **π** **÷** **RCL** **N** **>** **÷** **SIN** **↵** **π** **÷** **RCL** **N** **ENTER**

文法エラー

方程式の文法は、評価のときに初めてチェックされます。エラーがあれば、**SYNTAX ERROR** が表示され、最初のエラーが発生した位置にカーソルが表示されます。方程式を編集してエラーを補正してください。（この章で前述の「方程式の編集とクリア」を参照してください）。

HP 35s では、評価までは方程式の文法がチェックされないで、実際はメッセージである「方程式」を作成できます。これは、プログラムで特に便利です。第 13 章を参照してください。

方程式の確認

方程式を閲覧しているとき（入力しているときではありません）、**↵** **SHOW** を押して、方程式のチェックサムと長さの 2 点を表示できます。値をディスプレイに表示し続けるには、**SHOW** キーを押したままにします。

チェックサムは、その方程式を一意に識別する 4 桁の 16 進数です。方程式を間違えて入力したときは、チェックサムが異なってきます。長さは、その方程式によって使用される電卓メモリのバイト数です。

チェックサムと長さにより、入力した方程式が正しいかどうかを確認できます。例で入力した方程式のチェックサムと長さは、ここで示す値と一致するはずです。

例：方程式のチェックサムと長さ

この章の最初の方にある、パイプの体積を求める方程式のチェックサムと長さを確認します。

キー：	ディスプレイ：	説明：
	$V = 0.25 \times \pi \times D^2 \times L$	目的の方程式を表示します。
(必要であれば )		
  (hold)	CK=49CA LN=14	方程式のチェックサムと長さ を表示します。
(離す)	$V = 0.25 \times \pi \times D^2 \times L$	方程式を再表示します。
		方程式モードを終了します。

方程式を解く

第 6 章では、**[ENTER]** を使用して、**代入式** タイプの方程式の左辺変数の値を求める方法を説明しました。SOLVE を使えば、**すべての** 種類の方程式の、**すべての** 変数を計算できます。

例えば、次の方程式を考えます。

$$x^2 - 3y = 10$$

方程式の y が既知であれば、SOLVE で未知数 x を求めることができます。 x が既知であれば、SOLVE で未知数 y を求めることができます。次のような文章問題にも利用できます。

$$\text{利掛け} \times \text{仕入原価} = \text{売価}$$

3 つの変数のうち、任意の 2 変数がわかれば、SOLVE で残りの変数の値を計算できます。

方程式に変数が 1 つしかない場合や、未知の変数が 1 つのみの場合は、 x を解くため、その方程式の **根** を計算します。方程式の根を計算できるのは、**等価式** や **代入式** で左辺と右辺が完全に等しいときと、**式がゼロ** のときになります。

方程式を解く方法

未知の変数を求めるために方程式を解くには（組込み関数を除く）：

1. **[EQN]** を押し、目的の方程式を表示します。必要に応じて、第 6 章「方程式リストへの方程式の入力」の説明に従って方程式を入力します。
2. **[ SOLVE]** を押し、未知の変数のキーを押します。例えば x を計算するには、**[ SOLVE] X** を押します。方程式にその他の変数があれば、その値を入力するよう求められます。
3. それぞれのプロンプトで、必要な値を入力します。
 - 表示された値が正しければ、**[R/S]** を押します。
 - 別の値を希望する場合、値を入力するか計算し、**[R/S]** を押します。（詳細については、第 6 章「方程式のプロンプトへの応答」を参照してください）。

計算を中断するには、**[C]** または **[R/S]** を押します。

根が見つければ、関係変数に格納され、変数値がディスプレイに表示されます。さらに、X レジスタにはその根が、Y レジスタには直前の推定値またはゼロが、Z レジスタには D 値の平方根（これはゼロになるはず）が格納されます。

数学的に込み入った状態では、最終的な解が見つからないことがあり、NO ROOT FOUND が表示されます。この章で後述の「結果の確認」、および付録 D の「結果の解釈」と「SOLVE が根を発見できなかったときは」を参照してください。

特定の方程式では、方程式を解く前に未知の変数用に 1 つまたは 2 つの**初期推定値**を提供すると有効です。これにより、計算速度が向上し、答えを現実的な解に近づけることができ、状況によっては複数の解を発見できます。この章で後述の「SOLVE の初期推定値の選択」を参照してください。

例：線形運動方程式を解く

物体の自由落下の運動方程式は、次のようになります。

$$d = v_0 t + \frac{1}{2} g t^2$$

d は距離、 v_0 は初速度、 t は時間、 g は重力加速度です。

方程式を入力します。

キー：	ディスプレイ：	説明：
(3ALL)		メモリをクリアします。
(Y)		
	3*3 lin. solve EQN LIST TOP	方程式モードを選択します。
		方程式を開始します。
	D=V×T+_	
	←=V×T+0.5×G×T^2_	
	D=V×T+0.5×G×T^2	方程式を完了し、左端を表示します。
	CK=FB3C LN=15	チェックサムと長さ。

g （重力加速度）は、単位系が変化したとき変更できるように、変数として含まれています（9.8 m/s² または 32.2 ft/s²）。

静止状態から 5 秒間落下したときの移動距離をメートル単位で求めます。すでに方程式モードがオンで、目的の方程式が表示されている状態であるため、 D の解を求める作業を開始できます。

キー:	ディスプレイ:	説明:
	SOLVE_	未知の変数を入力するよう求めます。
	V? 値	D を選択し、 V の入力求めます。
 	T? 値	0 を D に格納し、 T の入力求めます。
 	G? 値	5 を T に格納し、 G の入力求めます。
   	SOLVING D= 122.5000	9.8 を G に格納し、 D の解を求めます。

同じ方程式で別の問題を解いてみます。ある物体が静止状態から 500 m 落下するまでにかかる時間を求めます。

キー:	ディスプレイ:	説明:
	$D = V \times T + 0.5 \times G \times T^2$	方程式を表示します。
 	D? 122.5	T の解を求めるため、 D の入力求めます。
   	V? 0	500 を D に格納し、 V の入力求めます。
	G? 9.8	0 を V に保持し、 G の入力求めます。
	SOLVING T= 10.1015	9.8 を G に保持し、 T の解を求めます。

例：理想気体の法則方程式を解く

理想気体の法則は、理想気体の圧力、体積、温度、量（モル）の関係を記述します。

$$P \times V = N \times R \times T$$

P は圧力（気圧または N/m^2 ）、 V は体積（リットル）、 N は気体のモル数、 R は一般気体定数（ $0.0821 \text{ liter-atm/mole-K}$ or 8.314 J/mole-K ）、 T は温度（ケルビン： $\text{K} = ^\circ\text{C} + 273.1$ ）です。

方程式を入力します。

キー：	ディスプレイ：	説明：
[EQN] [RCL] [P] [X]	$P \times _$	方程式モードを選択し、方程式を開始します。
[RCL] [V] [←] [=]		
[RCL] [N] [X]		
[RCL] [R] [X] [RCL] [T]	$P \times V = N \times R \times T _$	
[ENTER]	$P \times V = N \times R \times T$	方程式が完了し、式が表示されます。
[←] [SHOW]	$CK=EDC8$ $LN=9$	チェックサムと長さ。

2 リットルの容器に 0.005 mol の二酸化炭素が入っていて、その温度は 24°C です。この二酸化炭素を理想気体として、圧力を計算してください。すでに方程式モードがオンで、目的の方程式が表示されている状態であるため、 P の解を求める作業を開始できます。

キー：	ディスプレイ：	説明：
[2] [SOLVE] [P]	$V?$	P の解を求めるため、 V の入力を求めます。
[2] [R/S]	値	
	$N?$	2 を V に格納し、 N の入力を求めます。
	値	
[.] [0] [0] [5] [R/S]	$R?$	.005 を N に格納し、 R の入力を求めます。
	値	

0 8 2 1 R/S

T?

.0821 を R に格納し、 T の入力求めます。

2 4 + 2 7 3 0

T?

T (ケルビン) を計算します。

1 ENTER

297.1000

R/S

SOLVING

P=

297.1 を T に格納し、

0.0610

P の解を気圧で求めます。

5 リットルのフラスコに窒素が入っています。気体の温度が 18°C のとき、圧力は 0.05 気圧です。気体の密度を計算します ($N \times 28/V$, 28 は窒素の分子量)。

キー:

ディスプレイ:

説明:

EQN

$P \times V = N \times R \times T$

方程式を表示します。

→ SOLVE N

P?

N の解を求めるため、

0.0610

P の入力求めます。

0 0 5 R/S

V?

.05 を P に格納し、 V の入力求めます。

2.0000

5 R/S

R?

5 を V に格納し、 R の入力求めます。

0.0821

R/S

T?

前の R を保持し、 T の入力求めます。

297.1000

✓ 1 8 ENTER 2 7 3

T?

T (ケルビン) を計算します。

0 1 +

291.1000

R/S

SOLVING

291.1 を T に格納し、

N=

N の解を求めます。

0.0105

✓ 2 8 ×

0.2929

グラム単位の質量、 $N \times 28$ を計算します。

✓ RCL V ÷

0.0586

g / liter で密度を計算します。

組み込みの方程式を解く

組み込みの方程式は、“2*2 lin. solve” ($Ax+By=C$, $Dx+Ey=F$) と “3*3 lin. Solve” ($Ax+By+Cz=D$, $Ex+Fy+Gz=H$, $Ix+Jy+Kz=L$) の 2 つあります。どちらかを選択した場合、**[XEQ]**、**[ENTER]**、**[/]** キーは効力がなくなります。**[F2]** **[SOLVE]** を押すと、2*2 の場合は 6 つの変数 (A ~ F)、3*3 の場合は 12 の変数 (A ~ L) を要求します。これらの変数を使用して 2*2 連立一次方程式の場合は x と y を、3*3 連立一次方程式の場合は x 、 y 、 z を求めます。結果は、変数 x 、 y 、 z に格納されます。解の数が無限になる場合や、解のない場合でも取り扱うことができます。

例：次の連立方程式で x と y の解を求めます
$$\begin{cases} x+2y=5 \\ 3x+4y=11 \end{cases}$$

キー：	ディスプレイ：	説明：
[EQN]	3*3 lin. solve	方程式モードを入力します。
[v]	EQN LIST TOP	組み込みの方程式を表示します。
[F2] [SOLVE]	2*2 lin. solve	A の入力求めます。
[1] [R/S]	R?	
	値	
[2] [R/S]	B?	1 を A に格納し、B の入力求めます。
	値	
[5] [R/S]	C?	2 を B に格納し、C の入力求めます。
	値	
[3] [R/S]	D?	5 を C に格納し、D の入力求めます。
	値	
[4] [R/S]	E?	3 を D に格納し、E の入力求めます。
	値	
[1] [1] [R/S]	F?	4 を E に格納し、F の入力求めます。
	値	
[v]	X=	↑ 11 を F に格納し、 x と y を計算します。
	1.0000	↓
	Y=	↑ y の値
	2.0000	↓

SOLVE の理解と制御

SOLVE は最初に、方程式で未知の変数を直接解くことを試みます。それができなかった場合は、SOLVE が反復（繰り返し）プロセスに変わります。プロセスではまず、未知の変数の 2 つの初期推定値を使用して方程式を評価します。SOLVE は、これら 2 つの推定値に基づいてさらに良い推定値を生成します。その繰り返しによって、方程式の値がゼロになるような未知数の値を求めます。

SOLVE が方程式を評価するときは、**[XEQ]** の場合と同様、方程式内のすべての "=" を "-" として処理します。例えば、理想気体の法則方程式は、 $P \times V - (N \times R \times T)$ として評価されます。これにより、**等価式**や**代入式**の左辺と右辺が根で釣り合い、**式**が根でゼロに等しくなります。

解くことが困難な方程式もあります。場合によって、解を見つけるために初期推定値を入力する必要があります。（後述の「SOLVE の初期推定値の選択」を参照してください）。SOLVE が解を発見できなかった場合、電卓が NO ROOT FND と表示します。

SOLVE の動作のしくみの詳細については、付録 D を参照してください。

結果の確認

SOLVE の計算が終了したら、その結果が方程式の解であるか確認するため、スタックに残された値を確認できます。

- X レジスタ (**[C]** を押して表示変数をクリアします) には計算した未知数の解（根）が含まれます。これは、方程式の評価をゼロにする値です。
- ✓ ■ Y レジスタ (**[R↓]** を押します) には、根の前の推定値が含まれるか、またはゼロに等しくなります。この数値は、X レジスタの値と同じになるはずです。もし同じでなければ、返された根は単なる**近似値**で、X レジスタと Y レジスタの値により根が囲まれます。これらの根を囲む数値は、互いに近い値である必要があります。
- ✓ ■ Z レジスタ (**[R↓]** を再度押します) には、根での方程式の D 値が含まれます。正確な根を得るには、これがゼロである必要があります。ゼロでない場合、根は単なる**近似値**ですが、この数値はゼロに近い値である必要があります。

計算が NO ROOT FND で終わったときは、根に収束できませんでした。(メッセージをクリアするため **C** または **←** を押すと、根の最終推定値である X レジスタの値を表示できます)。X レジスタと Y レジスタの値により、根の計算のため最後に検索された区間が囲まれます。Z レジスタには、根の最終推定値での方程式の値が含まれます。

- X レジスタと Y レジスタの値が互いに遠くない場合や、Z レジスタの値がゼロに遠くない場合は、X レジスタの推定値が根でない可能性があります。
- X レジスタと Y レジスタの値が互いに近い場合や、Z レジスタの値がゼロに近い場合は、X レジスタの推定値は根の近似値です。

SOLVE 計算の中断

計算を中断するには、**C** または **R/S** を押します。メッセージ "INTERRUPTED" が表示されます。現在の最も良い根の推定値が未知変数に存在します。**↩** **VIEW** を使用すると、スタックを阻害せずにこの推定値を確認できますが、SOLVE を再開できなくなります。

SOLVE の初期推定値の選択

2 つの初期推定値は、次から来ます。

- 現在、未知変数に格納されている数値
- X レジスタの数値 (ディスプレイ)

推定値を入力したかどうかに関わらず、これらのソースが使用されます。1 つの推定値のみを入力し、変数に格納した場合、第 2 の推定値は同じ値になります。ディスプレイにも、変数に格納した数値が保持されるからです。(このような場合、電卓は片方の推定値を少し変更して、2 つ推定値が得られるようにします)。

独自の推定値を入力すると、次の利点があります。

- 検索範囲を狭めることで、解を見つけるための時間が短縮します。
- 数学的な解が複数ある場合、推定値により SOLVE プロシージャに目的の答えまたは答えの範囲を指示できます。例えば、線形な運動の方程式について考えてみます。

$$d = v_0 t + \frac{1}{2} g t^2$$

は、 t に対する解が 2 つあります。適切な推定値を入力することで、答えを必要な解に誘導できます。

この章の最初に示した、この方程式を使用した例では、 T の解を求める前に推定値を入力する必要がありませんでした。例の前半で T の値を格納し、 **D の解を求めたからです**。 T に残った値は良い（現実的な）値であったため、 T の解を求めるときに推定値として使用されました。

- 方程式で、未知変数に対して特定の値が許可されない場合、推定値によってこうした値の発生を防止できます。例えば、

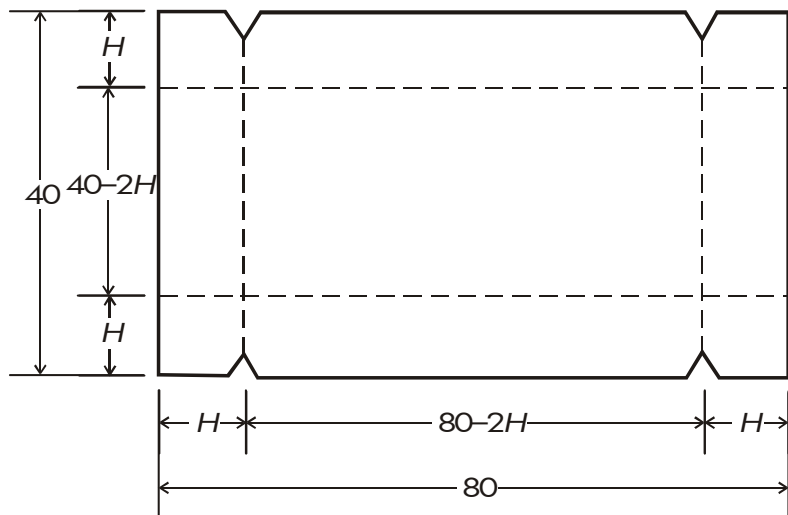
$$y = t + \log x$$

は、 $x \leq 0$ の場合エラーになります（メッセージ NO ROOT FND）。

次の例では、方程式の根が複数ありますが、推定値によって求める根が見つかりやすくなります。

例：推定値を使用して根を見つける

幅 40 cm、長さ 80 cm の長方形の金属板を使用して、 7500 cm^3 の容積を持つフタのない箱を作成します。指定の容積が得られる箱の高さ（4 辺をどれだけ折るか）を求める必要があります。高さの低い箱より高い箱を推奨します。



H を高さとして、箱の長さは $(80 - 2H)$ 、幅は $(40 - 2H)$ です。容積 V は、

$$V = (80 - 2H) \times (40 - 2H) \times H$$

簡略化し、次のように入力します。

$$V = (40 - H) \times (20 - H) \times 4 \times H$$

方程式を入力します。

キー：

ディスプレイ：

説明：

EQN

$V =$

方程式モードを選択し、方程式を開始します

RCL **V** **←** **=**

() **4** **0** **-**

$V = (40 - H)$

RCL **H** **>**

× **()** **2** **0** **-**

$(40 - H) \times (20 - H)$

RCL **H** **>**

× **4** **×** **RCL** **H**

$H) \times (20 - H) \times 4 \times H$

ENTER
 $V = (40 - H) \times (20 - H)$ 方程式が完了し、式が表示されます。
SHOW
 $CK = 49R4$
 $LN = 19$

チェックサムと長さ。

目的の容積を持つ、背が高くて幅が狭い箱と、背が低くて平らな箱が考えられます。背の高い箱が欲しいので、初期推定値に大きな値を指定します。しかし金属板の幅は 40 cm なので、高さは、20 cm を超えることができません。このため初期推定値には、10 cm と 20 cm が適しています。

キー:

ディスプレイ:

説明:

C

方程式モードを終了します。

1 0 **→** **STO** **H**

下限推定値と上限推定値を格納します。

ENTER **2 0**

20_

EQN $V = (40 - H) \times (20 - H)$

現在の方程式を表示します。

→ **SOLVE** **H**

V?

H の解を求めるため、V の入力値を求めます。

7 5 0 0 **R/S**

H=

7500 を V に格納し、H の解を求めます。

15.0000

この解の品質、つまり、正確な根が返ったかどうかをチェックします。チェックするには、Y レジスタに格納された直前の根の推定値と Z レジスタに格納された根での方程式の値を確認します。

キー:

ディスプレイ:

説明:

✓ **R↓**

15.0000

Y レジスタの値は、最終結果を出す直前の推定値です。最終結果と解が一致しているので、解は正確な根です。

✓ **R↓**

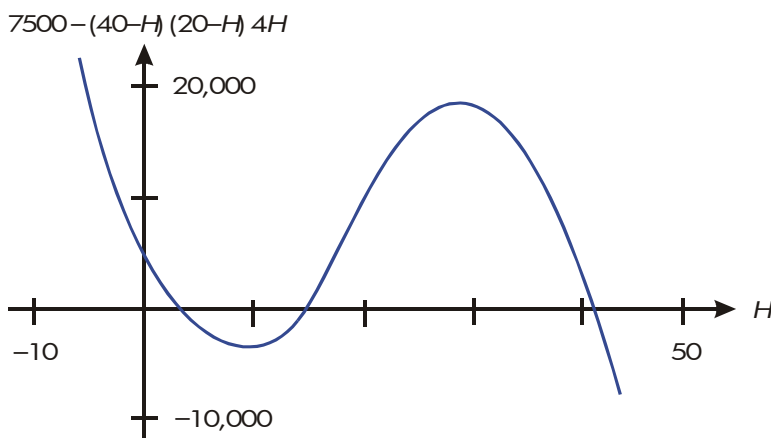
0.0000

Z レジスタの値は、方程式が根でゼロであることを示します。

目的の箱の寸法は、 $50 \times 10 \times 15$ cm です。高さの上限 (20 cm) を無視して、初期推定値に 30 cm と 40 cm を使えば、高さは 42.0256 cm になりますが、これは条件に合致しない根です。0 cm、10 cm などの小さい初期推定値を使用した場合、高さは 2.9774 cm になりますが、背の低い、フラットな箱になります。

7-12 方程式を解く

どのような推定値が良いか判断しにくいときは、グラフを使うと方程式の振る舞いが理解しやすくなります。方程式に複数の未知の値を当てはめます。グラフの各ポイントで、方程式を表示し、**XEQ** — を押して、 x のプロンプトで x -座標, を入力し、方程式の対応する値、 y -座標を取得します。上記の問題の場合、常に $V = 7500$ を設定し、 H の値を変化させて、方程式に対して異なる値を生成します。方程式の値は、右辺と左辺の差になるという点に注意してください。方程式の値をプロットすると、次の図のようになります。



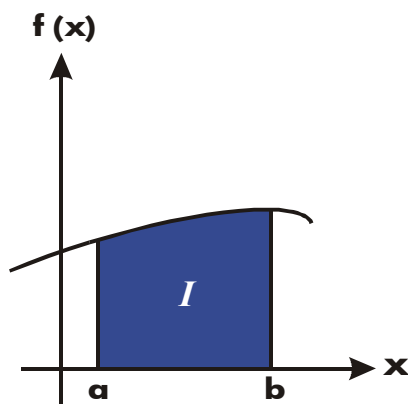
詳細

この章では、広範囲のアプリケーションで未知数や根の解を求める手順について説明しました。付録 D で、SOLVE のアルゴリズム、結果の解釈、解が見つからないときの原因、正しくない結果を引き起こす条件についてより詳しく説明します。

方程式の積分

数学、科学、工学などにおける多くの問題には、関数の定積分の計算が要求されます。関数が $f(x)$ で与えられ、積分区間が $a \sim b$ の場合、積分は次のように表現されます。

$$I = \int_a^b f(x) dx$$



I は、関数 $f(x)$ のグラフ、 x -軸、 $x=a$ と $x=b$ のリミットによって囲まれた領域の面積として解釈できます ($f(x)$ は積分区間において負にならないものとします)。

 操作 ($\int$ FN) は、現在の方程式を指定された変数 ($\int$ FN d_) で積分します。関数には複数の変数があっても問題ありません。

方程式の積分 (∫ FN)

方程式を積分するには：

1. 被積分関数を定義する方程式が方程式リストに格納されていない場合、方程式を入力し（第 6 章「方程式リストへの方程式の入力」を参照）、方程式モードを終了します。方程式には通常、式のみが含まれます。
- ✓ 2. 積分の範囲を入力します。下限を入力し、**[ENTER]** を押して、上限を入力します。
3. 方程式を表示します。**[EQN]** を押し、必要に応じて方程式リストをスクロールし（**[^]** または **[v]** を押します）、目的の方程式を表示します。
4. 積分の変数を選択します。**[G]** **[f]** 変数を押します。これにより計算が開始します。

[f] は、電卓で他の演算よりも多くのメモリを消費します。**[f]** を実行したときに MEMORY FULL メッセージが表示された場合は、付録 B を参照してください。

[C] または **[R/S]** を押すと積分の計算を中止できます。メッセージ“INTERRUPTED”が第 2 行に表示されますが、積分は再開することができません。計算が正常に終了するまで、積分に関する情報は得られません。

表示形式の設定は、関数に対して想定され、結果に使用される確度のレベルに影響します。ALL や、より高い FIX、SCI、ENG 設定の場合、積分の精度は上がりますが、所要時間がはるかに長くなります。結果の**不確定値**が Y レジスタに格納され、積分の範囲は T レジスタと Z レジスタに入ります。詳細については、この章の最後の「積分の確度」を参照してください。

同じ方程式を異なる情報で積分するには：

- ✓ 同じ積分区間を使用する場合、**[R]** **[R]** を押して、積分の範囲を X レジスタと Y レジスタに移動します。上記リストのステップ 3 から開始します。別の区間を使用する場合、ステップ 2 から開始します。

異なる方程式を使用して別の問題を処理する場合、被積分関数を定義する方程式を使ってステップ 1 からやり直します。

8-2 方程式の積分

例：ベッセル関数

次式は、ゼロ次の第 1 種ベッセル関数です。

$$J_0(x) = \frac{1}{\pi} \int_0^{\pi} \cos(x \sin t) dt$$

x- 値が 2 と 3 の場合のベッセル関数を求めます。

被積分関数を定義する式を入力します。

$$\cos(x \sin t)$$

キー：	ディスプレイ：	説明：
CLEAR 3		メモリをクリアします。
(3ALL) (Y) ENTER		
EQN	3*3 lin. solve EQN LIST TOP	方程式モードを選択します。
COS RCL X	COS(X)	方程式を入力します。
X SIN	COS(XxSIN(	
RCL T	COS(XxSIN(T)	
> >	COS(XxSIN(T))_	
ENTER	COS(XxSIN(T))	式を完了し、左端を表示します。
SHOW	CK=E1EC LN=13	チェックサムと長さ。
C		方程式モードを終了します。

この関数を区間 0 ~ π の t、x = 2 で積分します。

キー：	ディスプレイ：	説明：
MODE 2 (2RAD)		ラジアンモードを選択します。
0 ENTER π	3.1416	積分の範囲を入力します (下限値を最初に入力)。
EQN	COS(XxSIN(T))	関数を表示します。
/	∫ FN d_	積分の変数の入力を求めます。

T	X?	X の値の入力を求めます。
2 R/S	値 INTEGRATING ∫ = 0.7034	x = 2. 積分を開始し、結果を 計算します $\int_0^{\pi} f(t)$
✓ ↵ π ÷	0.2239	J ₀ (2) の最終結果

同じ積分区間で J₀(3) を計算します。π による後続の除算によってスタックから積分区間 (0, π) が消えたため、積分区間を再入力する必要があります。

キー:	ディスプレイ:	説明:
✓ 0 ENTER ↵ π	3.1416	積分の範囲を入力します (下限値を最初に入力)。
EQN ↵ f T	COS(X×SIN(T)) ∫ FN d_ X? 2.0000	現在の方程式を表示します。 積分の変数の入力を求めます。 X の値の入力を求めます。
3 R/S	INTEGRATING ∫ = -0.8170	x = 3. 積分を開始し、結果を 計算します $\int_0^{\pi} f(t) \cdot$
✓ ↵ π ÷	-0.2601	最終結果 J ₀ (3).

✓ 例：正弦積分

通信理論における問題（例えば、理想的なネットワークにおけるパルス伝達）では、**正弦積分**と呼ばれる、次のような積分の計算が必要となります。

$$S_i(t) = \int_0^t \left(\frac{\sin x}{x} \right) dx$$

S_i(2) を求めます。

被積分関数の関数を定義する式を入力します。

$$\frac{\sin x}{x}$$

8-4 方程式の積分

電卓が積分の下限値である $x = 0$ で式を評価しようとすると、エラー (DIVIDE BY 0) になります。しかし、積分区間が極端に狭過ぎたり、サンプル点の数が極端に多過ぎない限り、積分のアルゴリズムでは通常、どちらか一方のリミットで式を評価することはありません。

キー:	ディスプレイ:	説明:
[EQN]	3*3 lin. solve EQN LIST TOP	方程式モードを選択します。
[SIN] [RCL] [X]	SIN(X) ₂	方程式を開始します。
[>]	SIN(X) ₂ _	この場合、右の閉じ括弧が必要です。
[÷] [RCL] [X]	SIN(X)÷X ₂	
[ENTER]	SIN(X)÷X	方程式を終了します。
[←] [SHOW]	CK=0EE0 LN=8	チェックサムと長さ。
[C]		方程式モードを終了します。

この関数を x (すなわち X) について区間 $0 \sim 2$ ($t = 2$) で積分します。

キー:	ディスプレイ:	説明:
[MODE] [2] (2RAD)		ラジアンモードを選択します。
[0] [STO] [X] [ENTER]	2 ₂	積分の範囲を入力します (下限値を最初に入力)。
[2]		
[EQN]	SIN(X)÷X	現在の方程式を表示します。
[←] [/] [X]	INTEGRATING ∫ = 1.6054	$S(2)$ の結果を計算します。

積分の確度

電卓は、積分の計算で厳密解を求めることができないため、結果は**近似値**になります。近似値の確度は、方程式によって計算されるとき、被積分関数自体の確度に依存します。これは、電卓の丸め誤差と実験定数の確度に影響されます。

スパイク、非常に急激な発振などの特性を持つ関数の積分は、計算が不正確になる**可能性**がありますが、確率はわずかです。問題の原因となる関数の一般的な特性と、対処するための技術については、付録 E を参照してください。

確度の指定

ディスプレイの表示設定（FIX、SCI、ENG、ALL）により、積分計算の**精度**が決まります。表示される桁数が多いほど、計算された積分の精度は高くなります（計算にかかる時間が長くなります）。表示される桁数が少ないほど、計算速度は速くなりますが、電卓は、関数の精度を指定された桁数までと仮定しています。

積分の**確度**を指定するには、ディスプレイで表示される桁数が**被積分関数の値**で正確とみなされる桁数を**超えない**ように、表示設定を設定します。確度と精度のレベルが、積分の結果に反映されます。

分数表示モードがオンのとき（フラグ 7 がセットされているとき）は、確度が、前の表示形式によって指定されます。

確度の解釈

積分の計算後、電卓は、積分結果の推定した**不確定値**を Y レジスタに格納します。 $\boxed{x \div y}$ を押して、不確定値を表示します。

例えば、積分 $Si(2)$ が 1.6054 ± 0.0002 の場合、 0.0002 が不確定値です。

例：確度を指定

表示形式を SCI 2 に設定した状態で、前回の例の $S_i(2)$ の式で積分を計算します。

キー：	ディスプレイ：	説明：
  	1.61E0	科学表記の 2 桁に設定し、関数を小数点以下 2 桁の確度に指定します。
(2SCI) 		
✓  	0.00E0 2.00E0	積分区間を Z および T レジスタから、X および Y レジスタに移動します。
	SIN(X)÷X	
  X	INTEGRATING ∫ = 1.61E0	現在の方程式を表示します。 積分が、小数点以下 2 桁まで近似されます。
	1.61E-2	

積分値は 1.61 ± 0.0161 です。不確定値は小数点以下 2 桁まで積分の近似値に影響しないので、この近似で表示されるすべての桁が正確であるとみなせます。

もし近似値の不確定値が許容範囲を超える場合、表示形式の桁数を増やして積分を反復できます ($f(x)$ が、ディスプレイの表示桁数まで正確に計算されると仮定します)。一般的に、表示桁を 1 つ増やすと、積分の不確定値が 1 桁小さくなります。

例：確度を変更

直前に計算した $S_i(2)$ の積分の場合、結果が小数点以下 2 桁でなく 4 桁まで正確になるよう指定します。

キー：

ディスプレイ：

説明：

 **DISPLAY** **2**
(2SCI) **4**

1.6079E-2

確度を小数点以下 4 桁に指定します。最後の例の不確定値がディスプレイに残っています。

✓  

0.0000E0
2.0000E0

積分区間を Z および T レジスタから、X および Y レジスタに移動します。

EQN

SIN(X)÷X

現在の方程式を表示します。

  X

INTEGRATING
∫ =

結果を計算します。



1.6054E0
1.6056E-4

前に計算された SCI 2 結果の不確定値と比べて、不確定値が約 1/100 になりました。

DISPLAY **1** (2SCI) **4**

0.0002

表示形式を FIX 4 に戻します。

MODE **1** (1DEG)

0.0002

度モードに戻します。

この不確定値は、結果が小数点以下 3 桁までしか正確とは言えない**可能性がある**ことを示します。実際は、結果をこの積分の実際の値と比較した場合、結果は 7 桁まで正確です。結果の不確定値は控えめに計算されているため、ほとんどの場合、電卓の近似値は、不確定値が示すものより正確です。

詳細

この章では、広範囲のアプリケーションで HP 35s の積分を使用するための手順について説明しました。付録 E で、積分のアルゴリズム、正しくない結果を引き起こす条件や計算時間が長くなる条件、積分に対する現在の近似値の取得について、より詳しく説明します。

8-8 方程式の積分

複素数

HP 35s では次の形式の複素数を使用できます。

$$x+iy \quad x+y \cdot i \quad r \angle \theta$$

HP 35s は、複素四則演算 (+、-、×、÷)、複素三角関数 (sin、cos、tan)、および数学関数 $-z$ 、 $1/z$ 、 $z_1^{z_2}$ 、 $\ln z$ 、 e^z を処理します。(ここで z_1 と z_2 は複素数)。

$x+yi$ 形式は、ALG モードでのみ利用できます。

複素数を入力するには：

形式： $x+iy$

1. 実数部を入力します。
2. **[i]** を押します。
3. 虚数部を入力します。

形式： $x+y \cdot i$

1. 実数部を入力します。
2. **[+]** を押します。
3. 虚数部を入力します。
4. **[i]** を押します。

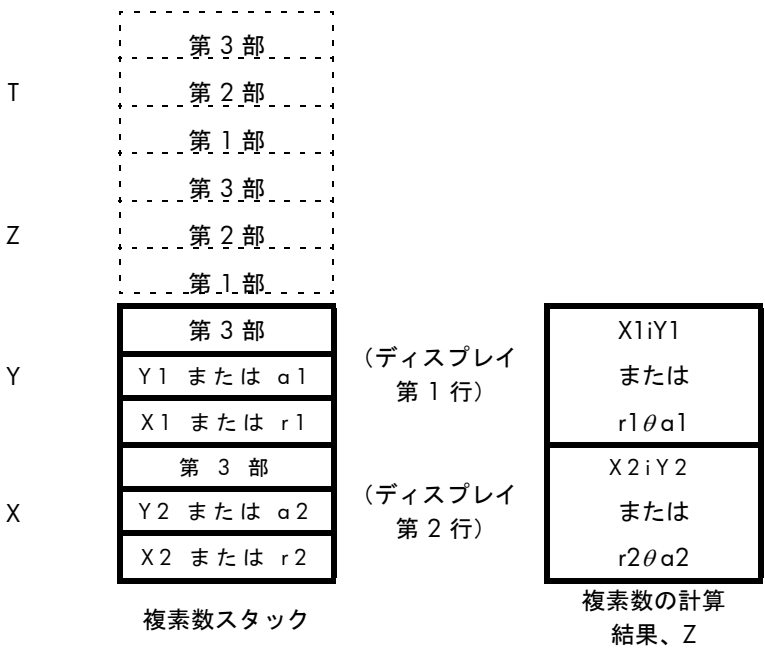
形式： $r \angle \theta$

1. r の値を入力します。
2. **[R]** **[θ]** を押します。
3. θ の値を入力します。

この章の例では、他に記載がない限り、すべて RPN モードを使用します。

✓ 複素数スタック

複素数は、スタックレベルの 第 1 部と第 2 部に格納されます。RPN モードでは、X レジスタの 第 1 部と第 2 部に格納された複素数がディスプレイ第 2 行に、Y レジスタの 第 1 部と第 2 部に格納された複素数がディスプレイ第 1 行に表示されます。



複素数の計算

複素数の計算は、ALG モードと RPN モードで実数の計算と同様に使用できます。

✓ 1 個の複素数を持つ演算を実行するには：

1. 前項の説明に従って複素数 z を入力します。
2. 複素関数を選択します。

1 個の複素数 z に対する関数

計算内容：	以下のキーを押します。
符号の変更、 $-z$	
逆数、 $1/z$	
自然対数、 $\ln z$	 LN
逆自然対数、 e^z	 e^x
$\sin z$	 SIN
$\cos z$	 COS
$\tan z$	 TAN
絶対値、ABS(z)	 ABS
引数値、ARG(z)	 ARG

✓ 2 個の複素数を持つ四則演算を実行するには：

1. 前項の説明に従って最初の複素数 z_1 を入力します。
2. 前項の説明に従って第 2 の複素数 z_2 を入力します。
3. 四則演算を選択します。

2 個の複素数 z_1 と z_2 を持つ四則演算

計算内容：	以下のキーを押します。
加算、 $z_1 + z_2$	
減算、 $z_1 - z_2$	
乗算、 $z_1 \times z_2$	
除算、 $z_1 \div z_2$	
累乗 $z_1^{z_2}$	 y^x

例:

複素数を持つ三角関数と四則演算の例をいくつか挙げます。

$\sin(2i3)$ を評価します

キー:	ディスプレイ:	説明:
 DISPLAY  ($9 \times i$)		表示形式を設定します。
   SIN	$9.1545i-4.1689$	結果は $9.1545i-4.1689$ です。

式を評価します

$$z_1 \div (z_2 + z_3),$$

ただし、 $z_1 = 23i13$ 、 $z_2 = -2i1$ 、 $z_3 = 4i-3$

次のように計算を実行します。

キー:	ディスプレイ:	説明:
 DISPLAY  ($9 \times i$)		表示形式を設定します。
     ENTER	$23.0000i13.0000$	z_1 を入力します
	$23.0000i13.0000$	
    ENTER	$-2.0000i1.0000$	z_2 を入力します
	$-2.0000i1.0000$	
    	$23.0000i13.0000$	$(z_2 + z_3)$ 。結果は $2i-2$ です。
	$2.0000i-2.0000$	
	$2.5000i9.0000$	$z_1 \div (z_2 + z_3)$ 。結果は $2.5i9$ です。

$(4i-2/5) \nabla (3i-2/3)$ を評価します。

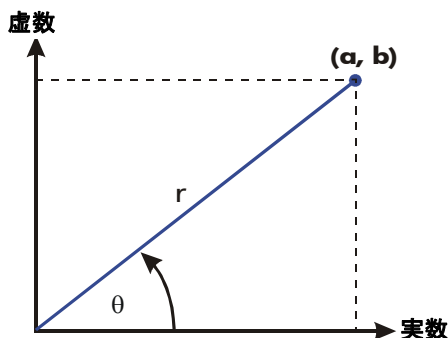
キー:	ディスプレイ:	説明:
$\leftarrow$ [DISPLAY] [9] ($9 \times i \cdot y$)		表示形式を設定します。
[4] [i] [.] [2] [.] [5] [=]	$4.0000i-0.4000$	$4i/5$ を入力します
[ENTER]	$4.0000i-0.4000$	
[3] [i] [.] [2] [.] [3] [=]	$4.0000i-0.4000$	$3i/2/3$ を入力します
	$3i-0.2/3$	
[x]	$11.7333i-3.8667$	結果は $11.7333i-3.8667$ です

e^{z-2} を評価します。 $z = (1i \ 1)$

キー:	ディスプレイ:	説明:
[1] [i] [1] [ENTER]	$1.0000i-1.0000$	$1i1$ を入力します。中間結果。
[2] [=] [y^x]	$1.0000i-1.0000$	
	$0.0000i-5.0000$	$Z-2$ 。結果は $0i-5$ です
[e^x]	$0.8776i-0.4794$	最終結果は $0.8776i-0.4794$ 。

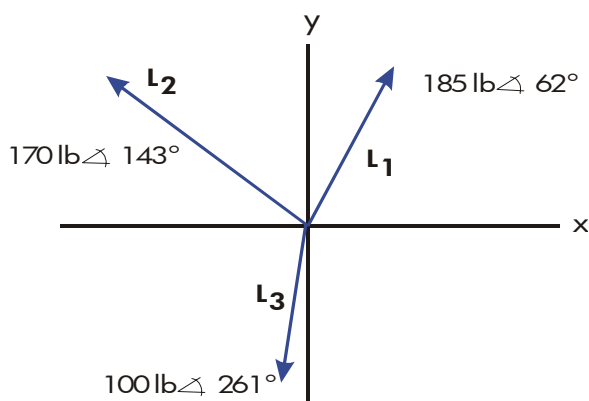
複素数と極座標

多くのアプリケーションでは、**極座標形式**または**極座標表記**の実数を使用します。これらの形式では複素数の場合と同様に 1 対の数値が使用されるため、これらの数値を複素数演算を使って計算できます。



例：ベクトル加算

次の3つの荷重を加算します。



キー：

ディスプレイ：

説明：

MODE **1** (1DEG)

DISP **0**

(10r8a)

1 **8** **5** **↵** **0**

6 **2** **ENTER**

1 **7** **0** **↵** **0**

1 **4** **3** **ENTER**

1 **0** **0** **↵** **0**

2 **6** **1** **+**

+

↵ **>** **FN=**

185.0000062.0000

185.0000062.0000

170.00000143.00...

170.00000143.000➡

185.0000062.0000

151.45290178.660➡

178.93720111.148➡

←9

度モードを設定します。

複素数モードを設定します。

L_1 を入力します

L_2 を入力します

L_3 を入力し、 $L_2 + L_3$ を計算します

$L_1 + L_2 + L_3$ を計算します

画面をスクロールして答えの残りを表示します

複素数の計算には異なる形式の複素数を使用できますが、結果の表示形式は

DISP メニューの設定に依存します。

$1i1+3\theta10+5\theta30$ を評価します

キー:	ディスプレイ:	説明:
MODE 1 (1DEG)		度モードを設定します。
← DISPLAY □ 0		複素数モードを設定します。
(10rθα)		
1 i 1 ENTER	1.4142045.0000	1i1 を入力します
	1.4142045.0000	
3 → θ 1 0	3.0000010.0000	3θ10 を入力します
ENTER	3.0000010.0000	
5 → θ 3 0	1.4142045.0000	5θ30 を入力し、3θ10
+	7.8861022.5241	を加算します
+	9.2088025.8898	1i1 を加算します。結果は 9.2088025.8898 です

方程式内の複素数

複素数を方程式に入力できます。方程式が表示されるときは、 xiy 、 $r\theta\alpha$ など、すべての数値が入力時の形式で表示されます。

方程式を評価する際に変数の値の入力を求められたときに、複素数を入力できます。結果の値と形式は、ディスプレイの設定によって制御されます。これは、ALG モードでの計算でも同じです。

複素数を含む方程式でも、SOLVE と積分が実行できます。

プログラムにおける複素数

プログラムで、複素数を入力できます。例えば、 $1i2+3\theta 10+5$

$\theta 30$ は、プログラムで次のようになります。

プログラム行 : (ALG モード)	説明
F001 LBL F	プログラムを開始します
F002 $1i2+3\theta 10+5\theta 30$	
F003 RTN	

プログラムを実行中、INPUT 命令によって値の入力を求められたときに、複素数を入力できます。結果の値と形式は、ディスプレイの設定によって制御されます。

複素数を含むプログラムでも、SOLVE と積分が実行できます。

ベクトルの演算

数学的な観点からは、ベクトルは複数の要素を持つ、1 行あるいは 1 列の配列です。

物理でのベクトルは、2 つまたは 3 つの成分から構成され、位置、速度、加速度、力、モーメント、線運動量、角運動量、角速度、角加速度などの物理量を表わします。

ベクトルを入力するには：

1.   を押します。
2. ベクトルの最初の数値を入力します。
3.   を押し、2 次元ベクトルまたは 3 次元ベクトルの第 2 の数値を入力します。
4.   を押し、3 次元ベクトルの第 3 の数値を入力します。

HP 35s では 3 次元を超えるベクトルは処理できません。

ベクトルの演算

✓ 加算と減算：

ベクトルの加算と減算では、2 つのベクトルオペランドが同じ長さである必要があります。異なる長さのベクトルを加算または減算しようとする、エラーメッセージ “INVALID DATA” が表示されます。

1. 第 1 のベクトルを入力します。
2. 第 2 のベクトルを入力します。
3.  または  を押します。

✓ $[1.5, -2.2] + [-1.5, 2.2]$ を計算します

キー:	ディスプレイ:	説明:
MODE 5 (5RPN)		RPN モードに変更します (必要な場合)
→ 1 • 5 ↵	$[1.5000, -2.2000]$	$[1.5, -2.2]$ を入力します
↵ +/- 2 • 2	$[1.5000, -2.2000]$	
ENTER		
→ +/- 1 • 5	$[1.5000, -2.2000]$	$[-1.5, 2.2]$ を入力します
↵ ↵ 2 • 2	$[-1.5, 2.2]$	
+	0.0000	2 つのベクトルを加算します
	$[0.0000, 0.0000]$	

$[-3.4, 4.5] - [2.3, 1.4]$ を計算します

キー:	ディスプレイ:	説明:
MODE 4 (4ALG)		ALG モードに変更します
→ +/- 3 • 4	$[-3.4, 4.5]$ _	$[-3.4, 4.5]$ を入力します
↵ ↵ 4 • 5 >		
- → 2 • 3	$\leftarrow [-3.4, 4.5] - [2.3, 1.4]$	$[2.3, 1.4]$ を入力します
↵ ↵ 1 • 4		
ENTER	$[-3.4, 4.5] - [2.3, 1.4]$ $[-5.7000, 3.1000]$	2 つのベクトルを減算します

✓ スカラーによる乗算と除算:

1. ベクトルを入力します
2. スカラーを入力します
3. 乗算の場合 **×** を、除算の場合 **÷** を押します

✓ [3,4]x5 を計算します

キー:	ディスプレイ:	説明:
[MODE] [5] (5RPN)		RPN モードに変更します
[⇨] [[]] [3] [⇩] [,] [4]	[3.0000,4.0000]	[3,4] を入力します
[ENTER]	[3.0000,4.0000]	
[5]	[3.0000,4.0000]	5 をスカラーとして入力
	5_	します
[x]	0.0000	乗算を実行します
	[15.0000,20.0000]	

[2,4]÷2 を計算します

キー:	ディスプレイ:	説明:
[MODE] [4] (4ALG)		ALG モードに変更します
[⇨] [[]] [+/-] [2] [⇩]	[-2,4]_	[2,4] を入力します
[,] [4] [>]		
[÷] [2]	[-2,4]÷2	5 をスカラーとして入力
		します
[ENTER]	[-2,4]÷2	除算を実行します
	[-1.0000,2.0000]	

ベクトルの絶対値

絶対値の関数“ABS”をベクトルに適用すると、そのベクトルの大きさが得られます。ベクトル $A=(A_1, A_2, \dots, A_n)$ の場合、大きさは

$$|A| = \sqrt{A_1^2 + A_2^2 + \dots + A_n^2} \text{ と定義されます。}$$

1. **[⇨]** **[ABS]** を押します。
2. ベクトルを入力します。
3. **[ENTER]** を押します。

例：ベクトル [5,12] の絶対値：

[⇨] **[ABS]** **[⇨]** **[[]]** **[5]** **[⇩]** **[,]** **[1]** **[2]** **[ENTER]**。結果は 13 です。RPN モードでは **[MODE]** **[5]** (5RPN) **[⇨]** **[[]]** **[5]** **[⇩]** **[,]** **[1]** **[2]** **[⇨]** **[ABS]** となります。

ドット積

関数 DOT は、2 つの同じ長さのベクトルの、ドット積の計算に使用します。異なる長さの 2 つのベクトルのドット積を計算しようとすると、エラーメッセージ "INVALID DATA" が表示されます。

2 次元ベクトル: $[A, B]$ 、 $[C, D]$ 、ドット積は $[A, B] \cdot [C, D] = A \times C + B \times D$ として定義されます。

3 次元ベクトル: $[A, B, X]$ 、 $[C, D, Y]$ 、ドット積は $[A, B, X] \cdot [C, D, Y] = A \times C + B \times D + X \times Y$ として定義されます。

1. 第 1 のベクトルを入力します。
2.  を押します。
3. 第 2 のベクトルを入力します。
4.  を押します。

注: 記号  は、ここでは「クロス積」ではなく「ドット積」を意味しています。クロス積については、第 17 章を参照してください。

ベクトル $[1, 2]$ と $[3, 4]$ のドット積を計算します

キー:	ディスプレイ:	説明:
  (4ALG)		ALG モードに変更します
     	$[1, 2]$	第 1 のベクトル $[1, 2]$ を入力します。
		
     	$[1, 2] \times [3, 4]$	ドット積の  を実行し、
		第 2 のベクトルを入力します
	11.0000	2 つのベクトルのドット積は 11 です

✓ ベクトル $[9, 5]$ と $[2, 2]$ のドット積を計算します

キー:	ディスプレイ:	説明:
  (5RPN)		RPN モードに変更します
     	$[9.0000, 5.0000]$	第 1 のベクトル $[9, 5]$ を入力します
	$[9.0000, 5.0000]$	
     	$[9.0000, 5.0000]$ $[2, 2]$	第 2 のベクトル $[2, 2]$ を入力します

10-4 ベクトルの演算

[X]

28.0000

ドット積の **[X]** を押します。
2 つのベクトルのドット積
は 28 です

ベクトル間の角度

2 つのベクトル A と B の角度を求める式は $\theta = \text{ACOS}(A \cdot B / |A||B|)$

次の 2 つのベクトルの角度を求めます。A=[1,0]、B=[0,1]

キー：

ディスプレイ：

説明：

[MODE] **[4]** (4ALG)

ALG モードに変更します

[MODE] **[1]** (1DEG)

度モードを設定します

[F1] **[ACOS]**

ACOS(<)

アークコサイン関数

[F1] **[1]** **[<]** **[,]** **[0]**

ACOS([1,0])

ベクトル A [1,0] を入力します

[>]

ます

[X] **[F1]** **[1]** **[0]** **[<]** **[,]**

ACOS([1,0]×[0,1])

A と B のドット積のベク

[1] **[>]**

トル B [0,1] を入力します

[÷] **[F1]** **[ABS]** **[F1]** **[1]**

←,1]÷ABS([1,0])→

ベクトル A [1,0] の大きさ

[1] **[<]** **[,]** **[0]** **[>]**

を計算します

[÷] **[F1]** **[ABS]** **[F1]** **[1]**

←1,0]÷ABS([0,1])→

ベクトル B [0,1] の大きさ

[0] **[<]** **[,]** **[1]**

を計算します

[ENTER]

ACOS([1,0]×[0,1])

2 つのベクトルの角度は

90.0000

90 度です

✓ 次の 2 つのベクトルの角度を求めます。A=[3,4]、B=[0,5]

キー：

ディスプレイ：

説明：

[MODE] **[5]** (5RPN)

RPN モードに変更します

[MODE] **[1]** (1DEG)

度モードを設定します

[F1] **[3]** **[<]** **[,]** **[4]**

90

2 つのベクトルのドット

[ENTER] **[F1]** **[1]** **[0]** **[<]**

20.0000

積を計算します

[,] **[5]** **[X]**

[F1] **[3]** **[<]** **[,]** **[4]**

20.0000

ベクトル [3,4] の大きさを

[F1] **[ABS]**

5.0000

計算します

  0  5	5.0000
 ABS	5.0000
	20.0000
	25.0000
	90
	0.8000
 ACOS	90
	36.8699

ベクトル [0,5] の大きさを
計算します
2 つのベクトルを乗算し
ます
2 つの値を除算します
2 つのベクトルの角度は
36.8699 度です

方程式内のベクトル

ベクトルは、方程式や方程式の変数内で、実数とまったく同様に使用できます。変数の入力を求められたときに、ベクトルを入力できます。

ベクトルを含む方程式を解くことができますが、未知数がベクトルの場合、解法の機能が制限されます。

ベクトルを含む方程式を積分することもできますが、方程式の結果は、実数または 1 次元ベクトルか、第 2 要素と第 3 要素がゼロのベクトルである必要があります。

プログラム内のベクトル

ベクトルは、プログラム内で、実数や複素数と同様に使用できます。

例えば、 $[5, 6] + 2 \times [7, 8] \times [9, 10]$ をプログラムで入力してみます。

プログラム行：
G0001 LBL G
G0002 [5,6] + 2 × [7,8] × [9,10]
G0003 RTN

説明：
プログラムを開始します
[5,6]

変数の値の入力を求められたときに、ベクトルを入力できます。ベクトルを含むプログラムを SOLVE と積分に使用できます。

10-6 ベクトルの演算

変数またはレジスタからのベクトルの作成

実行モードまたはプログラムモードで、メモリ変数の内容、スタックレジスタ、間接レジスタの値を含むベクトルを作成できます。

ALG モードでは、**[F2][I]** を押してベクトルの入力を開始します。RPN モードでも同じキーを使いますが、**[EQN]** キーを押してから **[F2][I]** を押す必要があります。

文字付き変数に格納された値を含む要素を入力するには、**[RCL]** と変数文字を押します。

スタックレジスタの要素を入力するには、**[R↓]** キーを押し、**[>]** または **[<]** キーを使用してアンダーラインを目的のスタックレジスタまで移動して、**[ENTER]** を押します。

I レジスタ や J レジスタ内の値によって間接的に指示された要素を入力するには、**[RCL]** の後に (I) または (J) を押します。

例えば、RPN モードでベクトル [C, REGZ, (J)] を作成するには、**[EQN]** **[F2]** **[I]**、次に **[RCL]** **[C]** **[↵]** **[,]** **[R↓]** **[>]** **[ENTER]** **[↵]** **[,]** **[RCL]** **[(J)]** **[ENTER]** を押します。

基数変換と四則および論理

BASE メニュー（ **BASE**）では、10 進数、2 進数、8 進数、16 進数の入力と変換が行えます。

LOGIC メニュー（ **LOGIC**）では、論理関数が利用できます。

BASE メニュー

メニューラベル	説明
DEC	10 進数モード。これは、通常の計算モードです。
HEX	16 進数モード。このモードがアクティブのとき、 HEX インジケータ C^{TM} 表示されます。数値が 16 進数で表示されます。RPN モードでは、キー SIN 、 COS 、 TAN 、 $\sqrt{x}$ 、 y^x 、 $1/x$ が、桁 A ~ F を入力するためのショートカットとして機能します。ALG モードでは、 RCL A、B、C、D、E、F を押して、桁 A ~ F を入力します。
OCT	8 進数モード。このモードがアクティブのとき、 OCT インジケータが表示されます。数値が 8 進数で表示されます。
BIN	2 進数モード。このモードがアクティブのとき、 BIN インジケータが表示されます。数値が 2 進数で表示されます。数値が 12 桁を超える場合、  > キーと  < キーにより全部の数値を表示できます（この章で後述の「長い 2 進数を表示させる」を参照してください）。
d	数値の最後にあれば、数値が 10 進数であることを示しています。
h	数値の最後にあれば、数値が 16 進数であることを示しています。16 進数を入力するには、数値の後に “h” を追加します。

○	数値の最後にあれば、数値が 8 進数であることを示しています。8 進数を入力するには、数値の後に “○” を追加します。
ⓑ	数値の最後にあれば、数値が 2 進数であることを示しています。2 進数を入力するには、数値の後に “ⓑ” を追加します。

例：数値の基数変換

基数変換では、以下のキー操作を実行します。

125.99₁₀ を 16 進数、8 進数、2 進数に変換します。

キー：	ディスプレイ：	説明：
1 2 5  BASE	7Dh	10 進数を 16 進数に変換します。
2 (2HEX)	175○	8 進数
 BASE 3 (3OCT)	1111101ⓑ	2 進数
 BASE 4 (4BIN)	125.0000	
 BASE 1 (1DEC)		

注：10 進数以外ときは、数値の整数部のみが表示されます。分数部はそのまま（分数を削除する操作をするまで）保持され、10 進数を選択した場合に表示されます。

24FF₁₆ を 2 進数に変換します。2 進数は、14 桁（表示可能な最大桁）以上になります。

キー：	ディスプレイ：	説明：
✓  BASE 2 (2HEX)	24FFh	 1/x キーを使用して、“F” を入力します。
2 4 1/x 1/x 		
BASE 6 (6h)		

BASE **4** (4BIN)

1001001111111111 ➡

2 進数は画面に納まりません。

➡ インジケータにより右側に
数値が続いていることがわか
ります。

>

←b

残りの数値を表示します。

数値全体は

1001001111111111_b です。

<

1001001111111111 ➡

最初の 14 桁を再表示します。

BASE **1** (1DEC)

9,471.0000

10 進数に戻します。

基数モードで 2/8/10/16 進数を表わすには、**BASE** メニューを使用して、オペランドの後に基数記号 b/o/d/h を入力します。基数記号がない数値は 10 進数です。

注：

ALG モード：

1. 計算結果の基数モードは、現在の基数モード設定によって決まります。
2. アクティブなコマンド行がない場合（点滅するカーソルがディスプレイ第 1 行にない場合）、基数を変更すると、第 2 行が新しい基数に更新されます。
3. **ENTER** を押すか、基数モードを変更すると、電卓は、第 2 行の基数 2/8/16 を表わすため、現在の基数記号 b/o/h を結果に自動的に追加します。
4. 式を再編集するには、**<** または **>** を押します。

RPN モードでは：

ディスプレイ第 2 行に数値を入力し、**ENTER** を押してから、基数モードを変更すると、電卓は、第 1 行と第 2 行の数値の基数を変換します。基数 2/8/16 を表わすため数値の後に記号 b/o/h が追加されます。

次の画面の内容を第 2 行に表示するには、 **<** または **>** を押して画面を変更します。

LOGIC メニュー

メニューラベル	説明
AND	2 つの引数のビット単位の論理積 "AND"。 例 : $\text{AND}(1100\text{b}, 1010\text{b}) = 1000\text{b}$
XOR	2 つの引数のビット単位の排他的論理和 "XOR"。 例 : $\text{XOR}(1101\text{b}, 1011\text{b}) = 110\text{b}$
OR	2 つの引数のビット単位の論理和 "OR"。 例 : $\text{OR}(1100\text{b}, 1010\text{b}) = 1110\text{b}$
NOT	引数の補数を返します。結果の各ビットは、引数内の対応するビットの補数です。 例 : $\text{NOT}(1011\text{b}) =$ 1111111111111111111111111111111110100b
NAND	2 つの引数のビット単位の否定論理積 "NAND"。 例 : $\text{NAND}(1100\text{b}, 1010\text{b}) = 11111111111111111111111111111111110111\text{b}$
NOR	2 つの引数のビット単位の否定論理和 "NOR"。 例 : $\text{NOR}(1100\text{b}, 1010\text{b}) =$ 1111111111111111111111111111111110001b

"AND"、"OR"、"XOR"、"NOT"、"NAND"、"NOR" は論理関数としても利用できます。分数、複素数、ベクトル引数は、論理関数に "INVALID DATA" として表示されます。

基数 2、8、16 の四則演算

任意の基数で $+$ 、 $-$ 、 $\times$ 、 $\div$ を使用して、四則演算を実行できます。HEX モードで非アクティブの関数キーは、 $\sqrt{x}$ 、 e^x 、 $\ln$ 、 y^x 、 $1/x$ 、 $\Sigma+$ だけです。ただし四則演算以外のほとんどの演算は、数値の小数部が切り捨てられるため、意味のある結果を生成しません。

2 進数、8 進数、16 進数での四則演算は 2 の補数形式で、整数のみが使用されます。

- 数値に小数部が含まれる場合、四則計算には整数部のみが使用されます。
- 演算の結果は常に整数になります（小数部は切り捨てられます）。

11-4 基数変換と四則および論理

基数変換は数値の表示のみを変更し、X レジスタ内の実際の数値を変更しないのに対し、**四則演算**は、X レジスタの数値そのものを入れ替えます。

演算結果を有効なビットで表現できないときは、ディスプレイに OVERFLOW と表示され、正または負の表示可能な最大値が表示されます。

例：

16 進数、8 進数、2 進数モードでの四則演算の例をいくつか挙げます。

$$12F_{16} + E9A_{16} = ?$$

キー：

BASE **2** (2HEX)

ディスプレイ：

FC9h

説明：

基数 16 を設定します。**HEX** インジケータがオンになります。結果。

✓ **1** **2** **/x** **BASE** **6**
(6h) **ENTER** **y^x** **9** **SIN**
BASE **6** (6h) **+**

$$7760_8 - 4326_8 = ?$$

BASE **3** (3OCT)

7711o

基数 8 を設定します。**OCT** インジケータがオンになります。表示数値を 8 進数に変換します。結果。

✓ **7** **7** **6** **0** **BASE**
7 (7o) **ENTER** **4** **3** **2**
6 **BASE** **7** (7o) **-**

$$100_8 \div 5_8 = ?$$

✓ **1** **0** **0** **BASE** **7**
(7o) **ENTER** **5** **BASE**
7 (7o) **÷**

14o 結果の整数部。

$$5A0_{16} + 1001100_2 = ?$$

✓ **BASE** **2** (2HEX) **5**
SIN **0** **BASE** **6**
(6h) **ENTER**

5A0h

基数 16 を設定します。**HEX** インジケータがオンになります。

BASE **4** (4BIN)
1 **0** **0** **1** **1** **0** **0**
 8 (8b)

1001100b

基数 2 に変更します。**BIN**
インジケータがオンになります。この操作で数値の入力が完了するため、数値の間に **ENTER** を押す必要はありません。
2 進数での結果。
16 進数での結果。
10 進数に戻します。

✓ **+**
 BASE **2** (2HEX)
 BASE **1** (1DEC)

10111101100b
5ECh
1,516.0000

数値の表現

基数変換のときには数値の表示が変化しますが、格納の形式は変化しないため、10 進数は、四則演算で利用されるまで切り捨てられません。

数値が 16 進数、8 進数、2 進数で表示されているときは、36 ビット（8 進数で 12 桁、16 進数で 9 桁）で表示されます。先頭のゼロは表示されませんが、正の数値を示しているため重要です。例えば、125¹⁰の 2 進数表現は、次のように表示されます。

1111101b

これは、次の 36 桁表示と同じです。

000000000000000000000000000000001111101b

負の数値

数値の 2 進数表現のいちばん左の（最上位または最高）ビットは符号ビットで、負の数値の場合は (1) に設定されています。先頭の（省略された）ゼロがある場合、符号ビットはゼロ（正）です。負の数値は、正の 2 進数の 2 の補数です。

キー：

ディスプレイ：

説明：

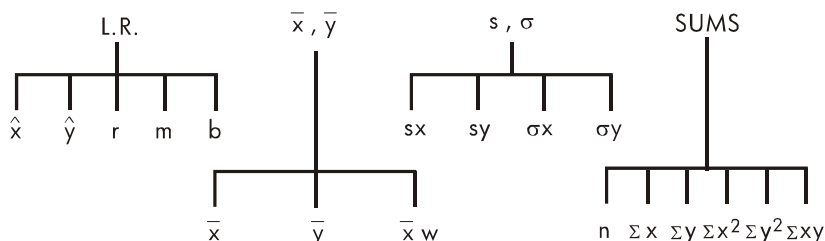
5 **4** **6** **BASE**
2 (2HEX)

222h 正の 10 進数を入力し、
16 進数に変換します。

統計演算

HP 35s の統計メニューには、次に示す 1 変数または 2 変数データ（実数）の統計分析機能があります。

- 平均、標本標準偏差と母標準偏差
- 線形回帰と推定 ($\hat{x}$ と $\hat{y}$)
- 加重平均 (x に対する y の加重)
- 総和統計: n , Σx , Σy , Σx^2 , Σy^2 , Σxy



統計データの入力

1 変数および 2 変数の統計データを、 $\Sigma+$ （または $\mathbf{\Delta} \Sigma-$ ）キーを使用してそれぞれ同様の方法で入力（または削除）できます。データ値は、6 個の統計レジスタ（- 27 から - 32）に総和統計として蓄積されます。名前は SUMS メニューに表示されます。（ $\mathbf{\Delta} \text{SUMS}$ ）を押し、 $n \Sigma x \Sigma y \Sigma x^2 \Sigma y^2 \Sigma xy$ を表示します）。

注



新しい統計データを入力する前に、統計レジスタをクリアしてください（ $\mathbf{\Delta} \text{CLEAR}$ $\mathbf{4}$ (4Σ) を押します）。

1 変数データの入力

1.   (4Σ) を押して、既存の統計データをクリアします。
2. それぞれの x 値を入力し、 を押します。
3. ディスプレイには蓄積された統計データの数 n が表示されます。

 を押すと、Y レジスタにすでに格納された値が y 値として蓄積されるため、実際には 2 つの変数が統計レジスタに入力されます。このため、 x データのみを入力した場合や、入力した x 値と y 値の数が異なる場合でも、電卓は線形回帰を実行し、 y に基づいて値を表示します。エラーは発生しませんが、結果は明らかに意味がありません。

値の入力が完了した直後に、値をディスプレイに呼び出すには、 を押します。

✓ 2 変数データの入力

データが変数ペアのときは、最初に従属変数（ペアの第 2 変数）を入力して  を押し、独立変数（ペアの第 1 変数）を入力して  を押します。

1.   (4Σ) を押して、既存の統計データをクリアします。
2. 最初に y の値を入力し、 を押します。
3. 対応する x の値を入力し、 を押します。
4. ディスプレイには蓄積した統計データペアの数 n が表示されます。
5. x, y ペアの入力を続けます。入力のたびに n が更新されます。

x 値の入力が完了した直後に、その値をディスプレイに呼び出すには、 を押します。

データ入力のエラーの修正

統計データの入力で誤りがあったときは、間違っただデータを削除して正しいデータを追加します。 x, y ペアのどちらか一方の値のみを修正する場合でも、**両方**の値を削除して再入力する必要があります。

統計データを修正するには：

1. 間違っているデータを再入力します。 $\Sigma+$ を押す代わりに、 $\leftarrow$ $\Sigma-$ を押します。これにより、値が削除され、 n が減少します。
2. $\Sigma+$ を使用して正しい値を入力します。

✓ 入力直後の値が間違っていたときは、 $\leftarrow$ **LASTx** を押して値を検索し、 $\leftarrow$ $\Sigma-$ を押して削除します。(間違っただ y 値はまだ Y レジスタに残っており、 x 値は **LAST X** レジスタに格納されました)。間違っただ統計データを削除した後、電卓は、 Y レジスタの値を第 1 行に、 n を第 2 行に表示します。

例：

次の表の左側の x, y 値を入力し、右側のように訂正します。

x, y の初期値	x, y の訂正值
20, 4	20, 5
400, 6	40, 6

キー：

$\leftarrow$ **CLEAR** **4** (4Σ)

ディスプレイ：

4.0000
1.0000
6.0000
2.0000
6.0000
400.0000

説明：

既存の統計データをクリアします。
最初の新しいデータペアを入力します。
入力したデータペアの数 n が表示されます。
Brings 最後の x 値を呼び出します。最後の y は、 Y レジスタに残っています。

✓ **4** **ENTER** **2** **0** $\Sigma+$

✓ **6** **ENTER** **4** **0** **0**

$\Sigma+$

$\leftarrow$ **LASTx**

6.0000

1.0000

最後のデータペアを削除します。

最後のデータペアを再入力します。

最初のデータペアを削除します。

最初のデータペアを再入力します。統計レジスタには合計 2 組のデータペアがあります。

✓     

6.0000

2.0000

✓     

4.0000

1.0000



✓     

5.0000

2.0000

統計の計算

データの入力が終わったら、統計メニューの機能を使用できます。

統計メニュー

メニュー	キー	説明
L.R.	 	線形回帰メニュー：線形推定 $\hat{x}$ $\hat{y}$ とカーブフィッティング r m b 。この章の「線形回帰」を参照してください。
$\bar{x}$, $\bar{y}$	 	平均メニュー： $\bar{x}$ $\bar{y}$ $\bar{x} \times 2$ $\bar{y} \times 2$ 。下の「平均」を参照してください。
s , σ	 	標準偏差メニュー： $s \times$ $s \div$ $\sigma \times$ $\sigma \div$ 。この章で後述の「標本標準偏差」と「母標準偏差」を参照してください。
SUMS	 	総和メニュー： Σx Σy Σx^2 Σy^2 Σxy 。この章で後述の「総和統計」を参照してください。

平均

平均とは、あるグループの数値の総計をデータの個数で除算したものです。

- x 値の平均を求めるには、 $\boxed{\text{ON}} \boxed{\overline{x}, y} (\overline{x})$ を押します。
- y 値の平均を求めるには、 $\boxed{\text{ON}} \boxed{\overline{x}, y} \boxed{>} (\overline{y})$ を押します。
- y 値を加重あるいは頻度として使用して x 値の加重平均を求めるには、 $\boxed{\text{ON}} \boxed{\overline{x}, y} \boxed{>} \boxed{>} (\overline{x}w)$ を押します。加重には整数も小数も利用できます。

例：平均（1 変数）

製品の製造で特定の処理にかかる時間の平均を求めてみます。6 人をランダムにピックアップして、各人が処理に費やした時間を計測し、必要な時間（単位分）を記録します。

15.5	9.25	10.0
12.5	12.0	8.5

時間の平均を計算します。（すべてのデータを x 値として処理します）。

キー：	ディスプレイ：	説明：
$\boxed{\text{ON}} \boxed{\text{CLEAR}} \boxed{4} (4\Sigma)$		統計レジスタをクリアします。
$\boxed{1} \boxed{5} \boxed{\cdot} \boxed{5} \boxed{\Sigma+}$	1.0000	最初の時間を入力します。
$\boxed{9} \boxed{\cdot} \boxed{2} \boxed{5} \boxed{\Sigma+} \boxed{1} \boxed{0}$		残りのデータを入力します。
$\boxed{\Sigma+} \boxed{1} \boxed{2} \boxed{\cdot} \boxed{5} \boxed{\Sigma+} \boxed{1}$	6.0000	6 個のデータ点が蓄積されます。
$\boxed{2} \boxed{\Sigma+} \boxed{8} \boxed{\cdot} \boxed{5} \boxed{\Sigma+}$		
$\boxed{\text{ON}} \boxed{\overline{x}, y} (\overline{x})$	$\overline{x} \overline{y} \overline{x}w$	処理を完了するための平均時間を計算します。
	11.2917	

例：加重平均（2 変数）

ある製造業者は、特定の部品を年 4 回購入します。昨年の購入は次の通りでした。

部品単価 (x)	\$4.25	\$4.60	\$4.70	\$4.10
部品の数 (y)	250	800	900	1000

この部品の（購入量に対して加重された）平均価格を求めます。価格 x の前に加重（頻度） y を入力します。

キー：

ディスプレイ：

説明：

CLEAR **4** (4Σ)

統計レジスタをクリアします。

✓ **2** **5** **0** **ENTER** **4** **.**

データを入力します。

2 **5** **$\Sigma+$**

n が表示されます。

✓ **8** **0** **0** **ENTER** **4** **.**

6 **$\Sigma+$**

✓ **9** **0** **0** **ENTER** **4** **.**

900.0000

7 **$\Sigma+$**

3.0000

✓ **1** **0** **0** **0** **ENTER** **4**

1,000.0000

1 **$\Sigma+$**

4.0000

$\bar{x}\bar{y}$ **>** **>** (**$\bar{x}\bar{w}$**)

$\bar{x}$ **$\bar{y}$** **$\bar{x}\bar{w}$**

4 つのデータペアが蓄積されます。

購入量に対して加重された平均価格を計算します。

4.4314

標本標準偏差

標本標準偏差は、データ値が平均値からどれだけ分散しているかの尺度です。

標本標準偏差では、データはより大きい、完全なデータセットの標本で、 $n-1$ を除数として使用しているものとします。

■ x 値の標準偏差を求めるには、 **$S\sigma$** (**σx**) を押します。

■ y 値の標準偏差を求めるには、 **$S\sigma$** **>** (**σy**) を押します。

このメニューの (**σx**) と (**σy**) については、次節「母標準偏差」で説明します。

例：標本標準偏差：

前述の「平均」の例と同じ処理時間を使用して、処理の標準偏差時間 (s_x) を求めます。

15.5	9.25	10.0
12.5	12.0	8.5

時間の標準偏差を計算します。(すべてのデータを x 値として処理します)。

キー：	ディスプレイ：	説明：
(4Σ)		統計レジスタをクリアします。
	1.0000	最初の時間を入力します。
		残りのデータを入力します。
		6 個のデータ点が入力されます。
	6.0000	
(s_x)	s_x s_y σ_x σ_y 2.5808	標準偏差時間を計算します。

母標準偏差

母標準偏差は、データ値が平均からどれだけ散らばっているかの尺度です。母標準偏差は、データが**完全な**データセットを構成し、 n を除数として計算されているとします。

- x 値の母標準偏差を求めるには、 (σ_x) を押します。
- y 値の母標準偏差を求めるには、 (σ_y) を押します。

例：母標準偏差

ある婦人の 4 人の孫の身長が、それぞれ 170、173、174、180 cm であるとして、彼らの身長の母標準偏差を計算します。

キー：	ディスプレイ：	説明：
(4Σ)		統計レジスタをクリアします。
		データを入力します。4 つのデータ点が蓄積されます。
	4.0000	
(σ_x)	s_x s_y σ_x σ_y 3.6315	母標準偏差を計算します。

線形回帰

線形回帰 L.R. (線形推定とも呼ばれます) は、 x, y データセットに一番適合する直線を求めるための統計手法です。

注



STAT ERROR メッセージを回避するため、L.R. メニューの機能を実行する前にデータを入力します。

L.R. (線形回帰) メニュー

メニューキー	説明
$\hat{x}$	データに適合するように計算された直線に基づいて、 y の所定の推定値に対し x を推定 (予想) します
$\hat{y}$	データに適合するように計算された直線に基づいて、 x の所定の推定値に対し y を推定 (予想) します
r	(x, y) データの相関係数。相関係数は、データと計算された線との一致度を表す -1 から $+1$ の範囲の数値です。
m	計算された線の傾き
b	計算された線の y 切片

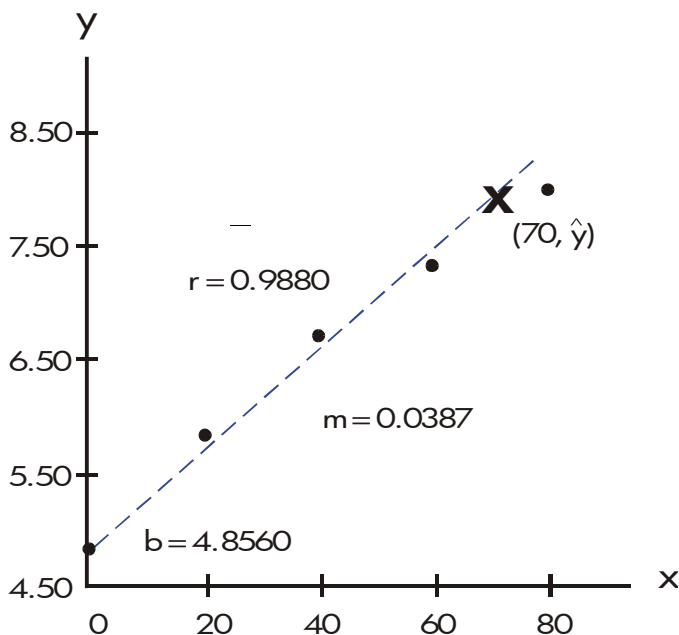
- x (または y) の推定値を求めるには、 y (または x) の所定仮想値を入力し、 (L.R.) ($\hat{x}$) (または  (L.R.)  ($\hat{y}$)) を押します。
- データに一番適合する線を定義する値を求めるには、 (L.R.) を押し、次に r 、 m 、または b を押します。

例：カーブフィッティング

あるお米の収穫量が、窒素肥料の量に依存しているとします。次のデータに対して、相関係数、傾き、 y 切片の直線関係を求めます。

X、窒素肥料の量 (kg/ヘクタール)	0.00	20.00	40.00	60.00	80.00
Y、収穫量 (トン/ヘクタール)	4.63	5.78	6.61	7.21	7.78

キー：	ディスプレイ：	説明：
CLEAR 4 (4Σ)		以前の統計データをクリアします。
✓ 4 0 6 3 ENTER 0 Σ+		データを入力します。 n が表示されます。
✓ 5 0 7 8 ENTER 2 0 Σ+	7.2100 4.0000	
✓ 6 0 6 1 ENTER 4 0 Σ+		
✓ 7 0 2 1 ENTER 6 0 Σ+		
✓ 7 0 7 8 ENTER 8 0 Σ+	7.7800 5.0000	5 個のデータペアが入力されます。
L.R. > > (r)	$\hat{x}$ $\hat{y}$ r m b 0.9880	線形回帰メニューを表示します。
>	$\hat{x}$ $\hat{y}$ r m b 0.0387	相関係数。データが直線を緊密に近似します。
>	$\hat{x}$ $\hat{y}$ r m b 4.8560	y 切片。



70 kg の肥料を水田に与えたらどうなるでしょうか？上記の統計に基づいて収穫量を予測します。

キー：

ディスプレイ：

説明：

✓ **C** **7** **0**

7.7800

想定される x 値を入力します。

70

↵ **L.R.** **>** ($\hat{y}$)

$\hat{x}$ $\hat{y}$ r m b

収穫量の推定値

7.5615

(トン／ヘクタール)

データの精度に関する制限

HP 35s では有限の精度を使っているため、計算には丸めによる制限があります。次に、2 つの例を示します。

近接する大きな数の標準化

データ値の差が比較的小さい変数の場合、電卓が標準偏差や線形推定を正確に計算できない可能性があります。これを避けるには、値をある中央値（平均値など）からの差として入力することで、データを標準化します。標準化された x 値の場合、この差を $\bar{x}$ と $\hat{x}$ の計算に加算しなおす必要があります。 $\hat{y}$ と b も調整する必要があります。例えば、 x 値が 7776999、7777000、7777001 だった場合、データを -1、0、1 として入力した後、7777000 を $\bar{x}$ と $\hat{x}$ に加算する必要があります。 b の場合、 $7777000 \times m$ を加算します。 $\hat{y}$ を計算するには、7777000 より小さい x 値を指定してください。

x 値と y 値の大きさに大きな差がある場合にも、同様に精度が低下します。この場合も、データのスケールリングによって問題を回避できます。

削除したデータの影響

  を実行しても、元のデータ値によって統計レジスタに生成された丸めエラーは削除されません。この差は、不正確なデータが正確なデータと比べて大きすぎない限り、深刻な問題にはなりません。差が大きい場合は、すべてのデータをクリアし、再入力してください。

総和と統計レジスタ

統計レジスタは、6 つの蓄積された総和値を格納する、6 個の固有のメモリ領域です。

総和統計

 **SUMS** を押すと、統計レジスタの内容を確認できます。

- (n) で、蓄積されるデータセットの数を呼び出します。
- Σx を押して、 x 値の合計を呼び出します。
- Σy を押して、 y 値の合計を呼び出します。
- Σx^2 、 Σy^2 、 Σxy を押して、 x 値と y 値の 2 乗の合計と積の合計を呼び出します。これらの値は、電卓によって提供される統計計算に加えてその他の統計計算を実行するときの対象となります。

統計データの入力が終わったら、統計レジスタの内容を表示できます。

 **MEM** **1** (1VAR) **ENTER** を押し、 $\wedge$ と $\vee$ を使用して、統計レジスタを表示します。

例：統計レジスタの表示

$\Sigma +$ を使用して、データペア (1,2) と (3,4) を統計レジスタに格納します。格納された統計値を表示します。

キー：

ディスプレイ：

説明：

 **CLEAR** **4** (4 Σ)

 **2** **ENTER** **1** $\Sigma +$

2.0000

1.0000

統計レジスタをクリアします。

最初のデータペア (1,2) を格納します。

2 番目のデータペア (3,4) を格納します。

 **4** **ENTER** **3** $\Sigma +$

4.0000

2.0000



MEM **1** (1VAR)

$n =$

2.0000

$\wedge$

$\Sigma xy =$

14.0000

↑ VAR カタログを表示し、 n レジ

スタを表示します。

↑ Σxy レジスタを表示します。

↓

	$\Sigma y^2 =$ 20.0000	↑ Σy^2 レジスタを表示します。 ↓
	$\Sigma x^2 =$ 10.0000	↑ Σx^2 レジスタを表示します。 ↓
	$\Sigma y =$ 6.0000	↑ Σy レジスタを表示します。 ↓
	$\Sigma x =$ 4.0000	↑ Σx レジスタを表示します。 ↓
	$n =$ 2.0000	↑ n レジスタを表示します。 ↓
	4.0000 2.0000	VAR カタログを終了します。

統計レジスタへのアクセス

次の表に、HP 35s の統計レジスタの割り当てを示します。総計レジスタは、式、方程式、プログラム内で番号ではなく、名前で参照する必要があります。

統計レジスタ

レジスタ	番号	説明
n	-27	蓄積されるデータペアの数
Σx	-28	蓄積された x 値の合計
Σy	-29	蓄積された y 値の合計
Σx^2	-30	蓄積された x 値の 2 乗の合計
Σy^2	-31	蓄積された y 値の 2 乗の合計
Σxy	-32	蓄積された x 値と y 値の積の合計

統計レジスタに合計をロードするには、レジスタ番号 (-27 から -32) を **I** または **J** に格納し、次に合計 (値 **STO** **(I)** または **(J)**) を格納します。同様に、**VIEW** **(I)** または **(J)** (または **RCL** **(I)** または **(J)**) を押すと、レジスタ値の表示 (または呼び出し) を実行できます。表示には、ラベルとしてレジスタ名が付きます。SUMS メニューにはレジスタ値を呼び出すための機能があります。詳細については、第 14 章「変数とラベルの間接アドレッシング」を参照してください。

第 2 部

プログラミング

プログラムの基礎

本書の第 1 部では、「手動」、すなわち個別演算ごとにキーを押して実行できる、関数と演算について説明しました。また、方程式を使用し、実行のたびにキー入力をすべてやり直すことなく、計算を繰り返す方法についても説明しました。

第 2 部では、反復計算に**プログラム**を使用する方法について説明します。こうした計算には、より多くの入力や出力の制御、より複雑な論理が関係します。プログラムを使用すると、操作や計算を希望どおりに反復できます。

この章では、一連の操作のプログラム方法について説明します。次章「プログラミングテクニック」で、サブルーチンと条件付き命令について説明します。

例：簡単なプログラム

半径 5 の円の面積を計算するには、

公式 $A = \pi r^2$ を使用し、

RPN モード : 5 x^2 $\rightarrow$ π $\times$

ALG モード : 5 y^x 2 $\times$ $\rightarrow$ π ENTER

を押して、この円の結果 78.5398 を求めます。

さまざまな円の**面積**を求める場合はどうでしょうか？

半径 "5" だけを変更して所定のキー入力を繰り返すこともできますが、繰り返し可能なキー入力をプログラムに入れることが可能です。

RPN モード0001 $\times 2$ 0002 π 0003 $\times$ **ALG モード**0001 $SQ(x) \times \pi$

このプログラムでは、プログラムの実行時に半径の値が X レジスタ（ディスプレイ）に存在すると仮定しています。面積を計算し、結果を X レジスタに格納します。

RPN モードでこのプログラムをプログラムメモリに入力するには、次の操作を行います。

キー：(RPN モード) ディスプレイ：**説明：** CLEAR **3**

メモリをクリアします。

(3ALL)  (Y) ENTER PRGM

プログラム入力モードをアクティブにします（**PRGM** インジケータがオンになります）
プログラムポインタを PRGM TOP にリセットします。
(半径) 2

GTO  

PRGM TOP

 x^2 0001 $\times 2$  π 0002 π  $\times$ 0003 $\times$ **面積** = πr^2  PRGM

プログラム入力モードを終了します。

プログラムを実行して半径 5 の円の面積を求めるには、次の操作を行います。

キー：(RPN モード) ディスプレイ：**説明：**GTO  

プログラムの先頭に移動します。

5 

78.5398

答え

ALG モードでこのプログラムをプログラムメモリに入力するには、次の操作を行います。

キー： (ALG モード)	ディスプレイ：	説明：
CLEAR 3		メモリをクリアします。
((3ALL)) (Y) ENTER		
PRGM		プログラム入力モードをアクティブにします (PRGM インジケータがオンになります) プログラムポインタを PRGM TOP にリセットします。
GTO	PRGM TOP	
x^2 RCL X X	0001 SQ(X)×π	面積 = πx^2
π		
PRGM		プログラム入力モードを終了します。

プログラムを実行して半径 5 の円の面積を求めるには、次の操作を行います。

キー： (ALG モード)	ディスプレイ：	説明：
GTO		プログラムの先頭に移動します。
5 STO X ENTER	5→X	5 を X に格納します
	5.0000	
R/S	78.9358	答え

引き続き円の面積を求める上記プログラムを使用して、プログラミングの概念と方法を説明します。

プログラムの設計

次の項目に、プログラム行に入力できる命令を示します。プログラムに入力する命令によって、表示の際の現れ方や実行時の動作が変わります。

モードの選択

RPN モードで作成して保存したプログラムの編集と実行は、RPN モードで行う必要があります。ALG モードで作成して保存したプログラムや手順の編集と実行は、ALG モードで行う必要があります。そうでないと、正しい結果は得られません。

プログラムの境界 (LBL と RTN)

複数のプログラムをメモリに保存するときには、プログラムの先頭を示すラベル (R0001 LBL R など) と、最後を示すリターン (R0005 RTN など) が必要です。行番号には、ラベルに対応して R が付いています。

プログラムラベル

プログラムとそのセグメント (ルーチン) は、ラベルから開始する必要があります。ラベルを記録するには、次を押します。

LBL 文字キー

ラベルは、A から Z の 1 文字です。文字キーの使用法は、変数の場合と同じです (第 3 章を参照してください)。同じラベルを重複して割り当てることはできません (重複した場合、メッセージ DUPLICATE LBL が表示されます) が、変数が使用している文字をラベルに使用することができます。

メモリに 1 つのプログラム (トップのプログラム) しかないときは、ラベル無しで登録できます。しかし、隣り合うプログラムには、プログラム同士を区別するためラベルが必要です。

プログラムには、999 行を超える行を入力できません。

プログラムの return

プログラムとサブルーチンは、return で終了する必要があります。キー入力は次の通りです。

RTN

プログラムが実行を終了したとき、最後の RTN 命令によりポインタが PRGM TOP (プログラムのトップ) に戻ります。

プログラムでの RPN、ALG、方程式の使用

キーボードで計算するときと同じ方法が、プログラムでも利用できます。

- RPN 操作の使用 (スタックを利用、第 2 章で説明)
- ALG 操作の使用 (付録 C で説明)
- 方程式の使用 (第 6 章で説明)

13-4 プログラムの基礎

前の例では、一連の **RPN 操作** を使用して円の面積を計算しました。プログラムでは、**方程式**を利用することができます（例については、この章で後述します）。多くのプログラムは、RPN と 方程式を組み合わせることで、両方の利点を利用しています。

RPN 操作の利点

メモリ使用量が少ない
実行速度が速い

方程式と ALG 操作の利点

読み書きが簡単
プロンプトを自動で表示できる

プログラムが方程式を含む行を実行すると、その方程式は、**[XEQ]** が方程式リストの方程式を評価する場合と同じ方法で評価されます。プログラムの評価では、方程式内の "=" が、基本的に "-" として処理されます。（代入式の場合、**[ENTER]** と等価の機能をプログラムできません。方程式を式として書いてから、**STO** を使用して値を変数に格納する必要があります）。

どちらのタイプの計算でも、入力、出力、プログラムフローを制御するため、RPN 命令を定義することができます。

データの入出力

複数の入力が必要なプログラムや複数の出力を返すプログラムの場合、プログラムが情報を入力する方法や返す方法を指定することができます。

入力の場合、INPUT 命令によって変数の入力を促すことができます。または、あらかじめスタックに入力された値を取り込むことができます。

出力の場合、VIEW 命令により、変数の表示、方程式から派生したメッセージの表示、ディスプレイ第 1 行でのプロセスの表示、ディスプレイ第 2 行でのプログラム結果の表示が行えます。またはマークされていない値をスタックに残すことができます。

詳細については、この章で後述の「データの入力と表示」を参照してください。

プログラムの入力

 **PRGM** を押すと、プログラム入力モードのオンとオフが切り替わります。

PRGM インジケータがオンまたはオフになります。プログラム入力モードでのキー入力は、メモリにプログラム行として格納されます。各命令（コマンド）や式は、1 つのプログラム行を占有します。ALG モードでは、式をプログラムに直接入力できます。

プログラムをメモリに入力するには：

1.  **PRGM** を押して、プログラム入力モードをアクティブにします。
2.   を押して、PRGM TOP を表示します。これにより**プログラムポインタ**がその他のプログラムの前の、既知のスポットにセットされます。プログラム行を入力すると、プログラム行は、その他のプログラム行の**前に**挿入されます。

メモリ内のプログラムが不要になったときは、 **CLEAR**  (3PGM) を押してプログラムメモリをクリアします。**すべての**プログラムを削除することを確認するため、メッセージ CLR PGMS? Y_N の後に  (Y)  を押します。

3. プログラムに**ラベル**として A から Z の 1 文字を指定します。 **LBL** **文字** を押します。面積には "A" など、プログラムに関係する文字を選択します。

メッセージ DUPLICAT LBL が表示された場合は、違う文字を使用します。あるいは、既存のプログラムをクリアすることもできます。

 **MEM**  (2PGM) を押し、 または  を使用してラベルを探し、 **CLEAR** と  を押します。

4. 電卓での操作をプログラム命令として記録するには、手動の操作と同じキーを押します。多くの関数は、キーボードに表示されておらず、メニューを使って呼び出す必要があります。
プログラム行に方程式を入力するには、下の命令を参照してください。
5. *return* 命令でプログラムを終了します。これにより、プログラムの実行後、プログラムポインタが PRGM TOP に戻ります。 **RTN** を押します。
6.  (または  **PRGM**) を押して、プログラム入力をキャンセルします。

プログラム行における数値は、入力した通りの精度で格納され、表示形式は ALL または SCI になります。(長い数値が短く表示されるときは、 **SHOW** を押してすべての桁を表示します)。

プログラム行に方程式を入力するには：

1. **EQN** を押して、方程式入力モードをアクティブにします。**EQN** インジケータがオンになります。
2. 方程式リストの場合と同様に、方程式を入力します。詳細については、第 6 章を参照してください。 を使用して入力時のエラーを訂正します。
3. **ENTER** を押して方程式を終了し、左端を表示します。(方程式は、方程式リストに保存されません)。

方程式の入力後、 **SHOW** を押してチェックサムと長さを表示できます。値をディスプレイに表示し続けるには、**SHOW** キーを押したままにします。

長い方程式では、プログラム行のスクロールがアクティブであることを示す ➡ インジケータと ◀ インジケータが表示されます。 ◀ と  ▶ を使用して、ディスプレイをスクロールできます。

クリア関数とバックスペースキー

プログラム入力中には、通常と異なる状態が発生します。

- **C** は、常にプログラム入力をキャンセルします。通常のように数値をクリアしてゼロにすることはありません。
- プログラム行の閲覧時には、 を押すと表示中のプログラム行が削除され、◀/▶ で編集を開始します。プログラム行の編集状態では、 がカーソル直前の文字を削除します。
- X レジスタをクリアする関数をプログラムするには、 **CLEAR** **1** (1×) を使用します。

プログラムに行を挿入したり削除するときは、必要に応じて GTO 文と XEQ 文が自動的に更新されます。

例：

A001 LBL A
 A002 2+3
 A003 1+2
 A004
 GTO A003

行 A002 を削除すると、行 A004 は“A003 GTO A002”に変わります。

プログラム内の関数名

プログラム行で使用する関数の名前を、キー、メニュー、方程式内の関数名と同じにする必要は**ありません**。プログラムでは通常、キーやメニューの場合よりもさらに短縮された名前が用いられます。

例：ラベル付きのプログラムを入力

次のキー入力により、以前の円の面積を計算するプログラムが削除され、ラベルと return 命令を含む新しいプログラムが入力されます。入力ミスがあれば、 を押して現在のプログラム行を削除し、正しい行を再入力します。

キー：(RPN モード)

ディスプレイ：

説明：

 PRGM

プログラム入力モードをアクティブにします
 (PRGM がオン)
 すべてのプログラムメモリをクリアします。

 CLEAR 

(3PGM)  (Y)

PRGM TOP



 LBL 

A001 LBL A

このプログラムルーチンのラベルを A とします (area の A)。
 3 行のプログラム行を入力します。



A002 $\times 2$



A003 π



A004 $\times$



A005 RTN

プログラムを終了します。

 MEM  (2PGM)

LBL A
 LN=15

ラベル A とプログラムの長さ (バイト数) を表示します。
 プログラムのチェックサムと長さ。



CK=DAF1
 LN=15

13-8 プログラムの基礎

C **C**

プログラム入力をキャンセルします (**PRGM** インジケータがオフ)。

チェックサムが異なっているときは、プログラムに入力ミスがあることを示します。

例：方程式を含むプログラムを入力

次のプログラムは、前のプログラムのような RPN 操作ではなく、方程式を使用して円の面積を計算します。

キー：(RPN モード)

ディスプレイ：

説明：

PRGM **GTO** **•**
•

PRGM TOP

プログラム入力モードをアクティブにします。ポインタはメモリのトップに移動します。このプログラムルーチンのラベルを E とします (equation の E)。

LBL **E**

E001 LBL E

STO **R**

E002 STO R

半径を変数 R に格納します。

EQN **π**

方程式入力モードを選択し、方程式を入力して、プログラム入力モードに戻ります。

x **RCL** **R**

y^x **2** **ENTER**

E003 $\pi \times R^2$

SHOW

CK=7E5B

LN=5

RTN

E004 RTN

プログラムを終了します。ラベル E とプログラムの長さ (バイト数) を表示します。

MEM **2** (2PGM)

LBL E

LN=17

プログラムのチェックサムと長さ。プログラム入力をキャンセルします。

SHOW

CK=2073

LN=17

C **C**

プログラムの実行

プログラムを動作または**実行**するには、プログラム入力をオフにしておく必要があります（プログラムの行番号が表示されておらず、**PRGM** がオフの状態）。

[C] を押すと、プログラム入力モードをキャンセルできます。

プログラムの実行（XEQ）

[XEQ] ラベルを押して、その文字をラベル名とするプログラムを実行します。

プログラムをその最初の行から実行するには、**[XEQ]** ラベル **[ENTER]** を押します。例えば、**[XEQ]** **[A]** **[ENTER]** を押します。ディスプレイに "XEQ R001" と表示され、ラベル A の最初の行から実行が開始されます。

[XEQ] ラベル 行番号を押すと、プログラムの実行を別の位置から開始することもできます（例えば **[XEQ]** **[A]** **[0]** **[0]** **[5]**）。

プログラムがメモリに 1 つしかない場合は、ポインタを最初の行に移動させ、**[R/S]** (run/stop) キーを押して実行することもできます。**PRGM** インジケータが表示され、プログラムの実行中は **[B]** インジケータが点灯します。

必要に応じて、プログラムの実行前にデータを入力します。

例：

ラベル A と E のプログラムを実行し、半径 5、2.5、 2π の 3 個の異なる円の面積を求めます。A または E の実行前に半径を入力してください。

キー：(RPN モード) ディスプレイ：

[5] **[XEQ]** **[A]** **[ENTER]** RUNNING
78.5398

[2] **[.]** **[5]** **[XEQ]** **[E]** 19.6350
[ENTER]

[2] **[π]** **[$\times$]**
[XEQ] **[A]** **[ENTER]** 124.0251

説明：

半径を入力し、プログラム A を開始します。計算結果の面積が表示されます。
プログラム E を使用して 2 番目の円の面積を計算します。
3 番目の円の面積を計算します。

プログラムのテスト

プログラムにエラーがあるのにエラーの場所を特定できない場合、テストでプログラムを 1 ステップずつ実行します。長いプログラムや複雑なプログラムの場合も、テストで信頼性を確認します。1 行ずつ実行すると、それぞれの行が実行されるたびに結果が表示されるので、正しい結果がわかっている既知のデータで進行状況を確認できます。

1. 通常の実行と同様に、プログラム入力がアクティブでないことを確認します（**PRGM** インジケータがオフ）。
2. プログラムポインタを最初の行にセットします（**LBL** 命令で）。この命令は、プログラムポインタを移動するだけで、実行を開始しません。
3. ☒ を押し続けます。これにより、現在のプログラム行が表示されます。
☒ を離すと、その行が実行されます。次に実行結果が表示されます（結果は X レジスタに格納されています）。
前の 行に移動するには、☐ を使用できます。プログラム行は実行されません。
4. プログラムポインタを次の行に移動します。エラー（不正確な結果の発生）を発見できるまで、またはプログラムの最後に達するまで、ステップ 3 を繰り返します。

プログラム入力モードがアクティブの場合、☒ または ☐ はプログラムポインタを移動するだけで、プログラム行は実行されません。プログラム入力中にカーソルキーを押し続けると、行が自動的にスクロールします。

例：プログラムをテストする

プログラム A を 1 行ずつ実行します。テストのデータとして半径 5 を使用します。プログラムの開始前に、プログラム入力モードがアクティブでないことを確認します。

キー：(RPN モード)	ディスプレイ：	説明：
☒ GTO ☒ A ENTER	5.0000	プログラムカウンタをラベル A に移動します。
☒ (hold) (release)	A001 LBL A 5.0000	
☒ (hold) (release)	A002 $\times 2$ 25.0000	入力を 2 乗します。
☒ (hold) (release)	A003 π 3.1416	π の値

 (hold) (release)	A004 × 78.5398	25π.
 (hold) (release)	A005 RTN 78.5398	プログラムの終了。結果が正しいことが確認できました。

データの入力と表示

電卓の**変数**は、データ入力、中間結果、最終結果の格納に使用されます。(変数は、第3章で説明したように、AからZの1文字で識別されますが、変数名は、プログラムラベルとは無関係です)。

プログラムでは、次の方法でデータを取得できます。

- INPUT 命令から。変数の値の入力を求めます。(最も便利な方法です)。
- スタックから。(STO を使って値を変数に格納し、後から使用することもできます)。
- すでに値が格納された変数から。
- 自動方程式プロンプトから (フラグ 11 がセットされている場合に有効)。(方程式を使用している場合にも便利です)。

プログラムでは、次の3種類の方法で情報を表示できます。

- VIEW 命令を使用。変数の名前と値を表示します。(最も便利な方法です)。
- スタック上。XレジスタとYレジスタの値のみが表示されます。(PSE を使用してXレジスタとYレジスタを1秒間表示できます)。
- 表示方程式内 (フラグ 10 がセットされている場合に有効)。(「方程式」は通常、メッセージで、真の方程式ではありません)。

次節で、これらの入出力方法の一部について説明します。

INPUT を使用したデータの入力

INPUT 命令 ( INPUT 変数) があると、そこでプログラムの実行が停止し、所定の変数のプロンプトが表示されます。変数に対する既存の値が、次のように表示されます。

R?
0.0000

ここで、

"R" は変数名です。

"?" は情報の入力を求めるプロンプトです。

0.0000 は、変数に格納されている現在の値です。

 (run/stop) を押して、プログラムを再開します。入力した値は、X レジスタの内容に上書きされ、さらに所定の変数に格納されます。表示値を変更しなかった場合、その値は X レジスタに保持されます。

円の面積のプログラムで INPUT 命令を使用すると、次のようになります。

RPN モード	ALG モード
A001 LBL A	A001 LBL A
A002 INPUT R	A002 INPUT R
A003 $\times 2$	A003 $SQ(R) \times \pi$
A004 π	A004 RTN
A005 $\times$	
A006 RTN	

プログラムで INPUT 関数を使用するには：

1. どのデータ値が必要かを決定し、名前を割り当てます。
(円の面積の例では、必要な入力半径のみです。R に割り当てることができます)。
2. プログラムの最初に、値が必要な変数それぞれの INPUT 命令を挿入します。
プログラムの後のほうで、所定の値が必要な計算部分を作成するときに、**[RCL]** 変数 命令を挿入して値をスタックに戻します。

INPUT 命令によって入力した値が X レジスタにも残るため、後からその変数を呼び出す必要はありません。INPUT を実行し、必要なときに使用できます。その分メモリの使用量を削減できます。ただし長いプログラムでは、前もってすべてのデータを入力し、必要に応じて個別の変数を呼び出す方が簡単です。

プログラムが停止し、入力を待っている状態でも、プログラムで計算を実行できます。計算を行うとスタックの内容が変更され、その後のプログラムによる計算に影響が出る可能性があります。このため、プログラムで、INPUT 命令の前後で X レジスタ、Y レジスタ、Z レジスタの内容が同じであると想定することは禁物です。プログラムの開始時にすべてのデータを収集し、あとで計算に必要なときに呼び出す場合、これにより、スタックの内容が計算直前に変化することを防止できます。

プロンプトへ応答するには：

プログラムを実行するとき、プログラムが各 INPUT で停止し、R?0.0000 など、変数の入力が求められます。表示される値（および X レジスタの内容）が、R の現在の内容になります。

- 数値をそのままにするには、**[R/S]** を押します。
- 数値を変更するには、新しい数値を入力し、**[R/S]** を押します。新しい数値は X レジスタの古い値に上書きされます。必要に応じて、数値を分数として入力することもできます。数値を計算する必要がある場合、通常のキーボード計算を使用してから、**[R/S]** を押します。例えば、RPN モードでは **[2] [ENTER] [5] [y^x] [R/S]**、ALG モードでは **[2] [y^x] [5] [ENTER] [R/S]** を押すことができます（**[ENTER]** を押す前に、式が第 2 行に表示されます。**[ENTER]** を押すと、式の結果が、式に代わって第 2 行に表示され、X レジスタに格納されます）。

13-14 プログラムの基礎

- **INPUT** プロンプトをキャンセルするには、**[C]** を押します。変数の現在の値は、X レジスタ内に残ります。**[R/S]** を押してプログラムを再開すると、キャンセルした **INPUT** プロンプトが再度現れます。数値の入力中に **[C]** を押すと、数値がゼロにクリアされます。**[C]** を再度押して、**INPUT** プロンプトをキャンセルします。

VIEW を使用したデータの表示

プログラムされた **VIEW** 命令 (**[VIEW]** 変数) があると、そこでプログラムの実行が停止し、所定の変数の内容が次のように表示され、識別されます。

R=
78.5398

これは**表示のみ**であり、数値を X レジスタへコピーしません。分数表示モードがアクティブの場合、値は分数として表示されます。

- **[ENTER]** を押すと、この数値が X レジスタにコピーされます。
- 数値が 2 進数、複素数、ベクトルなどで 14 桁より大きい場合、**[R]****[<]** と **[R]****[>]** で残りを表示します。
- **[C]** (または **[←]**) を押すと、**VIEW** の表示が消え、X レジスタが表示されます。
- **[R]****[CLEAR]** を押すと、表示中の変数の内容をクリアします。

[R/S] を押してプログラムを続行します。

プログラムを停止させたくない場合、後述の「停止なしの情報の表示」を参照してください。

例については、第 16 章の「正規分布と逆正規分布」のプログラムを参照してください。ルーチン T の最後で T015 と T016 が X の結果を表示しています。このプログラムでは、**VIEW** 命令が **RCL** 命令の直後にあります。**RCL** 命令は必須ではありませんが、これにより **VIEW** で表示した変数が X レジスタにコピーされ、手動での計算に利用できるのも便利です。(VIEW が表示されているときに **[ENTER]** キーを押しても同じ効力があります)。第 16 章と第 17 章のその他のアプリケーションプログラムでも、**VIEW** で表示された変数が X レジスタに格納されるように設計されています。

方程式を使用したメッセージの表示

方程式は、評価されるまで文法がチェックされません。このため、**ほぼすべての文字シーケンス**を方程式としてプログラムに入力できます。入力方法は、**任意の**方程式を入力する場合と同じです。プログラム行で、**[EQN]**を押して方程式を開始します。数値キーと演算キーを押して、数値と記号を取得します。各文字の前に**[RCL]**を押します。**[ENTER]**を押して方程式を終了します。

フラグ 10 がセットされている場合、方程式が**評価**される代わりに**表示**されます。これにより、入力したメッセージを方程式として表示できます。(フラグについては、第 14 章で説明します)。

メッセージが表示されると、プログラムは中断します。再開するには **[R/S]** を押します。表示されたメッセージが 14 文字より多い場合、メッセージが表示されたときに **▶** インジケータが点灯します。**[▶]** **[>]** と **[▶]** **[<]** を使用して、ディスプレイをスクロールできます。

プログラムを停止させたくない場合は、後述の「停止なしの情報の表示」を参照してください。

例：プログラムにおける INPUT、VIEW、メッセージ

半径と高さが指定された場合に、円筒の表面積と体積を求める方程式を作成します。プログラム名を **C** (cylinder の **C**) にし、変数 **S** (表面積)、**V** (体積)、**R** (半径)、**H** (高さ) を使用します。次の公式を使用します。

$$V = \pi R^2 H$$

$$S = 2\pi R^2 + 2\pi RH = 2\pi R(R + H)$$

キー：(RPN モード)	ディスプレイ：	説明：
[▶] [PRGM] [▶]		プログラム、エントリ。
[CLEAR] [3] (3PGM)	PRGM TOP	プログラムメモリをクリアします。
[<] [Y] [ENTER]		
[▶] [LBL] [C]	C001 LBL C	プログラムにラベルを付けます。
[G↓] [INPUT] [R]	C002 INPUT R	
[G↓] [INPUT] [H]	C003 INPUT H	半径と高さの入力を求めるよう指示します。

13-16 プログラムの基礎

キー：(RPN モード)

EQN π $\times$
 RCL R y^x 2 $\times$
 RCL H ENTER

π SHOW

STO V
 EQN 2 $\times$ π $\times$ RCL R $\times$
 () RCL R +
 RCL H ENTER
 π SHOW

STO S
 π FLAGS 1
 (1SF) $\div$ 0
 EQN RCL V
 RCL O RCL L
 $\rightarrow$ SPACE + $\rightarrow$
 SPACE RCL A
 RCL R RCL E
 RCL A ENTER
 π FLAGS 1

(2CF) $\div$ 0
 π VIEW V
 π VIEW S
 π RTN
 π MEM 2
 (2PGM)

π SHOW

C C

ディスプレイ：

C004 $\pi \times R^2 \times H$

CK=74FE

LN=7

C005 STO V

C006 $2 \times \pi \times R \times (R + H)$

CK=19B3

LN=11

C007 STO S

C008 SF 10

C009 VOL + ARE

C010 CF 10

C011 VIEW V

C012 VIEW S

C013 RTN

LBL C

LN=67

CK=97C3

LN=67

説明：

体積を計算します。

方程式のチェックサムと長さ。

体積を V に格納します。
 表面積を計算します。

方程式のチェックサムと長さ。

表面積を S に格納します。
 方程式を表示するためフラグ 10 をセットします。
 方程式のメッセージを表示します。

フラグ 10 をクリアします。

体積を表示します。
 表面積を表示します。
 プログラムを終了します。
 ラベル C とプログラムの長さ（バイト数）を表示します。
 プログラムのチェックサムと長さ
 プログラム入力をキャンセルします。

半径 $2\frac{1}{2}$ cm、高さ 8 cm の円筒の体積と表面積を求めます。

キー:	ディスプレイ:	説明:
(RPN モード)		
XEQ C ENTER	R?	C の実行を開始します。R の
	値	入力を求めます。(そのときに
2 . 1 . 2	H?	R に存在する値を表示します)。
R/S	値	$2\frac{1}{2}$ を分数として入力しま
8 R/S	VOL + AREA	す。H の入力を求めます。
R/S	V=	メッセージが表示されます。
	157.0796	体積 (cm ³)
R/S	S=	表面積 (cm ²)
	164.9336	

停止なしの情報の表示

通常、VIEW で変数を表示するときや、方程式メッセージを表示するときにはプログラムが停止します。再開するには **R/S** を押す必要があります。

必要に応じて、情報の表示中でもプログラムの実行を続けることができます。VIEW 命令または表示された方程式後、**次の** プログラム行に PSE (pause) 命令が含まれる場合、情報が**表示され**、1 秒間の停止後に実行が継続します。この場合、スクロールやキーボード入力ができなくなります。

他の表示操作があれば、ディスプレイはクリアされます。また、フラグ 7 がセットされており、RND 操作 (分数への丸め) が実行された場合も、ディスプレイがクリアされます。

PSE を押して、プログラムに PSE を入力します。

プログラムを 1 行ずつ実行するとき、VIEW 行と PSE 行 (または方程式と PSE 行) が 1 つの操作として処理されます。

プログラムの中断と一時停止

中断（STOP）または一時停止（PSE）のプログラミング

- プログラム入力中に **[R/S]** (*run/stop*) を押すと、STOP 命令が挿入されます。STOP は X レジスタの内容を表示し、キーボードから **[R/S]** を押して再開するまでプログラムの実行を中断します。プログラムポインタをメモリのトップに移動させずにプログラムを終了するには、RTN ではなく STOP を使用します。
- プログラム入力中に **[PSE]** を押すと、PSE (*pause*) 命令が挿入されます。PSE はプログラムの実行を一時停止し、X レジスタの内容を約 1 秒間表示します。ただし、PSE が VIEW 命令または表示された方程式（フラグ 10 がセットされている状態）の直後にあるときは、その変数または方程式が表示され、1 秒間の停止後にもその表示が残ります。

実行中のプログラムの停止

実行中のプログラムは、**[C]** または **[R/S]** を押すといつでも停止できます。停止前に現在の命令を完了します。**[R/S]** (*run/stop*) を押して、プログラムを再開します。

プログラムの停止後に **[XEQ]**、**[GTO]**、または **[RTN]** を押すと、**[R/S]** によってプログラムを再開できなくなります。その場合はプログラムを再実行してください（**[XEQ]** ラベル行番号）。

エラーによる停止

プログラム実行中にエラーが発生すると、プログラム実行が停止してエラーメッセージが表示されます。（付録 F にメッセージとその状態の一覧があります）。

エラーの原因となった命令を含むプログラム行を表示するには、**[PRGM]** を押します。プログラムがそのポイントで停止しています。（÷ 命令により、ゼロによる除算が発生した場合など）。

プログラムの編集

プログラム行の挿入、削除、編集により、プログラムメモリ内のプログラムを変更できます。プログラム行に方程式が含まれる場合、その方程式も編集できます。

プログラム行を削除するには：

1. 関連のプログラムまたはルーチンを選択し、 または  を押して、変更する必要があるプログラム行を見つけます。スクロールし続けるには、カーソルキーを押したままにします。
2. 変更する行を削除します。 を直接押します（Undo 機能がアクティブになります）。ポインタがその**前の** 行に移動します。（複数の連続したプログラム行を削除する場合は、そのグループの**最後の** 行から開始します）。
3. 必要に応じて新しい命令を入力します。これが、削除された行に置き換わります。
4. プログラム入力を終了します（ または  PRGM）。

プログラム行を挿入するには：

1. 行を挿入する位置の**前の**プログラム行を選択して表示します。
2. 新しい命令を入力します。現在表示されている行の**後ろ**に挿入されます。

例えば、プログラムの A004 行と A005 行の間に新しい行を挿入したいときは、A004 を表示してから命令を入力します。元の A005 行に続く行は後方に移動し、それに従って行番号が変更されます。

プログラム行内の演算対象、式、方程式を編集するには：

1. 編集するプログラム行を指定するか、表示します。
2.  または  を押して、プログラム行の編集を開始します。これにより編集カーソル “_” がオンになりますが、プログラム行内での削除は行われません。
 キーによりプログラム行の左のカーソルがアクティブになります。
 キーによりプログラム行の末尾のカーソルがアクティブになります。
3. カーソル “_” を移動し、 を繰り返し押して、不要な数値または関数を削除します。次にプログラム行の残りを再入力します。（ を押すと、Undo 機能がアクティブになります）

注意

1. プログラム行の編集カーソルがアクティブのとき、 または  キーが無効になります。
2. プログラム行の編集集中で（カーソルがアクティブのとき）、プログラム行に何も無いときには、 を押しても効力がありません。そのプログラム行を削除したい場合、 を押すとプログラム行が消去されます。
3.  キーと  キーを使用して、長いプログラム行を編集せずに確認できます。
4. ALG モードでは、 を関数として利用できません。プログラム行の検証に使用されます。
5. 方程式は、最初のモードが何であっても、任意のモードで編集できます。

プログラムメモリ

プログラムメモリの表示

  を押すと、プログラム入力のオンとオフが切り替わります（**PRGM** インジケータがオンになり、プログラム行が表示されます）。プログラム入力モードがアクティブのとき、プログラムメモリの内容が表示されます。

プログラムメモリは、PRGM TOP から開始します。プログラムのリストは循環しているため、プログラムポインタは、末尾から先頭（およびその逆）にジャンプします。プログラム入力がアクティブのときにプログラムポインタ（表示中の行）を変更するには、次の 4 種類の方法があります。

-   と   を使用すると、ラベル間を移動できます。ラベルが定義されていなければ、プログラムの先頭または末尾に移動します。
- 複数の行を一度に移動（スクロール）するには、 キーまたは  キーを押したままにします。
-    を押して、プログラムポインタを PRGM TOP に移動します。
-   *label nnn* を押して、特定の行に移動します。

プログラム入力モードがアクティブでないとき（プログラム行が表示されていないとき）でも、**[GTO]** ラベル行番号 を押してプログラムポインタを移動できます。

プログラム入力モードをキャンセルしても、プログラムポインタの位置は変更されません。

メモリの使用

プログラム入力中に MEMORY FULL というメッセージが表示された場合は、入力しようとしたプログラム行に対して、十分な空き領域がプログラムメモリにありません。プログラムや他のデータをクリアすることで、領域を確保できます。後述の「プログラムのクリア」、または付録 B「電卓のメモリ管理」を参照してください。

プログラムのカタログ (MEM)

プログラムのカタログは、すべてのプログラムラベルと、各ラベルおよびそれに関連づけられたプログラム行によって使用されるメモリのバイト数のリストです。**[<]** **[MEM]** **[2]** (2PGM) を押してカタログを表示し、**[v]** または **[^]** を押してリスト内を移動します。カタログは次に利用できます。

- プログラムメモリにおけるラベルと、それぞれのラベル付きプログラムまたはルーチンのメモリコストの確認。
- ラベル付きプログラムの実行（ラベルの表示中に、**[XEQ]** または **[R/S]** を押します）。
- ラベル付きプログラムの表示（ラベルの表示中に、**[>]** **[PRGM]** を押します）。
- 特定のプログラムの削除（ラベルの表示中に、**[>]** **[CLEAR]** を押します）。
- 所定のプログラムセグメントに関連したチェックサムの確認（**[<]** **[SHOW]** を押します）。

カタログに、それぞれのラベル付きプログラムのセグメントが使用するメモリのバイト数が表示されます。プログラムは、プログラムラベルによって識別されます。

LBL C
LN=67

67 は、そのプログラムが使用するバイト数です。

プログラムのクリア

特定のプログラムをメモリからクリアするには

1.  **MEM** **2** (2PGM) **ENTER** を押し、( と  を使用して) プログラムのラベルを表示します。
2.  **CLEAR** を押します。
3. **C** を押してカタログをキャンセルするか、 を押して戻ります。

すべてのプログラムをメモリからクリアするには：

1.  **PRGM** を押して、プログラム行を表示します (**PRGM** インジケータがオンになります)。
2.  **CLEAR** **3** (3PGM) を押して、プログラムメモリをクリアします。
3. メッセージ CLR PGMS? Y N により確認が求められます。 (Y) **ENTER** を押します。
4.  **PRGM** を押して、プログラム入力をキャンセルします。

すべてのメモリをクリアすると ( **CLEAR** **3** (3ALL))、プログラムもすべてクリアされます。

チェックサム

チェックサムは、各プログラムラベルとそれに関連づけられた (次のラベルまでの) プログラム行に対する、固有の 16 進数の数値です。この数値は、プログラムメモリに入力した既存のプログラムの既知のチェックサムとの比較に利用できます。既知のチェックサムと電卓によって表示されたチェックサムが同じであれば、すべてのプログラム行が正しく入力されています。チェックサムを表示するには：

1.  **MEM** **2** (2PGM) **ENTER** を押して、プログラムラベルのカタログを表示します。
2. 必要に応じて、カーソルキーを使用して目的のラベルを表示します。
3.  **SHOW** を押したままにして、CK= チェックサムと LN= 長さ を表示します。

例えば、現在のプログラム（「円筒」プログラム）のチェックサムを表示します。

キー：	ディスプレイ：	説明：
(RPN モード)		
 MEM 2	LBL C	ラベル C を表示します。
(2PGM) ENTER	LN=67	67 バイトを消費しています。
 SHOW (hold)	CK=97C3	チェックサムと長さ
	LN=67	

チェックサムがこの数値と適合しない場合、このプログラムを正しく入力していません。

第 16 章と第 17 章のすべてのアプリケーションプログラムには、チェックサム値がラベル付きのルーチンごとに記載されており、プログラム入力の確度を確認できます。

プログラム内のそれぞれの方程式にもチェックサムがあります。この章で前述の「プログラム行に方程式を入力するには」を参照してください。

プログラムできない機能

HP 35s の次の機能は、プログラムできません。

 CLEAR 3 (3PGM)	GTO  
 CLEAR 3 (3ALL)	GTO  ラベル行番号
	 MEM
 ,  ,  , 	 SHOW
 PRGM	EQN
  ,  	 FDISP
 UNDO	 CLEAR 6 (6CLVAR×)

基数変換（BASE）を使ったプログラム

 **BASE** を使用して基数モードを変更する手順をプログラムできます。これらの設定は、プログラムでキーボードから基数変換するときと同様に動作します。これにより、4 種類の基数の数値を受け入れ、任意の基数で演算を実行し、結果を表示するプログラムを作成できます。

10 進数以外の基数の数値を使用するプログラムを作成するときは、電卓の現在の設定として、およびプログラム内で（命令として）基数モードを設定します。

プログラムにおける基数モードの選択

BIN、OCT、または HEX 命令をプログラムの最初に挿入します。通常、プログラムの最後に DEC 命令を含めて、プログラムの実行後に 10 進数モードに戻るようします。

基数モードを変更するプログラム内の命令は、**プログラム実行中と実行後の入力の解釈方法と出力の表示方法を決定しますが**、入力するプログラム行には影響しません。

プログラム行に入力された数値

プログラムの入力を開始する前に、基数モードを設定します。現在の基数モード設定によってプログラムの結果が決まります。

インジケータで現在の設定の基数を確認できます。以下で、10 進数モードとそれ以外のモードのプログラム行を比較します。すべての 10 進数および非 10 進数が、電卓のディスプレイに左揃えで表示されます。

10 進数モード設定：

```
      :  
      :  
      :  
PRGM  
A009 BIN  
  
A010 10  
  
      :  
      :  
      :
```

10 進数は記号 “d” を省略できます

2 進数モード設定：

```
      :  
      :  
      :  
PRGM BIN  
A009 BIN  
  
A010 10b  
  
      :  
      :  
      :
```

2 進数は記号 “b” を追加する必要があります

多項式とホーナーの方法

多項式のような式では、同じ変数が解に何度も使用されます。例えば、次の式

$$Ax^4 + Bx^3 + Cx^2 + Dx + E$$

は、変数 x を 4 回使用します。RCL 操作を使用する式などを計算するプログラムでは、変数から x の保存済みコピーを繰り返し呼び出すことができます。

例：

RPN 操作で $5x^4 + 2x^3$ を解くプログラムを作成し、 $x = 7$ として評価します。

キー：	ディスプレイ：	説明：
(RPN モード)		
PRGM GTO		
	PRGM TOP	
LBL A	A001 LBL A	
INPUT X	A002 INPUT X	
5	A003 5	5
RCL X	A004 RCL X	
4	A005 4	
y^x	A006 y^x	x^4
X	A007 $\times$	$5x^4$
RCL X	A008 RCL X	
3	A009 3	
y^x	A010 y^x	x^3
2	A011 2	
X	A012 $\times$	$2x^3$
+	A013 +	$5x^4 + 2x^3$
RTN	A014 RTN	
MEM 2	LBL A	ラベル A を表示します。
(2PGM)	LN=46	46 バイトを消費しています。
SHOW	CK=ER18	チェックサムと長さ
	LN=46	
C C		プログラム入力をキャンセルします。

$x = 7$ でこの多項式を評価します。

キー：
(RPN モード)

ディスプレイ：

説明：

XEQ **A** **ENTER**

X?

x の入力をお願いします。

7 **R/S**

値

12,691,0000

結果。

方程式に対するより汎用性の高い形式

$$Ax^4 + Bx^3 + Cx^2 + Dx + E$$

A001 LBL A
A002 INPUT A
A003 INPUT B
A004 INPUT C
A005 INPUT D
A006 INPUT E
A007 INPUT X
A008 RCL X
A009 RCL× A
A010 RCL+ B
A011 RCL× X
A012 RCL+ C
A013 RCL× X
A014 RCL+ D
A015 RCL× X
A016 RCL+ E
A017 RTN

チェックサムと長さ：9E5E 51

プログラムのテクニック

第 13 章ではプログラミングの基礎について説明しました。この章では、次に示すような、より複雑で有用なテクニックについて説明します。

- サブルーチンの使用。プログラムを簡潔にするため、プログラムの一部を特定のタスク専用にして、その部分にラベルを付けます。サブルーチンを使用すれば、短いプログラムで一連の手順を複数回実行できます。
- 条件命令の使用（比較とフラグ）。使用する命令またはサブルーチンを決定します。
- ループの使用。カウンタを使ってある命令セットを指定された回数だけ実行します。
- 間接アドレッシングの使用。同じプログラム命令を使用して異なる変数にアクセスします。

プログラムのルーチン

プログラムは、1 つ以上のルーチンで構成されています。ルーチンは、ある特定の動作をする機能的な単位です。複雑なプログラムでは、タスクの分割とグループ化のために複数のルーチンが必要になります。複数のルーチンにより、プログラムの作成、理解、変更が容易になります。

通常、ルーチンは、ラベルで開始し、プログラム／ルーチンの実行を停止する命令（RTN、STOP など）で終了します。

サブルーチンの呼び出し (XEQ、RTN)

サブルーチンは、他のルーチンから呼び出され（実行され）、サブルーチンが完了したときにそのルーチンに戻るルーチンです。

- 電卓のメモリに 1 つのプログラムだけを保存する場合、ルーチンをさまざまなラベルで区切ることができます。電卓のメモリに複数のプログラムを保存する場合、ルーチンをメインプログラムのラベルの一部とし、特定の行番号で開始します。
- あるサブルーチンから他のサブルーチンを呼び出すこともできます。

この章のフロー図では、次の表記方法を使用します。

A005 GTO B001 → ①	プログラムの実行が、この行から ← ① (「1 から」) が付いた行に分岐します。
B001 LBL B ← ①	プログラムの実行が、→ ① (「1 へ」) が付いた 行からこの行に分岐します。

次の例では、入力した数値の符号を変換するサブルーチンを呼び出します。サブルーチン E はルーチン D の D003 XEQ E001 という行から呼び出され、その数値の符号を変換します。サブルーチン E は RTN 命令で終了し、RTN が（結果を格納し、表示するため）ルーチン D の D004 行にプログラム実行を戻します。フロー図を参照してください。

D001 LBL D		ここから開始します。
D002 INPUT X		
D003 XEQ E001 → ①		サブルーチン E を呼び出します。
D004 STO X ← ②		ここに戻ります。
D005 VIEW X		
D006 RTN		
E001 LBL E ← ①		サブルーチンを開始します。
E002 +/-		数値の符号を変更します。
E003 RTN → ②		ルーチン D に戻ります。

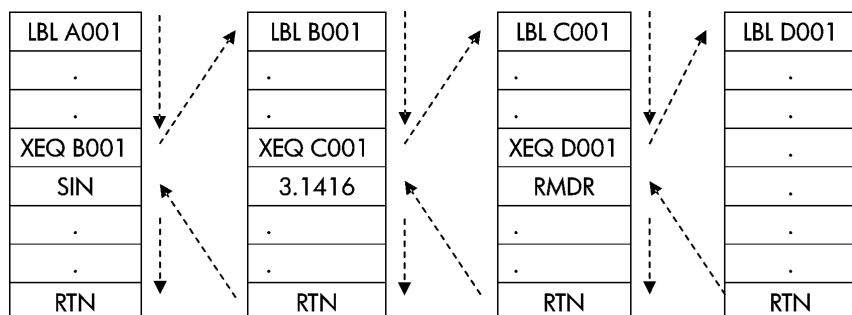
14-2 プログラムのテクニック

サブルーチンの入れ子

サブルーチンから他のサブルーチン呼び出すことができ、さらにそのサブルーチンが他のサブルーチン呼び出せます。このサブルーチンの入れ子（サブルーチン内からの別のサブルーチンの呼び出し）は、最上位のプログラムレベルを除いて 20 レベルの深さまで許容されています。次の図に、入れ子になったサブルーチンを示します

メインプログラム

(最上位)



プログラムの終了

入れ子のレベルが 20 レベルを超えるサブルーチンを実行しようとする、と、**※EQ OVERFLOW** エラーが発生します。

例：サブルーチンの入れ子

次のサブルーチン S は、式の値を計算します。

$$\sqrt{a^2 + b^2 + c^2 + d^2}$$

計算は、より長いプログラムの長い計算の一部です。サブルーチンは、別のサブルーチン Q（サブルーチンの入れ子）を呼び出し、2 乗と加算を反復して計算させます。サブルーチンを使用することで、使用しない場合よりもプログラムが短くなるため、メモリを節約できます。

RPN モードでは、

S001 LBL S		ここでサブルーチンを開始します。
S002 INPUT A		A. を入力します
S003 INPUT B		B. を入力します
S004 INPUT C		C. を入力します
S005 INPUT D		D. を入力します
S006 RCL D		データを呼び出します
S007 RCL C		
S008 RCL B		
S009 RCL A		
S010 x2		A2.
S011 XEQ Q001 → ①		A2 + B2.
② → S012 XEQ Q001 → ③		A2 + B2 + C2
④ → S013 XEQ Q001 → ⑤		A2 + B2 + C2 + D2
⑥ → S014 $\sqrt{\times}$		$\sqrt{A^2+B^2+C^2+D^2}$
S015 RTN		メインルーチンに戻ります。
Q001 LBL Q	← ①③⑤	入れ子になったサブルーチン
Q002 x<>y		
Q003 x2		
Q004 +		x2 を加算します。
②④⑥ ← Q005 RTN		サブルーチン S に戻ります。

分岐（GTO）

サブルーチンの場合と同様に、プログラムの実行を次の行以外の部分に転送することができます。これを分岐と呼びます。

無条件分岐は、GTO（go to）命令を使用して、特定のプログラム行（ラベルと行番号）に分岐します。

プログラムされた GTO 命令

GTO ラベル 命令（**GTO** ラベル行番号を押します）で、動作中のプログラムの実行が指定されたプログラム行に移動します。プログラムは、新しい位置から実行を継続します。自動的に元のポイントに戻ることは**けっしてない**ため、GTO はサブルーチンには使用されません。

例えば、第 16 章の「カーブフィッティング」プログラムを考えます。GTO Z001 命令は、3 つの独立した初期化ルーチンのどれかから LBL Z に実行を分岐します。ラベル Z は、プログラムの心臓部への共通の入力点です。

S001 LBL S		ここから開始できます。
.		
.		
S004 GTO Z001	→①	Z001 に分岐します。
L001 LBL L		ここから開始できます。
.		
.		
L004 GTO Z001	→①	Z001 に分岐します。
E001 LBL E		ここから開始できます。
.		
.		
E004 GTO Z001	→①	Z001 に分岐します。
Z001 LBL Z	←①	ここに分岐します。
.		
.		
.		

キーボードからの GTO の使用

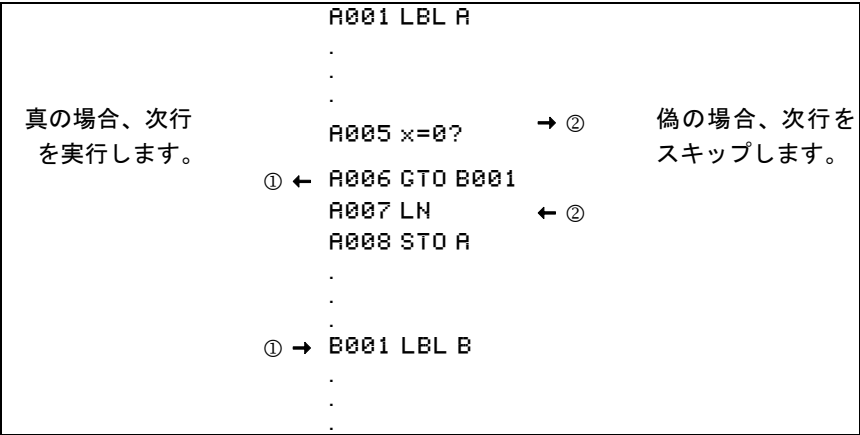
[GTO] を使用すると、プログラムの実行を開始しなくても、プログラムポインタを特定のラベル行番号に移動できます。

- PRGM TOP へ移動：**[GTO]** **[.]** **[.]**.
- 特定の行番号へ移動：**[GTO]** **ラベル行番号**（行番号 < 1000）。
例、**[GTO]** **[.]** **[A]** **[0]** **[0]** **[5]**。例えば、**[GTO]** **[A]** **[0]** **[0]** **[5]**。ディスプレイに "GTO A005" と表示されます。
- ラベルの最初の行に移動する場合。例、A001: **[GTO]** **[A]** **[ENTER]**（押したままにします）。ディスプレイに "GTO A001" と表示されます。

条件命令

プログラム実行の流れを変えるもう 1 つの方法に**条件テスト**があります。これは、2 つの数値を比較し、命題が偽の場合に次のプログラム命令をスキップする合否テストです。

例えば、A005 行の条件命令が $x=0?$ （ x がゼロに等しいかどうか？）であった場合、プログラムが X レジスタの内容とゼロを比較します。X レジスタがゼロだった場合は、そのまま次行に進みます。X レジスタがゼロでなかった場合は、次行をスキップし、A007 行に分岐します。このルールは通常、「Do if true（真なら実行）」と呼ばれます。



上記の例は、条件テストを使った一般的なテクニックを示したものです。テスト直後の行（「真」のときにのみ実行されます）は、他のラベルへの**分岐**です。したがってテストの効力は、特定の条件下で異なるルーチンに分岐することです。

条件命令には、次に示す 3 つのカテゴリがあります。

- 比較テスト。これらは、X レジスタと Y レジスタ、または X レジスタとゼロを比較します。
- フラグテスト。フラグの状態をチェックし、セットまたはクリアを判定します。
- ループカウンタ。通常、特定回数のループに使用します。

比較テスト ($x?y$, $x?0$)

プログラミングには、12 の比較が利用できます。 $x?y$ または  $x?0$ を押すと、2 つのカテゴリのテストのどちらかに対するメニューが表示されます。

- $x?y$ 。x と y の比較テストの場合
- $x?0$ 。x と 0 の比較テストの場合

x は X レジスタに格納された数値を参照し、y は Y レジスタに保存された数値を参照します。これらは、**変数 X と Y を比較しません**。 $x?y$ と $x?0$ を使用して 2 つの数値を比較できます。どちらかが実数でない場合には、エラーメッセージ INVALID DATA が表示されます。

比較のカテゴリを選択したら、メニューキーを押して必要な条件命令を選択します。

テストメニュー

$x?y$	$x?0$
$\neq$ for $x \neq y?$	$\neq$ for $x \neq 0?$
$\leq$ for $x \leq y?$	$\leq$ for $x \leq 0?$
$<$ for $x < y?$	$<$ for $x < 0?$
$>$ for $x > y?$	$>$ for $x > 0?$
$\geq$ for $x \geq y?$	$\geq$ for $x \geq 0?$
$=$ for $x = y?$	$=$ for $x = 0?$

キーボードで条件テストを実行すると、電卓が YES または NO を表示します。

例えば、 $x=2$ 、 $y=7$ のときに、 $x < y$ をテストします。

フラグ

フラグは、状態を示します。フラグは、**セット（真）**と**クリア（偽）**のどちらかです。**フラグテスト**は、もう 1 つの条件テストで、"Do if true" ルールに従います。フラグがセットされていれば次行が実行され、クリアのときは次行をスキップします。

フラグの意味

HP 35s には、0 から 11 の番号が付いた 12 個のフラグがあります。すべてのフラグは、セットまたはクリア状態を持ち、キーボード操作またはプログラム命令でテストできます。12 のフラグのデフォルト状態は、すべて**クリア**です。すべてのフラグをクリアするには、付録 B で説明する 3 つのキーを同時に押すメモリクリア方法を使用します。フラグは、 **CLEAR**  (3ALL)  (Y)  **ENTER** による影響を受けません。

- **フラグ 0、1、2、3、4**には既定の意味は割り当てられていません。したがって、プログラム中で状態の意味を任意に定義できます（次の例を参照）。
- **フラグ 5**をセットすると、プログラムの実行中にオーバーフローが発生した場合にプログラムを中断し、**OVERFLOW** と  を表示します。**オーバーフロー**が発生するのは、プログラムの計算結果が、この電卓で処理可能な数値の最大値を超えたときです。オーバーフロー結果は、最大許容値に置き換わります。フラグ 5 がクリアのときは、オーバーフローが発生してもプログラムが中断されませんが、プログラムが停止したときに **OVERFLOW** が少しの間、表示されます。
- **フラグ 6**は、オーバーフロー **TOO BIG** が発生すると**自動的に**セットされます（手動でセットすることもできます）。フラグ 6 には効力がありませんが、フラグをテストできます。プログラムで 10 進数以外の基数を使用している場合、フラグ 6 は、プログラムで **TOO BIG** が発生したかどうかを判定する場合にもセットされます。

フラグ 5 とフラグ 6 により、プログラム中に発生したオーバーフロー状態を制御できます。フラグ 5 をセットしておく、とオーバーフローの原因となった行の直後の行でプログラムが停止します。また、プログラムの中でフラグ 6 を判定することで、オーバーフローの発生時にプログラムの流れや結果を変更できます。

- **フラグ 7、8、9**は、分数の表示方式を制御します。フラグ 7 は、キーボードからも変更できます。 **FDISP** で分数表示モードのオンとオフを切り替えると、フラグ 7 のセットとクリアが切り替わります。

フラグ 状態	分数制御フラグ		
	7	8	9
クリア (デフォルト)	分数表示モードが オフ。現在の表示 形式で実数を表示。	分数の分母は / c 値以下。	分数を最後まで 約分。
セット	分数表示モードが オン。実数を分数 として表示。	分数の分母は / c 値の約数。	分数を約分しない (フラグ 8 がセッ トされているとき にのみ有効)

- **フラグ 10** は、プログラムによる方程式の実行を制御します：
フラグ 10 がクリア（デフォルト設定）のときは、プログラム実行中に方程式が評価され、計算結果がスタックに格納されます。

フラグ 10 がセットされていると、プログラム実行中に方程式がメッセージとして表示され、VIEW 文と同じ挙動になります。

1. プログラムの実行を停止します。
2. プログラムポインタを次のプログラム行に移動します。
3. 方程式が表示されますが、スタックには影響しません。表示をクリアするには、 または  を押します。その他のキーを押すと、そのキーの機能が実行されます。
4. 次のプログラム行が PSE 命令の場合、1 秒間の停止後にプログラム実行が再開します。

フラグ 10 の状態は、キーボードでは SF または CF 操作の実行、プログラムでは SF 文と CF 文によって制御できます。

- **フラグ 11** は、プログラム内の方程式を実行するときのプロンプトを制御します。このフラグは、キーボードから実行したときに自動で表示されるプロンプトには影響しません。

フラグ 11 がクリア（デフォルト設定）のときは、プログラム内の方程式の評価、SOLVE、積分（ $\int$ FN）が中断なく進みます。方程式で変数が現れるたびに、その変数の現在の値が自動的に呼び出されます。INPUT プロンプトはこのフラグの影響を受けません。

フラグ 11 がセットされていると、方程式に変数が最初に現れたときに、変数に対するプロンプトが表示されます。各変数に対してプロンプトが表示される回数は、方程式に変数が現れる回数に関係なく、1 回だけです。SOLVE では、未知変数についてのプロンプトは表示されません。積分では、被積分変数についてのプロンプトは表示されません。プロンプトが表示されるとプログラム実行が停止します。**[R/S]** を押すと、変数に対して入力した値、または **[R/S]** しか押さなかった場合にはその変数の表示された値（現在の値）を使って、計算を再開します。

フラグ 11 は、プログラムで方程式の評価、SOLVE、JFN が終わると自動的にクリアされます。フラグ 11 の状態も、キーボードでは SF または CF 操作の実行、プログラムでは SF 文と CF 文によって制御できます。

セットフラグのインジケータ

フラグ 0、1、2、3、4 には、対応するフラグがセットされているときにオンとなる、インジケータがあります。ディスプレイ上での **0、1、2、3、4** の表示の有無により、これら 5 つのフラグの状態がセットとクリアのどちらであるかがわかります。ただし、フラグ 5 から 11 には、インジケータがありません。これらのフラグの状態は、キーボードから FS? 命令を実行することで確認できます（次の「フラグの利用」を参照してください）。

フラグの利用

[F1] [FLAGS] を押すと、FLAGS メニュー SF CF FS? が表示されます。

必要な機能を選択すると、フラグ番号（0 から 11）の入力を求められます。

例えば、**[F1] [FLAGS] [1] (1SF) [0]** を押してフラグ 0 をセットします。

[F1] [FLAGS] [1] (1SF) [.] [0] を押してフラグ 10 をセットします。

[F1] [FLAGS] [1] (1SF) [.] [1] を押してフラグ 11 をセットします。

FLAGS メニュー

メニューキー	説明
SF <i>n</i>	Set flag. フラグ <i>n</i> をセットします。
CF <i>n</i>	Clear flag. フラグ <i>n</i> をクリアします。
FS? <i>n</i>	Is flag set? フラグ <i>n</i> の状態をテストします。

フラグテストは、条件テストで、比較テストの場合と同様にプログラムの実行に影響します。FS? n 命令は、所定のフラグがセットされているかどうかを判定します。セットされていれば、プログラムの次行が実行されます。クリアのときは、次行はスキップされます。これは、この章で前述の「条件命令」で示した "Do if True" ルールです。

キーボードでフラグをテストすると、電卓が YES または NO を表示します。プログラムでは、テストする条件を既知の状態から始めます。現在のフラグ設定は、その前に実行したプログラムがフラグ設定をどう残したかに依存します。所定のフラグはクリアで、プログラムがセットしたときにだけセットされると**想定する**のは危険です。フラグをセットする条件が発生する前にフラグをクリアすることで、**明示的に指定**する必要があります。次の例を参照してください。

例：フラグの利用

プログラム行： (RPN モード)

説明：

S001 LBL S	
S002 CF 0	In X のインジケータであるフラグ 0 をクリアします
S003 CF 1	In Y のインジケータであるフラグ 1 をクリアします
S004 INPUT X	X の入力を求め、格納します
S005 FS? 0	フラグ 0 がセットされている場合
S006 LN	... X 入力の自然対数を取り込みます
S007 STO X	フラグテスト後に値を X に格納します
S008 INPUT Y	Y の入力を求め、格納します
S009 FS? 1	フラグ 1 がセットされている場合
S010 LN	... Y 入力の自然対数を取り込みます
S011 STO Y	フラグテスト後に値を Y に格納します
S012 VIEW X	値を表示します
S013 VIEW Y	値を表示します
S014 RTN	
チェックサムと長さ：16B3 42	

S002 CF0 行と S003 CF1 行を（前述のように）作成した場合、フラグ 0 とフラグ 1 がクリアされるため、S006 行と S010 行は X 入力と Y 入力の自然対数を取り込みません。

14-12 プログラムのテクニック

S002 行と S003 行を SF 0 と CF 1 に置き換えた場合、フラグ 0 がセットされるため、S006 行が X 入力の自然対数を取り込みます。

S002 行と S003 行を CF 0 と SF1 に置き換えた場合、フラグ 1 がセットされるため、S010 行が Y 入力の自然対数を取り込みます。

S002 行と S003 行を SF0 と SF1 に置き換えた場合、フラグ 0 とフラグ 1 がセットされるため、S006 行と S010 行は X 入力と Y 入力の自然対数を取り込みます。

プログラムを使用して、フラグの使用方法を確認します

キー：	ディスプレイ：	説明：
(RPN モード)		
XEQ S ENTER	X?	ラベル S を実行し、X 値の入力を求めます
1 R/S	Y?	1 を X に格納し、Y 値の入力を求めます。
1 R/S	X=	1 を X に格納し、フラグテスト
	1.0000	後の値を表示します
R/S	Y=	フラグテスト後の Y 値を表示し
	1.0000	ます

他の 3 つのケースを試みることができます。試みる前に、**◀** **FLAGS** **2**(2CF) **0** と **◀** **FLAGS** **2**(2CF) **1** を押してフラグ 1 と 0 をクリアします。

例：分数表示の制御

次のプログラムで電卓の分数表示機能を使用できます。プロンプトが表示され、入力した分数と分母（/c 値）が使用されます。このプログラムには、3 個の分数表示フラグ（7、8、9）と「メッセージ表示」フラグ（10）の使用例も含まれています。

プログラムでは、メッセージは **MESSAGE** のように示されており、方程式として入力されています。

1. 方程式入力モードを設定するため、**[EQN]** を押します (**EQN** インジケータがオンになります)。
2. メッセージの各英文字には **[RCL]** 文字を押します。スペース文字には **[ SPACE]** を押します。
3. **[ENTER]** を押して、現在のプログラム行にメッセージを挿入し、方程式入力モードを終了します。

プログラム行 : (RPN モード)		説明 :
F001	LBL F	分数プログラムを開始します。
F002	CF 7	3 つの分数フラグをクリアします。
F003	CF 8	
F004	CF 9	
F005	SF 10	メッセージを表示します。
F006	DEC	10 進数を選択します。
F007	INPUT V	数値の入力を求めます。
F008	INPUT D	分母 (2 - 4095) の入力を求めます。
F009	RCL V	メッセージを表示し、10 進数を表示します。
F010	DECIMAL	
F011	PSE	
F012	STOP	
F013	RCL D	
F014	/C	/C 値を設定し、フラグ 7 をセットします。
F015	RCL V	
F016	MOST PRECISE	メッセージを表示し、分数を表示します。
F017	PSE	
F018	STOP	
F019	SF 8	フラグ 8 をセットします。
F020	FACTOR DENOM	メッセージを表示し、分数を表示します。
F021	PSE	
F022	STOP	
F023	SF 9	フラグ 9 をセットします。
F024	FIXED DENOM	メッセージを表示し、分数を表示します。
F025	PSE	
F026	STOP	
F027	GTO F001	プログラムの先頭に移動します。

チェックサムと長さ : BE54 123

14-14 プログラムのテクニック

プログラムを使用して、分数のさまざまな表示形式を確認できます。

キー:	ディスプレイ:	説明:
(RPN モード)		
[XEQ] [F] [ENTER]	V? 値	ラベル F を実行し、分数 (V) の入力を求めます。
[2] [.] [5] [3] [R/S]	D? 値	2.53 を V に格納し、分母 (D) の入力を求めます。
[1] [6] [R/S]	DECIMAL 16.0000 2.5300	16 を /c 値として格納します。メッセージを表示し、10 進数を表示します。
[R/S]	MOST PRECISE 2 8/15 ▼ 2 8/15	分数表示形式 (分母は 16 以下) を示すメッセージと、分数が表示されます。▼ は、分子が「8 より少し小さい」ことを示しています。
[R/S]	FACTOR DENOM 2 1/2 ▲ 2 1/2	分数表示形式 (分母は 16 の約数) を示すメッセージと、分数が表示されます。
[R/S]	FIXED DENOM 2 8/16 ▲	分数表示形式 (分母は 16) を示すメッセージと、分数が表示されます。
[R/S] [C] [←] [FLAGS]	2.5300	プログラムを停止し、フラグ 10 をクリアします。
[2] (2CF) [.] [0]	2.5300	

ループ

後方、つまり前の行への分岐により、プログラムの一部を自動的に複数回実行させることが可能になります。これを**ループ**と呼びます。

```

0001 LBL D
0002 INPUT M
0003 INPUT N
0004 INPUT T
0005 GTO 0001

```

このルーチンは、**無限ループ**の例です。初期データの収集に使用できます。3 個の値を入力後、[XEQ] ラベル行番号を押してこのループを手動で終了し、他のルーチンを実行します。

プログラムでは、メッセージは MESSAGE のように示されており、方程式として入力されています。

1. 方程式入力モードを設定するため、**[EQN]** を押します (**EQN** インジケータがオンになります)。
2. メッセージの各英文字には **[RCL]** 文字を押します。スペース文字には **[ SPACE]** を押します。
3. **[ENTER]** を押して、現在のプログラム行にメッセージを挿入し、方程式入力モードを終了します。

プログラム行 : (RPN モード)		説明 :
F001	LBL F	分数プログラムを開始します。
F002	CF 7	3 つの分数フラグをクリアします。
F003	CF 8	
F004	CF 9	
F005	SF 10	メッセージを表示します。
F006	DEC	10 進数を選択します。
F007	INPUT V	数値の入力を求めます。
F008	INPUT D	分母 (2 - 4095) の入力を求めます。
F009	RCL V	メッセージを表示し、10 進数を表示します。
F010	DECIMAL	
F011	PSE	
F012	STOP	
F013	RCL D	
F014	/C	/C 値を設定し、フラグ 7 をセットします。
F015	RCL V	
F016	MOST PRECISE	メッセージを表示し、分数を表示します。
F017	PSE	
F018	STOP	
F019	SF 8	フラグ 8 をセットします。
F020	FACTOR DENOM	メッセージを表示し、分数を表示します。
F021	PSE	
F022	STOP	
F023	SF 9	フラグ 9 をセットします。
F024	FIXED DENOM	メッセージを表示し、分数を表示します。
F025	PSE	
F026	STOP	
F027	GTO F001	プログラムの先頭に移動します。

チェックサムと長さ : BE54 123

14-16 プログラムのテクニック

プログラムを使用して、分数のさまざまな表示形式を確認できます。

キー:	ディスプレイ:	説明:
(RPN モード)		
[XEQ] [F] [ENTER]	V? 値	ラベル F を実行し、分数 (V) の 入力を求めます。
[2] [.] [5] [3] [R/S]	D? 値	2.53 を V に格納し、分母 (D) の 入力を求めます。
[1] [6] [R/S]	DECIMAL 16.0000 2.5300	16 を /c 値として格納します。 メッセージを表示し、10 進数を 表示します。
[R/S]	MOST PRECISE 2 8/15 ▼ 2 8/15	分数表示形式 (分母は 16 以下) を示すメッセージと、分数が表示 されます。▼ は、分子が「8 より 少し小さい」ことを示しています。
[R/S]	FACTOR DENOM 2 1/2 ▲ 2 1/2	分数表示形式 (分母は 16 の約数) を示すメッセージと、分数が表示 されます。
[R/S]	FIXED DENOM 2 8/16 ▲	分数表示形式 (分母は 16) を 示すメッセージと、分数が表示さ れます。
[R/S] [C] [↵] [FLAGS]	2.5300	プログラムを停止し、フラグ 10
[2] (2CF) [.] [0]	2.5300	をクリアします。

ループ

後方、つまり前の行への分岐により、プログラムの一部を自動的に複数回実行させることが可能になります。これを**ループ**と呼びます。

```

D001 LBL D
D002 INPUT M
D003 INPUT N
D004 INPUT T
D005 GTO D001

```

このルーチンは、**無限ループ**の例です。初期データの収集に使用できます。3 個の値を入力後、**[XEQ]** ラベル行番号を押してこのループを手動で終了し、他のルーチンを実行します。

14-18 プログラムのテクニク

条件ループ (GTO)

特定の条件に適合するまで操作を実行する場合、必要なループの繰り返し回数が不明のときには、条件テストと GTO 命令を持つループを作成できます。

例えば、次のルーチンではループを使用して、得られる A が B 以下になるまで、値 A から定数 B を減算します。

プログラム行： (RPN モード)

説明：

S001 LBL S	
S002 INPUT A	
S003 INPUT B	
S004 RCL A	A を呼び出す方が、スタック内の格納場所を覚えるより簡単です。
S005 RCL - B	A - B を計算します。
S006 STO A	古い A を新しい結果で上書きします。
S007 RCL B	比較のために定数を呼び出します。
S008 < >?	B < 新しい A?
S009 GTO S004	真：ループにより減算を繰り返します。
S010 VIEW A	偽：新しい A を表示します。
S011 RTN	

チェックサムと長さ：2737 33

カウンタによるループ (DSE、ISG)

ループを指定した回数実行したいときは、 **ISG** (増加、より大きければスキップ) または  **DSE** (減少、以下であればスキップ) 条件関数キーを使用します。プログラムでループ機能が実行されるたびに、変数に格納されたカウンタ値が自動的に増加または減少します。次に、現在のカウンタ値と最終のカウンタ値を比較し、結果に基づいてループを継続または終了します。

カウント値を減少させるループには、 **DSE** 変数を使用します。

カウント値を増加させるループには、 **ISG** 変数を使用します。

これらの機能は、BASIC 言語の FOR-NEXT ループに相当します。

FOR 変数 = 初期値 TO 最終値 STEP 増減値

.
. .
.

NEXT 変数

DSE 命令は、増加が負の FOR-NEXT ループに相当します。

ISG または DSE のシフトキー ( ISG) または ( DSE) を押すと、**ループの制御値**（次で説明）を格納する変数の入力が求められます。

ループの制御値

指定された変数には、ループ制御値 $\pm \text{ccccccc}.fffii$ が含まれている必要があります。

- $\pm \text{ccccccc}$ は、現在のカウンタ値（1 から 12 桁）です。値は、ループの実行に伴って**変化**します。
- fff は、最終のカウンタ値（3 桁）です。値は、ループの実行中に**変化しません**。指定されていなければ、 fff の値を 000 とします。
- ii は、増減の間隔です（2 桁または指定無し）。値は、ループの実行中に**変化しません**。指定されていなければ、 ii の値を 01 とします（1 ずつ増加 / 減少）。

ループの制御値が $\text{ccccccc}.fffii$ のとき、DSE が ccccccc を $\text{ccccccc} - ii$ に減少し、新しい ccccccc を fff と比較します。 $\text{ccccccc} \leq fff$ であれば、次のプログラム行をスキップします。

ループの制御値が $\text{ccccccc}.fffii$ のとき、ISG が ccccccc を $\text{ccccccc} + ii$ に増加し、新しい ccccccc を fff と比較します。 $\text{ccccccc} > fff$ であれば、次のプログラム行をスキップします。

① → 現在値 > 最終値 であれば、ルー プを継続します ① ←	<pre> W001 LBL W . . W009 DSE A W010 GTO W001 W011 XEQ X001 . . </pre>	→ ② ← ② 現在値 ≤ 最終値 であれば、ルー プを終了します
① → 現在値 ≤ 最終値 であれば、ルー プを継続します ① ←	<pre> W001 LBL W . . W009 ISG A W010 GTO W001 W011 XEQ X001 . . </pre>	→ ② ← ② 現在値 > 最終値 であれば、ルー プを終了します

例えば、ISG のループの制御値が 0.050 の場合、カウントはゼロから開始し、50 まで、ループごとに 1 ずつ増加します。

ループの制御値が複素数またはベクトルのときは、実数部または最初の部分が、ループの制御に使用されます。

次のプログラムは、ISG を使用して RPN モードで 10 回ループします。ループカウンタ (I.010) は、変数 Z に格納されています。ループカウンタの先頭と末尾のゼロは省略できます。

```

L001 LBL L
L002 1.01
L003 STO Z
L004 ISG Z
L005 GTO L004
L006 RTN

```

[XEQ] **[L]** **[ENTER]** を押し、**[◀]** **[VIEW]** **[Z]** を押して、ループの制御値が 11.0100 であることを確認します。

変数とラベルの間接アドレッシング

間接アドレッシングは、高度なプログラムで使用するテクニックで、変数やラベルをあらかじめ明示的に指定せずに呼び出す方法です。プログラムの実行時に決定されるため、変数やラベルは、プログラムの中間の計算結果（または入力）に依存します。

間接アドレッシングでは 4 個のキー **[I]**、**[O]**、**[J]**、**[U]** を使用します。

これらのキーは、A から Z を変数またはラベルとして取る多数の機能に対してアクティブになります。

- **I** と **J** は、その内容が他の変数を参照できる変数です。他の変数 (A から Z) と同様に数値を保持します。
- **(I)** と **(J)** は、「**I** または **J** に格納された数値を使用してアドレス先の変数またはラベルを決定する」ことを指示するプログラミング機能です。これは**間接アドレス**です (A から Z は**直接アドレス**です)。

[I] と **[O]** を一緒に使用して、1 つの間接アドレスを作成します。**[J]** と **[U]** も同様です。

(I) と **(J)** はそれだけでは、未定義の (**(I)** または **(J)** に数値が保存されていない) 状態か、または未制御の (**I** や **J** にどんな数値が残るかわからない) 状態です。

変数 "I" と "J"

I または **J** の内容に対して、他の変数と同様に格納、呼び出し、操作が行えます。**I**、**J** の解を求めたり、**I** や **J** を積分に使用することもできます。次に、変数 "**i**" を使用できる関数を示します (変数 **J** も同じです)。

STO I	INPUT I	DSE I
RCL I	VIEW I	ISG I
STO +, -, ×, ÷ I	f FN d I	x < > I
RCL +, -, ×, ÷ I	SOLVE I	

間接アドレス (I) と (J)

A から Z を（変数またはラベルとして）使用する多くの関数が、(I) または (J) を使用して、A から Z（変数またはラベル）または統計レジスタを間接的に参照できます。関数 (I) と (J) は、変数 I と J 内の値を使用して、呼び出す変数、ラベル、レジスタを決定します。次の表に方法を示します。

I/J の内容 :	(I)/(J) のアドレス先 :
-1	変数 A または A
.	.
.	.
.	.
-26	変数 Z またはラベル Z
-27	n レジスタ
-28	Σx レジスタ
-29	Σy レジスタ
-30	Σx^2 レジスタ
-31	Σy^2 レジスタ
-32	Σxy レジスタ
0	ここから名前のない間接変数
.	.
.	.
.	.
800	最大アドレスは 800
I<-32 または I>800 または変数が未定義	エラー : INVALID (I)
J<-32 または J>800 または変数が未定義	エラー : INVALID (J)

INPUT(I)、INPUT(J)、VIEW(I)、VIEW(J) 操作は、間接変数またはレジスタの名前を使ってディスプレイにラベルを付けます。

SUMS メニューで、統計レジスタの値を呼び出すことができます。ただし、STO、VIEW、INPUT などの他の操作を実行するには、間接アドレスを使用する必要があります。

アドレスとして (I) または (J) を使用できる関数を示します。FN= の場合、(I) または (J) はラベルを参照します。その他の関数の場合、(I) または (J) は変数かレジスタを参照します。

STO(I)/(J)

RCL(I)/(J)

STO +, -, ×, ÷, (I)/(J)

RCL +, -, ×, ÷, (I)/(J)

X<>(I)/(J)

FN=(I)/(J)

INPUT(I)/(J)

VIEW(I)/(J)

DSE(I)/(J)

ISG(I)/(J)

SOLVE(I)/(J)

∫ FN d(I)/(J)

名前のない変数と統計レジスタは、解を求めたり、積分することができません。

(I)/(J) によるプログラムの制御

I の内容は、プログラム実行のたびに（同じプログラムの別の部分でも）変化する可能性があるため、STO (I)、STO (J) などのプログラム命令は、値を異なる時間に異なる変数に格納できます。例えば、STO(- 1) は、値を変数 A に格納するよう指示します。この機能により、必要な変数またはプログラムラベルをプログラムの実行時まで未定義の状態に保つことができ、プログラムの柔軟性を確保できます。

間接アドレッシングは、ループのカウントや制御に非常に便利です。変数 I または J は、インデックスとして利用され、DSE 関数と ISG 関数のループ制御番号を格納する変数のアドレスを保持します。

(I)/(J) を使った方程式

方程式の (I) または (J) を使用して、変数を間接的に指定できます。〈I〉または〈J〉は、変数 I または J 内の数値によって指定された変数を意味します（間接参照）。I または J および (I) または (J) キーではなくユーザ括弧を使用した〈I〉または〈J〉は、変数 I または J を意味します。

名前のない間接変数

変数 I または J に正の数値を格納すると、最大 801 個の間接変数にアクセスできます。次の例で使用方法を示します。

14-24 プログラムのテクニック

**プログラム行 :
(RPN モード)**

説明 :

A001 LBL A

A002 100

A003 STO I

A004 12345

A005 STO (I) 保存アドレス範囲 "0-100" を定義し、"12345" をアドレス 100 に保存します。

A006 150

A007 STO I

A008 67890

A009 STO (I) "67890" をアドレス 150 に保存します。定義された間接保存範囲は "0-150" です。

A010 100

A011 STO I

A012 0

A013 STO (I) 0 を間接レジスタ 100 に格納します。定義された範囲は "0-150" のままです。

A014 170

A015 STO I

A016 RCL(I) アドレス "170" は未定義なので "INVALID (I)" と表示されます

A017 RTN

注 :

1. 未定義の保存アドレスから値を呼び出す場合、エラーメッセージ "INVALID (I)" が表示されます (A014 を参照してください)。
2. 電卓は、変数 0 から最後のゼロでない変数までのメモリ領域を割り当てます。レジスタの使用後は、メモリを解放するために変数にゼロを格納することが重要です。割り当てられた間接レジスタはそれぞれ、プログラムメモリの 37 バイトを使用します。
3. 最大 800 の変数があります。

プログラムの SOLVE と積分

プログラムの SOLVE

第 7 章では、方程式を入力する方法（方程式が方程式リストに追加されます）と、その方程式をある変数について解く方法について説明しました。ある関数を計算するプログラムを入力し、それをある変数について解くこともできます。これは、解いている方程式が特定の条件で変化する場合や、繰り返し計算を必要とする場合に特に便利です。

プログラムされた関数を解くには：

1. 関数を定義するプログラムを入力します（「SOLVE のためのプログラムを書くには」を参照してください）。
2. SOLVE の対象のプログラムを選択します。  **FN=** ラベルを押します（同じプログラムを解いている場合、このステップをスキップできます）。
3. 未知の変数を求めます。  **SOLVE** 変数を押します。

プログラムされた関数を解くには FN= が必要ですが、方程式リストの方程式を解いている場合は不要です。

計算を中断するには、**C** または **R/S** を押します。メッセージ INTERRUPTED が第 2 行に表示されます。現在の最も良い根の推定値が未知変数に存在します。 **VIEW** を使用すると、スタックを障害せずにこの推定値を確認できます。計算を再開するには、**R/S** を押します。

SOLVE のためのプログラムを書くには：

プログラムでは、方程式と ALG または RPN 操作を最も適切な組合せで使用できます。

1. プログラムをラベルから開始します。このラベルは、SOLVE が評価する関数を識別します (FN= ラベル)。
2. 未知数など、各変数の INPUT 命令を含めます。INPUT 命令により、多変数の関数でも変数を解くことができます。**未知の変数**の INPUT は、電卓によって無視されます。したがって、**すべての変数** (未知の変数を含む) の **個別**の INPUT 命令を含むプログラムを 1 つだけ作成する必要があります。

INPUT 命令を定義しないと、変数に格納された値または式プロンプトで入力された値が使用されます。

3. 関数を評価する命令を入力します
 - 複数行の RPN または ALG シーケンスとしてプログラムされている関数は、解でゼロになる式の形式をとる必要があります。方程式が $f(x) = g(x)$ のとき、プログラムでは $f(x) - g(x)$ を計算します。"=0" が暗に含まれています。
 - 単一の方程式としてプログラムされた関数では、等価式、代入式、式など、方程式の種類に制限はありません。方程式はプログラムで評価され、解で値がゼロになります。INPUT 命令を定義する代わりに、方程式が変数の入力を求めるプロンプトを表示する場合、フラグ 11 をセットしてください。
4. プログラムを RTN で終了します。プログラムの実行は、関数の値を X レジスタに格納した状態で終了する必要があります。

例：ALG を使用したプログラム

「理想気体の法則」の方程式の未知数を解く、ALG 操作を使用したプログラムを作成します。方程式は次の通りです。

$$P \times V = N \times R \times T$$

ここで、

P = 圧力 (気圧または N/m²)

V = 体積 (リットル)

N = 気体のモル数

R = 気体定数

(0.0821 liter-atm/mole-K or 8.314 J/mole-K)

T = 温度 (ケルビン ; K = °C + 273.1)

15-2 プログラムの SOLVE と積分

まず、プログラムモードに設定し、必要に応じてプログラムポインタをメモリの先頭に移動します。

キー：
(ALG モード)

ディスプレイ：

説明：

 PRGM

プログラムモードに設定します。

PRGM TOP

プログラムを入力します。

プログラム行：
(ALG モード)

説明：

G001 LBL G

プログラムされた関数を識別します。

G002 INPUT P

圧力の場合 P を格納します

G003 INPUT V

体積の場合 V を格納します

G004 INPUT N

気体のモル数の場合 N を格納します

G005 INPUT R

気体定数の場合 R を格納します

G006 INPUT T

温度の場合 T を格納します

G007 $P \times V = N \times R \times T$

 を押します。

圧力 $\times$ 体積 = モル数 $\times$ 気体定数 $\times$ 温度

G008 RTN

プログラムを終了します。

チェックサムと長さ：F425 33

 を押して、プログラム入力モードをキャンセルします。

プログラム "G" を使用して、2 リットルの容器に 24°C で保存されている 0.005 モルの二酸化炭素の圧力を求めます。

キー：
(ALG モード)

ディスプレイ：

説明：

 FN= 

プログラム "G" を選択します。

SOLVE は評価により、未知の変数の値を求めます。

 SOLVE 

V?

P を選択し、 V の入力を求めます。

値

N?

2 を V に格納し、 N の入力を求めます。

値

[.] [0] [0] [5] [R/S]

[.] [0] [8] [2] [1]

[R/S]

[2] [4] [+][2] [7]

[3] [.] [1] [ENTER]

[R/S]

R?

値

T?

値

T?

297.1000

SOLVING

P=

0.0610

.005 を N に格納し、 R の入力をお願いします。

.0821 を R に格納し、 T の入力をお願いします。
 T を計算します。

297.1 を T に格納し、 P の解を求めます。圧力は 0.0610 気圧です。

例：方程式を使用したプログラム

方程式を使用して「理想気体の法則」を解くプログラムを作成します。

キー：
(RPN モード)

[P] [PRGM]

[GTO] [.] [.]

[P] [LBL] H

[F] [FLAGS] [1]

(1SF) [.] [1]

[EQN]

[RCL] [P] [X]

[RCL] [V] [F] [=]

[RCL] [N] [X]

[RCL] [R] [X]

[RCL] [T] [ENTER]

[F] [RTN]

[C]

ディスプレイ：

PRGM TOP

H001 LBL H

H002 SF 11

H003 $P \times V = N \times R \times T$

H004 RTN

0.0610

説明：

プログラム入力モードを選択します。プログラムポインタをプログラムの先頭に移動します。プログラムにラベルを付けます。方程式のプロンプトを有効にします。方程式を評価し、フラグ 11 をクリアします。(チェックサムと長さ：EDC8 9)

プログラムを終了します。プログラム入力モードをキャンセルします。

プログラムのチェックサムと長さ DF52 21

前回の例から二酸化炭素の温度が 10 °C 下がったときの圧力の変化を計算します。

キー:	ディスプレイ:	説明:
(RPN モード)		
$\rightarrow$ STO $\downarrow$ L	0.0610	前の圧力を格納します。
$\leftarrow$ FN= $\downarrow$ H	0.0610	プログラム "H" を選択します。
$\rightarrow$ SOLVE $\downarrow$ P	V?	変数 P を選択し、V の入力を求め
	2.0000	ます。
$\downarrow$ R/S	N?	2 を V に保持し、N の入力を求め
	0.0050	ます。
$\downarrow$ R/S	R?	.005 を N に保持し、R の入力を求
	0.0821	めます。
$\downarrow$ R/S	T?	.0821 を R に保持し、T の入力を
	297.1000	求めます。
$\downarrow$ ENTER 1 0 $\downarrow$ -	T?	新しい T を計算します。
	287.1000	
$\downarrow$ R/S	SOLVING	287.1 を T に格納し、新しい P の
	P=	解を求めます。
	0.0589	
$\downarrow$ RCL $\downarrow$ L $\downarrow$ -	-0.0021	温度が 297.1 K から 287.1 K に変
		化するときの、気体の圧力の変化
		を計算します (負の結果は圧力の
		低下を示します)。

プログラムでの SOLVE の使用

SOLVE 操作をプログラムの一部として使用できます。

適切な場合、SOLVE **変数**命令を実行する前に初期推定値を指定するか、初期推定値のプロンプトを定義し、未知変数と X レジスタに格納します。方程式を解いて未知の変数を求めるための 2 つの命令が、プログラムに次のように示されます。

FN= ラベル

SOLVE 変数

プログラム中の SOLVE 命令では、ラベル付きの表示 (**変数 = 値**) が生成されません。プログラムの有意な出力でない場合があるからです (表示する前にこの数値を使ってさらに計算を実行する必要がある場合など)。この結果を表示する場合、VIEW **変数**命令を SOLVE 命令の後に追加します。

未知変数について解が見つからないときは、(第 14 章で説明した "Do If True" ルールに従って) 次のプログラム行がスキップされます。プログラムは、新しい初期推定値の選択、入力値の変更などにより、根が見つからない問題进行处理する必要があります。

例：プログラムでの SOLVE

次に、**[XEQ]** X または Y を押すことで、 x または y を求めることができるプログラムの抜粋を示します。

プログラム行： (RPN モード)

```
X001 LBL X
X002 24
X003 GTO L001
チェックサムと長さ：62A0 11
Y001 LBL Y
Y002 25
Y003 GTO L001
チェックサムと長さ：221E 11
L001 LBL L
L002 STO I
L003 FN= F
L004 SOLVE(I)
L005 VIEW(I)
L006 RTN
チェックサムと長さ：D45B 18
F001 LBL F
,
,
,
F010 RTN
```

説明：

X の設定。
X のインデックス。
メインルーチンに分岐します。

Y の設定。
Y のインデックス。
メインルーチンに分岐します。

メインルーチン。
インデックスを I に格納します。
解を求めるようプログラムを定義します。
該当する変数の解を求めます。
解を表示します。
プログラムを終了します。

$f(x,y)$ を計算します。必要に応じて INPUT
または方程式プロンプトを定義します。

プログラムの積分

第 8 章では、方程式（または式）を入力する方法（式が方程式リストに追加されます）と、その方程式をある変数について積分する方法について説明しました。ある関数を計算する**プログラム**を入力し、**それをある変数について積分**することもできます。これは、積分している方程式が特定の条件で変化する場合や、繰り返し計算を必要とする場合に特に便利です。

プログラムされた関数を積分するには：

1. 被積分関数を定義するプログラムを入力します（下の「**FN のためのプログラムを書くには**」を参照してください）。
2. 被積分関数を定義するプログラムを選択します。 **FN=** ラベルを押します（同じプログラムを再度積分している場合、このステップをスキップできます）。
3. 積分の範囲を入力します。**下限**を入力し、 を押して、**上限**を入力します。
4. 積分の変数を選択し、計算を開始します。  **変数**、を押します。

プログラムされた関数を積分するには **FN=** が必要ですが、方程式リストの方程式を積分している場合は不要です。

 または  を押すと積分の計算を中止できます。メッセージ **INTERRUPTED** が第 2 行に表示されます。しかし計算を再開することはできません。計算が正常に終了するまで、積分に関する情報は得られません。

積分計算の実行中に  を押すと、 **FN=** 操作がキャンセルされます。この場合は、最初から  **FN=** をやり直してください。

「FN のためのプログラムを書くには :

プログラムでは、方程式、ALG または RPN 操作を最も適切な組合せで使用できます。

1. プログラムをラベルから開始します。このラベルは、積分する関数を識別します (FN= ラベル)。
2. 積分変数など、各変数の INPUT 命令を定義します。INPUT 命令により、多変数の関数でも変数を積分することができます。積分変数の INPUT は、電卓によって無視されます。したがって、**すべての変数** (積分変数を含む) の**個別**の INPUT 命令を含むプログラムを 1 つだけ作成する必要があります。

INPUT 命令を定義しないと、変数に格納された値または式プロンプトで入力された値が使用されます。

3. 関数を評価する命令を入力します。
 - 複数行の RPN または ALG シーケンスとしてプログラムされている関数は、積分する関数値を計算する必要があります。
 - 単一の方程式としてプログラムされた関数は通常、被積分関数を指定する式として定義されます。方程式の種類は制限されません。INPUT 命令を含める代わりに方程式が変数の入力を求めるプロンプトを表示する場合、フラグ 11 をセットしてください。
4. プログラムを RTN で終了します。プログラムの実行は、関数の値を X レジスタに格納した状態で終了する必要があります。

例 : 方程式を使用したプログラム

第 8 章の例は、次のような正弦積分関数でした。

$$Si(t) = \int_0^t \left(\frac{\sin x}{x} \right) dx$$

この関数を評価するには、被積分関数を定義するプログラムを積分します。

S001 LBL S 関数を定義します。
S002 SIN(X)÷X 関数を式として定義します（チェックサムと長さ：
 OEE0 8）。
S003 RTN サブルーチンを終了します。
プログラムのチェックサムと長さ D57E 17

プログラムを入力し、正弦積分関数を x について区間 0 ~ 2 ($t=2$) で積分します。

キー：	ディスプレイ：	説明：
(RPN モード)		
(2RAD)		ラジアンモードを選択します。
		ラベル S を被積分関数として選択します。
	2_	積分の下限値と上限値を入力します。
	INTEGRATING	関数を区間 0 から 2 で積分し、結果を表示します。
	∫ =	
	1.6054	
(1DEG)	1.6054	度モードに戻します。

プログラムでの積分の使用

積分をプログラムから実行できます。積分の前に必ず、区間を指定するか、そのプロンプトを定義しておいてください。確度と実行時間が、プログラムの実行時におけるディスプレイの表示形式に依存する点にも注意してください。2 つの積分命令が、プログラムに次のように示されます。

FN= ラベル

∫ FN 変数

プログラム中の ∫ FN 命令では、ラベル付きの表示 (∫ = 値) が生成されません。プログラムの有意な出力でない場合があるからです（表示する前にこの数値を使ってさらに計算を実行する必要がある場合など）。この結果を表示するときは、X レジスタの結果を表示するため、∫ FN 命令の後に PSE () または STOP () 命令を追加します。

15-10 プログラムの SOLVE と積分

積分または SOLVE の反復実行中、PSE 命令が表示された方程式（フラグ 10 がセットされている状態）の直後に存在する場合、その方程式が 1 秒間表示され、各反復の最後まで実行が継続されます。方程式の表示中、スクロールやキーボード入力はできなくなります。

例：プログラムでの「FN.

第 16 章のプログラム「正規分布と逆正規分布」では、正規密度関数の方程式の積分を使用しています。

$$\frac{1}{S\sqrt{2\pi}} \int_M^D e^{-\left(\frac{D-M}{S}\right)^2/2} dD.$$

$e^{((D-M)/S)^2/2}$ 関数は、ラベル F のルーチンで計算しています。その他のルーチンでは既知の値に対するプロンプト、 $Q(D)$ の計算、正規分布曲線の上側確率の計算が行われます。積分は、ラベル Q のルーチンで設定され、実行されます。

Q001 LBL Q

Q002 RCL M 積分の下限值を呼び出します。

Q003 RCL X 積分の上限値を呼び出します。(X = D)

Q004 FN= F 関数を指定します。

Q005 ? FN d D ダミー変数 D. を使って正規関数を積分します。

SOLVE と積分の制約

SOLVE 変数 と JFN d 変数 命令は、別の SOLVE または JFN 命令を含むルーチン
を呼び出せません。つまり、これらの命令を再帰的に使用できません。例えば、
多重積分を計算しようとする、J<JFN> エラーが発生します。また、SOLVE と
JFN は、FN= ラベル 命令を含むルーチンを呼び出せません。呼び出そうとする
と、SOLVE ACTIVE または JFN ACTIVE エラーになります。SOLVE は、JFN
命令を含むルーチンを呼び出せません (SOLVE<JFN> エラーが発生します)。
同様に、JFN は、SOLVE 命令を含むルーチンを呼び出せません (J<SOLVE>
エラーが発生します)。

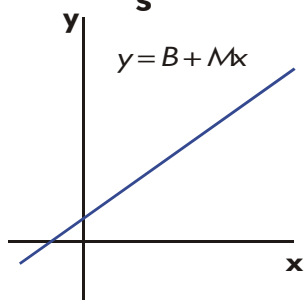
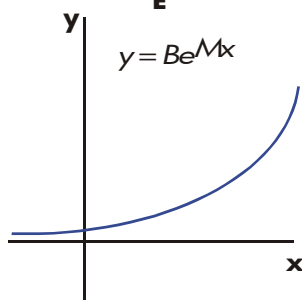
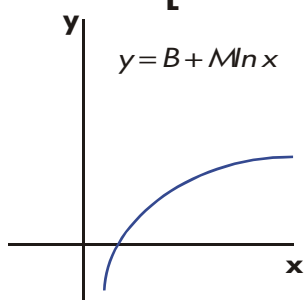
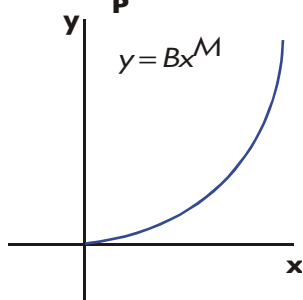
プログラム内の SOLVE 変数 と JFN d 変数 命令は、電卓内の 20 個の未決のサ
ブルーチン戻り値のどれかを使用します (第 14 章の「サブルーチンの入れ子」
を参照してください)。

統計のプログラム

カーブフィッティング

このプログラムにより、4 つの方程式モデルのどれかをデータに適合できます。モデルとして、直線、対数曲線、指数曲線、累乗曲線があります。プログラムは、複数の (x, y) データペアを受け入れ、相関係数 r 、および 2 つの回帰係数 m と b を計算します。プログラムには、予測値 $\hat{x}$ と $\hat{y}$ を計算するルーチンもあります（予測値の定義については、第 12 章「線形回帰」を参照してください）。

次の図に、曲線と関連する方程式の例を示します。HP 35s に内蔵の回帰関数を使用して、回帰係数を計算します。

Straight Line Fit**S****Exponential Curve Fit****E****Logarithmic Curve Fit****L****Power Curve Fit****P**

対数曲線を適合するには、 x の値が正である必要があります。指数曲線を適合するには、 y の値が正である必要があります。累乗曲線を適合するには、 x と y の両方が正である必要があります。これらの場合に負の数値を入力すると、LOG(NEG) エラーが発生します。

非常に大きなデータ値であるにもかかわらず、データ値間の差が比較的小さい場合、データ値の大きさが大きく異なる場合と同様、精度問題が発生します。第 12 章「データの精度に関する制限」を参照してください。

プログラム :

プログラム行 : (RPN モード)

説明

S001 LBL S	このルーチンは直線モデル用の状態を設定します。
S002 CF 0	In X のインジケータであるフラグ 0 をクリアします。
S003 CF 1	In Y のインジケータであるフラグ 1 をクリアします。
S004 GTO Z001	共通の入力ポイント Z に分岐します。
チェックサムと長さ : 8E85 12	
L001 LBL L	このルーチンは対数モデル用の状態を設定します。
L002 SF 0	In X のインジケータであるフラグ 0 をセットします。
L003 CF 1	In Y のインジケータであるフラグ 1 をクリアします。
L004 GTO Z001	共通の入力ポイント Z に分岐します。
チェックサムと長さ : AD1B 12	
E001 LBL E	このルーチンは指数モデル用の状態を設定します。
E002 CF 0	In X のインジケータであるフラグ 0 をクリアします。
E003 SF 1	In Y のインジケータであるフラグ 1 をセットします。
E004 GTO Z001	共通の入力ポイント Z に分岐します。
チェックサムと長さ : D6F1 12	
P001 LBL P	このルーチンは累乗モデル用の状態を設定します。
P002 SF 0	In X のインジケータであるフラグ 0 をセットします。
P003 SF 1	In Y のインジケータであるフラグ 1 をセットします。
チェックサムと長さ : 3800 9	
Z001 LBL Z	すべてのモデルの共通入力ポイントを定義します。
Z002 CLΣ	統計レジスタをクリアします。  CLEAR  (4Σ) を押します。
Z003 0	最初の入力のため、ループカウンタをゼロに設定します。
チェックサムと長さ : 8611 10	
W001 LBL W	入力ループの開始を定義します。
W002 1	入力プロンプトのため、ループカウンタを 1 ずつ調整します。
W003 +	
W004 STO X	X のプロンプトに表示させるため、ループカウンタを X に格納します。
W005 INPUT X	カウンタをプロンプト付きで表示し、X 入力を格納します。

**プログラム行 :
(RPN モード)**

説明

W006 FS? 0	フラグ 0 がセットされている場合...
W007 LN	... X 入力 of 自然対数を取り込みます。
W008 STO B	補正ルーチンで使用するため、値を格納します。
W009 INPUT Y	Y の入力を求め、格納します。
W010 FS? 1	フラグ 1 がセットされている場合...
W011 LN	... Y 入力 of 自然対数を取り込みます。
W012 STO R	
W013 RCL B	
W014 $\Sigma+$	B と R を x,y データペアとして統計レジスタに累積します。
W015 GTO W001	ループして別の X, Y ペアを処理します。

チェックサムと長さ : 9560 46

U001 LBL U	"undo" ルーチンの開始を定義します。
U002 RCL R	最新のデータペアを呼び出します。
U003 RCL B	
U004 $\Sigma-$	このペアを統計の累積から削除します。
U005 GTO W001	ループして別の X, Y ペアを処理します。

チェックサムと長さ : A79F 15

R001 LBL R	出力ルーチンの開始を定義します。
R002 r	相関係数を計算します。
R003 STO R	それを R に格納します
R004 VIEW R	相関係数を表示します。
R005 b	係数 b を計算します。
R006 FS? 1	フラグ 1 がセットされている場合、b の自然真数を取り込みます。

R007 eX	
R008 STO B	b を B に格納します。
R009 VIEW B	値を表示します。
R010 m	係数 m を計算します。
R011 STO M	m を M に格納します。
R012 VIEW M	値を表示します。

チェックサムと長さ : 850C 36

Y001 LBL Y	予測値 (見積もり) ループの開始を定義します。
------------	--------------------------

16-4 統計のプログラム

**プログラム行 :
(RPN モード)**

説明

Y002 INPUT X	x 値を表示し、入力を求め、変更された場合は値を X に格納します。
Y003 FS?0	フラグ 0 がセットされている場合...
Y004 GTO K001	K001 に分岐します。
Y005 GTO M001	M001 に分岐します。
Y006 STO Y	$\hat{y}$ 値を Y に格納します。
Y007 INPUT Y	y 値を表示し、入力を求め、変更された場合は値を Y に格納します。
Y008 FS?0	フラグ 0 がセットされている場合...
Y009 GTO O001	O001 に分岐します
Y010 GTO N001	N001 に分岐します
Y011 STO X	次のループのため $\hat{x}$ を X に格納します。
Y012 GTO Y001	ループして別の予測値を処理します。

チェックサムと長さ : C3B7 36

A001 LBL A	このサブルーチンは、直線モデルの $\hat{y}$ を計算します。
A002 RCL M	
A003 RCL× X	
A004 RCL+ B	$\hat{y} = MX + B$ を計算します。
A005 RTN	呼び出したルーチンに戻ります。

チェックサムと長さ : 9688 15

G001 LBL G	このサブルーチンは、直線モデルの $\hat{x}$ を計算します。
G002 RCL Y	
G003 RCL- B	
G004 RCL÷ M	$\hat{x} = (Y - B) \div M$ を計算します。
G005 RTN	呼び出したルーチンに戻ります。

チェックサムと長さ : 9C0F 15

B001 LBL B	このサブルーチンは、対数モデルの $\hat{y}$ を計算します。
B002 RCL X	
B003 LN	
B004 RCL× M	
B005 RCL+ B	$\hat{y} = M \ln X + B$ を計算します。
B006 RTN	呼び出したルーチンに戻ります。

プログラム行 :
(RPN モード)

説明

チェックサムと長さ : 889C 18

H001 LBL H	このサブルーチンは、対数モデルの $\hat{x}$ を計算します。
H002 RCL Y	
H003 RCL - B	
H004 RCL ÷ M	
H005 eX	$\hat{x} = e(Y - B) \div M$ を計算します
H006 RTN	呼び出したルーチンに戻ります。

チェックサムと長さ : 0DBE 18

C001 LBL C	このサブルーチンは、指数モデルの $\hat{y}$ を計算します。
C002 RCL M	
C003 RCL × X	
C004 eX	
C005 RCL × B	$\hat{y} = Be^{MX}$ を計算します。
C006 GTO M005	M005 に分岐します

チェックサムと長さ : 9327 18

I001 LBL I	このサブルーチンは、指数モデルの $\hat{x}$ を計算します。
I002 RCL Y	
I003 RCL ÷ B	
I004 LN	
I005 RCL ÷ M	$\hat{x} = (\ln(Y \div B)) \div M$ を計算します。
I006 GTO N005	N005 に移動します

チェックサムと長さ : 7219 18

D001 LBL D	このサブルーチンは、累乗モデルの $\hat{y}$ を計算します。
D002 RCL X	
D003 RCL M	
D004 √X	
D005 RCL × B	$Y = B(X^M)$ を計算します。
D006 GTO K005	K005 に移動します

チェックサムと長さ : 11B3 18

J001 LBL J	このサブルーチンは、累乗モデルの $\hat{x}$ を計算します。
------------	------------------------------------

16-6 統計のプログラム

**プログラム行 :
(RPN モード)**

説明

J002 RCL Y

J003 RCL ÷ B

J004 RCL M

J005 1/x

J006 yX $\hat{x} = (Y/B)^{1/M}$ を計算します。

J007 GTO 0005 O005 に移動します

チェックサムと長さ : 8524 21

K001 LBL K D001 と B001 のどちらを実行するかを決定します

K002 FS?1 フラグ 1 がセットされている場合 ...

K003 XEQ D001 D001 を実行します

K004 XEQ B001 B001 を実行します

K005 GTO Y006 Y006 に移動します

チェックサムと長さ : 4BFA 15

M001 LBL M C001 と A001 のどちらを実行するかを決定します

M002 FS?1 フラグ 1 がセットされている場合 ...

M003 XEQ C001 C001 を実行します

M004 XEQ A001 A001 を実行します

M005 GTO Y006 Y006 に移動します

チェックサムと長さ : 1C4D 15

O001 LBL O J001 と H001 のどちらを実行するかを決定します

O002 FS?1 フラグ 1 がセットされている場合 ...

O003 XEQ J001 J001 を実行します

O004 XEQ H001 H001 を実行します

O005 GTO Y011 Y011 に移動します

チェックサムと長さ : 0AA5 15

N001 LBL N I001 と G001 のどちらを実行するかを決定します

N002 FS?1 フラグ 1 がセットされている場合 ...

N003 XEQ I001 I001 を実行します

N004 XEQ G001 G001 を実行します

N005 GTO Y011 Y011 に移動します

チェックサムと長さ : 666D 15

フラグの使用：

X 入力に自然対数が必要な場合、フラグ 0 がセットされます。Y 入力に自然対数が必要な場合、フラグ 1 がセットされます。

フラグ 1 がルーチン N でセットされている場合、I001 が実行されます。フラグ 1 がクリアの場合、G001 が実行されます。

プログラム利用手順：

1. プログラムルーチンを入力します。終了したら **[C]** を押します。
2. **[XEQ]** を押し、次を押して適合させる曲線の種類を選択します。
 - 直線の場合 **[S]** **[ENTER]**
 - 対数曲線の場合 **[L]** **[ENTER]**
 - 指数曲線の場合 **[E]** **[ENTER]**
 - 累乗曲線の場合 **[P]** **[ENTER]**
3. x 値を入力し、**[R/S]** を押します。
4. y 値を入力し、**[R/S]** を押します。
5. データペアに対してステップ 3 と 4 を繰り返します。ステップ 3 で **[R/S]** を押した後にミスを発見した場合 (Y? 値プロンプトがまだ表示されている場合)、**[R/S]** を再度押し (X? 値プロンプトを表示)、**[XEQ]** **[U]** **[ENTER]** を押して最後のデータペアを元に戻します (削除します)。ステップ 4 の後にミスを発見した場合、**[XEQ]** **[U]** **[ENTER]** を押します。どちらの場合も、ステップ 3 から続行します。
6. すべてのデータを入力したら、**[XEQ]** **[R]** **[ENTER]** を押して、相関係数 R を確認します。
7. **[R/S]** を押して、回帰係数 B を確認します。
8. **[R/S]** を押して、回帰係数 M を確認します。
9. **[R/S]** を押して、 $\hat{x}$, $\hat{y}$ 予測ルーチンに対する X? 値プロンプトを確認します。
10. x に基づいて $\hat{y}$ を予測する場合、X? 値プロンプトで x を入力し、**[R/S]** を押して、 $\hat{y}$ (Y?) を確認します。
11. y に基づいて $\hat{x}$ を予測する場合、Y? 値プロンプトが表示されるまで **[R/S]** を押し、y を入力してから **[R/S]** を押して、 $\hat{x}$ (X?) を確認します。
12. 予測値を更に計算するには、ステップ 10 または 11 を繰り返します。
13. 新しい事例の場合は、ステップ 2 へ移動します。

16-8 統計のプログラム

変数の使用：

B	回帰係数（直線の y 切片）。スクラッチにも使用されます。
M	回帰係数（直線の傾き）。
R	相関係数。スクラッチにも使用されます。
X	データ入力的时候は、データペアの x 値。 $\hat{y}$ を予測するときは、仮説 x 。仮説 y が与えられたときは、 $\hat{x}$ (x 予測値)。
Y	データ入力的时候は、データペアの y 値。 $\hat{x}$ を予測するときは、仮説 y 。仮説 x が与えられたときは、 $\hat{y}$ (y 予測値)。
統計レジスタ	統計の累積と計算。

例 1：

直線を次のデータに適合させます。3 番目のデータペアで入力ミスがあったとして、これを Undo ルーチンで修正します。 x 値 37 に対する y を予測します。 y 値 101 に対する x を予測します。

X	40.5	38.6	37.9	36.2	35.1	34.6
Y	104.5	102	100	97.5	95.5	94

キー：
(RPN モード)

XEQ **S** **ENTER**

4 **0** **.** **5** **R/S**

1 **0** **4** **.** **5**

R/S

3 **8** **.** **6** **R/S**

1 **0** **2** **R/S**

ディスプレイ：

$X?$

1.0000

$Y?$

値

$X?$

2.0000

$Y?$

104.5000

$X?$

3.0000

説明：

直線ルーチンを開始します。

データペアの x 値を入力します。

データペアの y 値を入力します。

データペアの x 値を入力します。

データペアの y 値を入力します。

37.9 ではなく 379 を入力してしまった場合の修正方法を示します。

キー:	ディスプレイ:	説明:
(RPN モード)		
3 7 9 R/S	Y? 102.0000	データペアに対して間違った x 値を入力します。
R/S	X? 4.0000	X? プロンプトを取得します。
XEQ U ENTER	X? 3.0000	最後のペアを削除します。正しいデータ入力に移ります。
3 7 . 9 R/S	Y? 102.0000	データペアに対して正しい x 値を入力します。
1 0 0 R/S	X? 4.0000	データペアの y 値を入力します。
3 6 . 2 R/S	Y? 100.0000	データペアの x 値を入力します。
9 7 . 5 R/S	X? 5.0000	データペアの y 値を入力します。
3 5 . 1 R/S	Y? 97.5000	データペアの x 値を入力します。
9 5 . 5 R/S	X? 6.0000	データペアの y 値を入力します。
3 4 . 6 R/S	Y? 95.5000	データペアの x 値を入力します。
9 4 R/S	X? 7.0000	データペアの y 値を入力します。
XEQ R ENTER	R= 0.9955	相関係数を計算します。
R/S	B= 33.5271	回帰係数 B を計算します。
R/S	M= 1.7601	回帰係数 M を計算します。
R/S	X? 7.0000	仮説 x 値を入力するよう求めます。
3 7 R/S	Y? 98.6526	37 を X に格納し、 $\hat{y}$ を計算します。
1 0 1 R/S	X? 38.3336	101 を Y に格納し、 $\hat{x}$ を計算します。

16-10 統計のプログラム

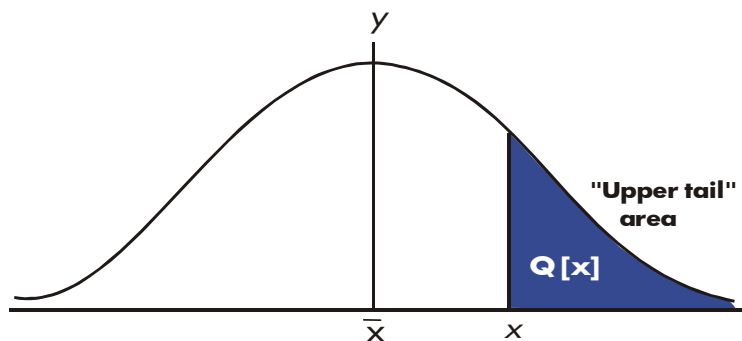
例 2 :

例 1 を、同じデータを使用して対数曲線、指数曲線、累乗曲線のカーブフィットに対して繰り返します。次の表に、それぞれの曲線タイプの開始実行ラベルと結果（相関係数、回帰係数、 x と y の予測値）を示します。異なるカーブフィットのプログラムを実行するときには、そのつどデータ値を再入力する必要があります。

	対数	指数	累乗
開始するには :	<code>XEQ</code> <code>[L]</code> <code>ENTER</code>	<code>XEQ</code> <code>[E]</code> <code>ENTER</code>	<code>XEQ</code> <code>[P]</code> <code>ENTER</code>
R	0.9965	0.9945	0.9959
B	-139.0088	51.1312	8.9730
M	65.8446	0.0177	0.6640
$Y(\hat{y} \text{ when } X=37)$	98.7508	98.5870	98.6845
$X(\hat{x} \text{ when } Y=101)$	38.2857	38.3628	38.3151

正規分布と逆正規分布

正規分布は、平均からのランダムな変動をモデル化するためによく利用されます。モデルは、標本が標準偏差 S で平均 M を軸に対称に分布していると見なし、次の図に示す釣り鐘状の曲線を近似します。このプログラムでは、値 x が与えられた場合に、標本データからランダムに選択した数値が x より大きくなる確率を計算します。これは、上側確率 $Q(x)$ と呼ばれています。このプログラムではその逆、つまり、与えられた確率 $Q(x)$ に対する値 x も計算できます。



$$Q(x) = 0.5 - \frac{1}{\sigma\sqrt{2\pi}} \int_{\bar{x}}^x e^{-((x-\bar{x})/\sigma)^2/2} dx$$

正規分布曲線の方程式を積分するため、HP 35s に内蔵の積分機能を使用しています。逆を求めるには、ニュートン法を使用して、与えられた確率 $Q(x)$ を生成する値 x を反復検索します。

プログラム：

プログラム行： (RPN モード)	説明
S001 LBL S	このルーチンは、正規分布プログラムを初期化します。
S002 0	平均値のデフォルト値を格納します。
S003 STO M	
S004 INPUT M	平均値 M の入力求めて、格納します。
S005 1	標準偏差のデフォルト値を格納します。
S006 STO S	
S007 INPUT S	標準偏差 S の入力求めて、格納します。
S008 RTN	標準偏差の値の表示を終了します。
チェックサムと長さ：70BF 26	
D001 LBL D	このルーチンは、与えられた X に対する $Q(X)$ を計算します。
D002 INPUT X	X の入力求めて、格納します。
D003 XEQ Q001	上側確率を計算します。
D004 STO Q	計算結果を変数 Q に格納し、VIEW 関数で表示できるようにします。
D005 VIEW Q	$Q(X)$ を表示します。
D006 GTO D001	ループして別の $Q(X)$ を計算します。
チェックサムと長さ：042A 18	
I001 LBL I	このルーチンは、与えられた $Q(X)$ に対する X を計算します。
I002 INPUT Q	$Q(X)$ の入力求めて、格納します。
I003 RCL M	平均値を呼び出します。
I004 STO X	平均値を X の推定値 X_{guess} として格納します。
チェックサムと長さ：A970 12	
T001 LBL T	このラベルは、繰り返しループの開始を定義します。
T002 XEQ Q001	$(Q(X_{guess}) - Q(X))$ を計算します。
T003 RCL- Q	
T004 RCL X	
T005 STO D	
T006 R↓	

16-12 統計のプログラム

**プログラム行 :
(RPN モード)**

説明

T007 XEQ F001	X_{guess} の導関数を求めます。
T008 RCL ÷ T	
T009 ÷	X_{guess} の補正値を計算します。
T010 STO+ X	補正値を加算して、新しい X_{guess} を生成します。
T011 ABS	
T012 0.0001	
T013 × < > ?	補正値が有意かどうかテストします。
T014 GTO T001	補正値が有意であれば、ループの初めに戻ります。補正値が有意でなければ続行します。
T015 RCL X	
T016 VIEW X	計算した X の数値を表示します。
T017 GTO I001	ループして別の X を計算します。

チェックサムと長さ : EDF4 57

Q001 LBL Q	このサブルーチンは、上側確率 $Q(x)$ を計算します。
Q002 RCL M	積分の下限値を呼び出します。
Q003 RCL X	積分の上限値を呼び出します。
Q004 FN= F	被積分関数に対して LBL F によって定義される関数を選択します。
Q005 ∫ FN d D	ダミー変数 D を使って正規関数を積分します。
Q006 2	
Q007 π	
Q008 ×	
Q009 $\sqrt{\times}$	
Q010 RCL× S	$S \times \sqrt{2\pi}$ を計算します。
Q011 STO T	逆計算ルーチンのため、結果を一時的に格納します。
Q012 ÷	
Q013 + / -	
Q014 0.5	
Q015 +	積分の下限に平均値を用いたので、曲線の下の方の面積の半分を加算します。
Q0016 RTN	呼び出したルーチンに戻ります。

チェックサムと長さ : 8387 52

プログラム行 : (RPN モード)

説明

F001 LBL F このサブルーチンは、正規関数 $e^{-((X-M) \div S)^2 \div 2}$ の被積分関数を計算します。

F002 RCL D

F003 RCL - M

F004 RCL $\div$ S

F005 $\times 2$

F006 2

F007 $\div$

F008 + / -

F009 e $\times$

F010 RTN 呼び出したルーチンに戻ります。

チェックサムと長さ : B3EB 31

フラグの使用 :

使用されません。

注意点 :

このプログラムの確度は、数値の表示設定に依存します。平均値 ± 3 標準偏差の範囲の入力であれば、ほとんどのアプリケーションでは 4 桁以上の有効桁の表示で十分です。

フル精度での入力限界は、平均値 ± 5 標準偏差の範囲になります。計算時間は、表示桁数が減るにつれて大幅に短縮します。

ルーチン Q では、定数 0.5 の代わりに 2 と $\boxed{1/x}$ を使用できます。

逆計算が不要であれば、逆ルーチンの入力 (I ルーチン と T ルーチン内) も不要です。

プログラム利用手順：

1. プログラムルーチンを入力します。終了したら **[C]** を押します。
2. **[XEQ] [S] [ENTER]** を押します。
3. M のプロンプトが出たら、母平均を入力し、**[R/S]** を押します（平均値が 0 の場合は、**[R/S]** のみを押します）。
4. S のプロンプトが出たら、母標準偏差を入力し、**[R/S]** を押します（標準偏差が 1 であれば、**[R/S]** のみを押します）。
5. $Q(X)$ が与えられた場合に X を計算するには、手順のステップ 9 までスキップします。
6. X が与えられた場合に $Q(X)$ を計算するには、**[XEQ] [D] [ENTER]** を押します。
7. プロンプトが出たら、 X の値を入力し、**[R/S]** を押します。結果 $Q(X)$ が表示されます。
8. 同じ平均値と標準偏差を使って、新しい X に対する $Q(X)$ を計算するには、**[R/S]** を押してステップ 7 に移動します。
9. $Q(X)$ が与えられた場合に X を計算するには、**[XEQ] [I] [ENTER]** を押します。
10. プロンプトが出たら、 $Q(X)$ の値を入力し、**[R/S]** を押します。結果 X_i が表示されます。
11. 同じ平均値と標準偏差を使って、新しい $Q(X)$ に対する X を計算するには、**[R/S]** を押してステップ 10 に移動します。

変数の使用：

D	積分のダミー変数
M	母平均。デフォルト値は 0
Q	上側面積に対応する確率
S	母標準偏差。デフォルト値は 1
T	逆計算を行うプログラムに値 $S \times \sqrt{2\pi}$ を渡すため一時的に使用する変数
X	上側面積の左側を定義する入力値

例 1 :

あなたはある友人に、今度のブラインドデートのお相手の「知性」が " 3σ " だと言われたとします。この相手は、平均値 + 3 標準偏差を超える人数を除いた地元住民より、知性的であると解釈できます。

ブラインドデートに参加する可能性のある異性が 10,000 人いるとします。どれだけの人数が " 3σ " の範囲に入るでしょうか？ この問題は標準偏差に関するものなので、M には 0、S には 1 のデフォルト値を使用します。

キー:	ディスプレイ:	説明:
(RPN モード)		
XEQ S ENTER	M? 0.0000	初期化ルーチンを開始します。
R/S	S? 1.0000	M に対してデフォルト値 0 を受け入れます。
R/S	1.0000	S に対してデフォルト値 1 を受け入れます。
XEQ D ENTER	X? 値	分布プログラムを開始し、X の入力 を求めます。
3 R/S	Q= 0.0013	X に対して 3 を入力し、Q(X) の計算を開始します。平均値 + 3 標準偏差の範囲内にいる人より知性的な集団の割合を表示します。
1 0 0 0 0	13.4984	人口を乗算します。地元住民のうち、
X		基準に合致するブラインドデート相手の人数の近似値を表示します。

この友人は誇張して伝える癖があるので、" 2σ " のデートの割合を計算します。

R/S を押すだけでプログラムの再実行できます。

キー:	ディスプレイ:	説明:
(RPN モード)		
R/S	X? 3.0000	プログラムを再開します。
2 R/S	Q= 0.0228	X 値として 2 を入力し、Q(X) を計算します。
1 0 0 0 0	227.5012	人口で乗算して修正した推測値を得ます。
X		

16-16 統計のプログラム

例 2 :

ある試験の平均得点が 55 点だとします。標準偏差は 15.3 です。正規分布曲線で分布を適切にモデル化できるとして、ランダムに選択した学生の点数が 90 点以上である確率はいくらでしょうか？学生の中でトップ 10%に入るには、何点とれば良いでしょうか？20%の学生を落第にするときは、及第点は何点になるでしょうか？

キー：	ディスプレイ：	説明：
(RPN モード)		
[XEQ] [S] [ENTER]	M? 0.0000	初期化ルーチンを開始します。
[5] [5] [R/S]	S? 1.0000	平均値には 55 を格納します。
[1] [5] [.] [3] [R/S]	15.3000	標準偏差には 15.3 を格納します。
[XEQ] [D] [ENTER]	X? 値	分布プログラムを開始し、X の入力 を求めます。
[9] [0] [R/S]	Q= 0.0111	90 を X に入力し、Q(X) を計算 します。

この計算から、得点が 90 点を超える学生は、わずか約 1% であると推察できます。

キー：	ディスプレイ：	説明：
(RPN モード)		
[XEQ] [I] [ENTER]	Q? 0.0111	逆計算のルーチンを開始します。
[0] [.] [1] [R/S]	X= 74.6077	0.1 (10 %) を Q(X) に格納し、X を 計算します。
[R/S]	Q? 0.1000	逆計算のルーチンを再開します。
[0] [.] [8] [R/S]	X= 42.1232	0.8 (100 % - 20 %) を Q(X) に格納 し、X を計算します。

グループ化された標準偏差

グループデータの標準偏差 S_{xy} は、正の整数頻度 $f_1, f_2, \dots, f_n$ で発生するデータ点 $x_1, x_2, \dots, x_n$ の標準偏差です。

$$S_{xy} = \sqrt{\frac{\sum x_i^2 f_i - \frac{(\sum x_i f_i)^2}{\sum f_i}}{(\sum f_i) - 1}}$$

このプログラムでは、データの入力、入力の修正、グループデータの標準偏差と加重平均の計算を行います。

プログラム：

プログラム行： (ALG モード)	説明
S001 LBL S	グループ化された標準偏差プログラムを開始します。
S002 CLΣ	統計レジスタ (-27 から -32) をクリアします。
S003 0	
S004 STO N	カウント N をクリアします。
チェックサムと長さ：E5BC 13	
I001 LBL I	統計データ点を入力します。
I002 INPUT X	データ点を X に格納します。
I003 INPUT F	データ点の頻度を F に格納します。
I004 1	N の増加値を入力します。
I005 STO B	
I006 RCL F	データ点の頻度 f_i を呼び出します。
チェックサムと長さ：3387 19	
F001 LBL F	総和を累積します。
F002 -27	
F003 STO I	レジスタ -27 のインデックスを格納します。
F004 RCL F	
F005 STO+(I)	レジスタ -27 の $\sum f_i$ を更新します。
F006 RCL×X	$x_i f_i$
F007 STO Z	
F008 -28	

16-18 統計のプログラム

**プログラム行 :
(ALG モード)**

説明

F009 STO I	レジスタ -28 のインデックスを格納します。
F010 RCL Z	
F011 STO+(I)	レジスタ -28 の $\sum x_i f_i$ を更新します。
F012 RCL×X	$x_i^2 f_i$
F013 STO Z	レジスタ -30 のインデックスを格納します。
F014 -30	
F015 STO I	
F016 RCL Z	
F017 STO+(I)	レジスタ -30 の $\sum x_i^2 f_i$ を更新します。
F018 RCL B	
F019 STO+N	N を増加（または減少）します。
F020 RCL N	
F021 RCL F	
F022 ABS	
F023 STO F	
F024 VIEW N	データペアの現在の数を表示します。
F025 GTO I001	次のデータ入力のため、ラベル行番号 I に移動します。
チェックサムと長さ : F6CB 84	
G001 LBL G	グループデータの統計を計算します。
G002 s×	グループ化された標準偏差。
G003 STO S	
G004 VIEW S	グループ化された標準偏差を表示します。
G005 $\bar{x}$	加重平均。
G006 STO M	
G007 VIEW M	加重平均を表示します。
G008 GTO I001	さらに点を処理するため戻ります。
チェックサムと長さ : DAF2 24	
U001 LBL U	Undo でデータ入力エラーを元に戻します。
U002 -1	N の減少値を入力します。
U003 STO B	
U004 RCL F	最後のデータの頻度入力呼び出しします。
U005 +/-	f_i の符号を変更します。
U006 STO F	
U007 GTO F001	カウンタと総和を調整します。

プログラム行： (ALG モード)

説明

チェックサムと長さ：03F4 23

フラグの使用：

使用されません。

プログラム利用手順：

1. プログラムルーチンを入力します。終了したら **[C]** を押します。
2. **[XEQ] [S] [ENTER]** を押して、新しいデータの入力を開始します。
3. x_i 値（データ点）を入力し、**[R/S]** を押します。
4. f_i 値（頻度）を入力し、**[R/S]** を押します。
5. VIEW で入力した点の数を表示後、**[R/S]** を押します。
6. 各データ点に対してステップ 3 から 5 を繰り返します。

ステップ 4 で **[R/S]** を押した後にデータ入力ミスを発見した場合（ x_i または f_i ）、**[XEQ] [U] [ENTER]** を押して、**[R/S]** を再度押します。ステップ 3 に戻って正しいデータを入力します。

7. データペアの入力が終わったら、**[XEQ] [G] [ENTER]** を押して、グループ化された標準偏差を計算し、表示します。
8. **[R/S]** を押して、グループデータの加重平均を表示します。
9. データ点を追加するには、**[R/S]** を押し、ステップ 3 から続行します。
新しい問題を開始するには、ステップ 2 から開始します。変数の使用：

X	データ点
F	データ点の頻度
N	データペアカウンタ
S	グループ化された標準偏差
M	加重平均
i	補正統計レジスタ に間接的にアドレスするために使用する インデックス変数
レジスタ -27	総和 $\sum f_i$
レジスタ -28	総和 $\sum x_i f_i$
レジスタ -30	総和 $\sum x_i^2 f_i$

16-20 統計のプログラム

例：

次のデータを入力し、グループ化された標準偏差を計算します。

グループ	1	2	3	4	5	6
x_i	5	8	13	15	22	37
f_i	17	26	37	43	73	115

キー：
(ALG モード)

ディスプレイ：

説明：

XEQ **S** **ENTER**

X?

最初の x_i の入力をお願いします。

5 **R/S**

値

5 を X に格納し、最初の f_i の入力をお願いします。

1 **7** **R/S**

F?

17 を F に格納し、カウンタを表示します。

R/S

N=

第 2 の x_i の入力をお願いします。

8 **R/S**

1.0000

第 2 の f_i の入力をお願いします。

2 **6** **R/S**

X?

カウンタを表示します。

R/S

5.0000

第 3 の x_i の入力をお願いします。

1 **4** **R/S**

F?

第 3 の f_i の入力をお願いします。

3 **7** **R/S**

17.0000

カウンタを表示します。

N=

2.0000

X?

8.0000

F?

26.0000

N=

3.0000

x₃に13と入力すべきところを14と入力しました。エラーを元に戻すには、ルーチンUを実行します。

[XEQ] [U] [ENTER]

N=

2.0000

間違ったデータを削除します。カウンタが戻ります。

[R/S]

X?

新しい第3の x_i の入力を求めます。

14.0000

[1] [3] [R/S]

F?

新しい第3の f_i の入力を求めます。

37.0000

[R/S]

N=

カウンタを表示します。

3.0000

[R/S]

X?

第4の x_i の入力を求めます。

13.0000

[1] [5] [R/S]

F?

第4の f_i の入力を求めます。

37.0000

[4] [3] [R/S]

N=

カウンタを表示します。

4.0000

[R/S]

X?

第5の x_i の入力を求めます。

15.0000

[2] [2] [R/S]

F?

第5の f_i の入力を求めます。

43.0000

[7] [3] [R/S]

N=

カウンタを表示します。

5.0000

[R/S]

X?

第6の x_i の入力を求めます。

22.0000

[3] [7] [R/S]

F?

第6の f_i の入力を求めます。

73.0000

[1] [1] [5] [R/S]

N=

カウンタを表示します。

6.0000

[XEQ] [G] [ENTER]

S=

第6のデータ点のグループ化された標準偏差 (s_x) を計算し、表示します。

11.4118

[R/S]

M=

加重平均 ($\bar{x}$) を計算し、表示します。

23.4084

[C]

23.4084

VIEW をクリアします。

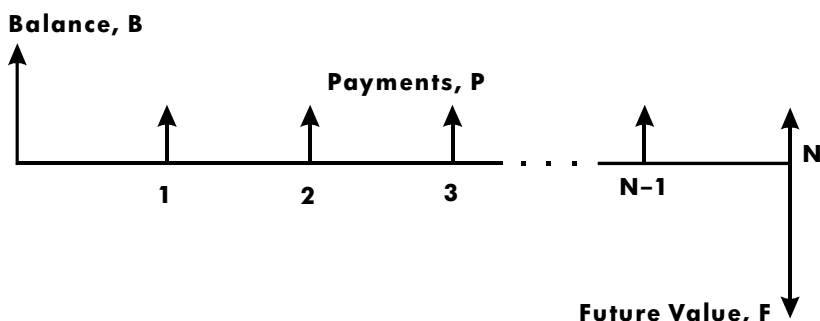
その他のプログラムと方程式

貨幣の時間価値

貨幣の時間的価値（TVM、Time Value of Money）方程式の 5 個の値のうち、4 個がわかれば、5 番目の値を求めることができます。消費者ローンや住宅ローン、積立預金など、金融のさまざまな問題でこの方程式が利用されています。

TVM 方程式は次の通りです。

$$P \left[\frac{1 - (1 + I/100)^{-N}}{I/100} \right] + F(1 + (I/100))^{-N} + B = 0$$



貨幣価値の符号（残高 B 、支払 P 、将来価値 F ）は、キャッシュフロー図の向きに一致します。受け取りには正の符号を、支払いには負の符号を用います。すべての金融問題には、貸し手側と借り手側という、2 種類の観点があり、それぞれで符号が異なります。

方程式の入力：

方程式を入力します。

$$P \times 100 \times (1 - (1 + I \div 100)^{-N}) \div I + F \times (1 + I \div 100)^{-N} + B$$

キー：
(RPN モード)

[EQN]

[RCL] [P] [×] [1] [0] [0]

[×] [()] [1] [−]

[()] [1] [+]

[RCL] [I] [÷] [1] [0] [0]

[>] [y^x]

[+/-] [RCL] [N] [>]

[÷] [RCL] [I] [+] **[RCL] [F]**

[×]

[()] [1] [+] **[RCL] [I]**

[÷] [1] [0] [0] [>]

[y^x] [+/-] [RCL] [N]

[+] [RCL] [B]

[ENTER]

[ON] [SHOW] (hold)

ディスプレイ：

EQN LIST TOP

または現在の方程式
P×100_

P×100×(1−_

P×100×(1−(1+_)

←0×(1−(1+I÷100)_ →

←(1−(1+I÷100)^_ →

←(1+I÷100)^-N)_

←100)^-N)÷I+F×_

←^-N)÷I+F×(1+I)_

←I+F×(1+I÷100)_

←×(1+I÷100)^-N)_

←1+I÷100)^-N+B)_

P×100×(1−(1+I÷

CK=CEFR

LN=41

説明：

方程式モードを選択
します。
方程式の入力を開始
します。

➡ 方程式を終了します。
チェックサムと長さ。

注意点：

TVM 方程式では、DIVIDE BY 0 エラーを回避するため、I がゼロにならないようにする必要があります。I の解を求めるとき、現在の値が不明の場合、SOLVE 計算を開始する (**[SOLVE]**) 前に **[1] [STO] [I]** を押します。

値の入力を求める順番は、解を求める変数に依存します。

17-2 その他のプログラムと方程式

SOLVE 命令 :

1. 最初の TVM 計算で 利率 i を求める場合、**1** **→** **STO** **I** を押します。
2. **EQN** を押します。必要に応じて、**↑** または **↓** を押し、TVM 方程式が表示されるまで方程式リストをスクロールします。
3. 次の 5 つの操作のどれかを行います。
 - a. **→** **SOLVE** **N** を押して、複利期間の数を計算します。
 - b. **→** **SOLVE** **I** を押して、利子を計算します。

月払いの場合、 i に対して戻る結果は、月利 i です。12 **×** を押して年利を確認します。

- c. **→** **SOLVE** **B** を押して、ローンや預金の期首残高を計算します。
 - b. **→** **SOLVE** **P** を押して、定期払い込みを計算します。
 - e. **→** **SOLVE** **F** を押して、ローンの将来価値または残高を計算します。
4. プロンプトに従って、4 つの既知の変数の値を入力し、各値の後に **R/S** を押します。
 5. 最後の **R/S** を押したときに、未知変数の値が計算され、表示されます。
 6. 新しい変数を計算するか、異なるデータを使って同じ変数を再計算するには、ステップ 2 に移動します。

このアプリケーションでは、初期推定値を入力しなくても SOLVE が効率的に機能します。

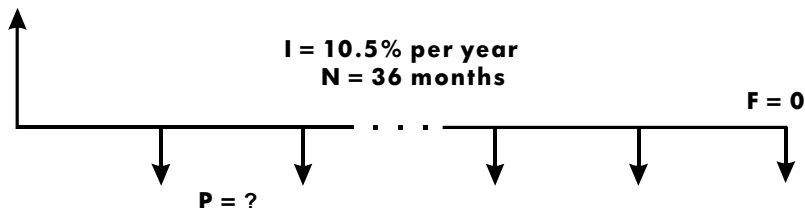
変数の使用 :

N	複利期間の数
i	パーセントで表わされる 定期的な 利率 例えば、 年利 が 15% で、年 12 回の支払いがある場合、 定期的な 利率 i は、 $15 \div 12 = 1.25\%$ になります)。
B	ローンや預金の期首残高
P	定期的な支払い
F	預金やローン残高の将来価値

例：

その 1. 自動車をローンで購入するとします。期間は 3 年（36 ヶ月）で、月毎の複利で年利 10.5% です。自動車の購入価格は 7,250 ドルです。頭金を 1,500 ドルとします。

$$B = 7,250 - 1,500$$



キー：

ディスプレイ：

説明：

(RPN モード)

DISPLAY **1** (1FIX) **2**

EQN (▼) 必要であれば $P \times 100 \times (1 - (1 + I)^{-N})$

▶ **SOLVE** **P**

1 **0** **.** **5** **ENTER**

1 **2** **÷**

R/S

3 **6** **R/S**

0 **R/S**

7 **2** **5** **0** **ENTER**

1 **5** **0** **0** **-**

R/S

I?

値

I?

0.88

N?

値

F?

値

B?

値

B?

5,750.00

SOLVING

P=

-186.89

FIX 2 表示形式を選択します。

TVM 方程式の左端の部分を表示します。

P を選択し、I の入力を求めます。

年利の入力を等価の月利に換算します。

0.88 を I に格納し、N の入力を求めます。

36 を N に格納し、F の入力を求めます。

0 を F に格納し、B の入力を求めます。

期首ローン残高 B を計算します。

5750 を B に格納し、月々の支払い P を計算します。

ローンは借り手の観点から見ているため、計算結果は負になります。借り手が受け取るお金（期首残高）は正で、支払うお金は負です。

その 2. 月の支払いを 10 ドル減らす利率はいくらでしょうか？

キー:	ディスプレイ:	説明:
(RPN モード)		
[EQN]	$P \times 100 \times (1 - (1 + I)^{-N})$	TVM 方程式の左端の部分を表示します。
[F2] [SOLVE] [I]	P? -186.89	I を選択し、P の入力求めます。
[F2] [RND]	P? -186.89	支払いを小数点以下 2 桁に丸めます。
[1] [0] [+]	P? -176.89	新しい支払いを計算します。
[R/S]	N? 36.00	-176.89 を P に格納し、N の入力求めます。
[R/S]	F? 0.00	36 を N に保持し、F の入力求めます。
[R/S]	B? 5,750.00	0 を F に保持し、B の入力求めます。
[R/S]	SOLVING I= 0.56	5750 を B に保持し、月利を計算します。
[1] [2] [X]	6.75	年利を計算します。

その 3. 計算した年利（6.75%）を使用し、2 年後に自動車を売却するとします。ローンはいくら残っているのでしょうか？つまり、2 年後の将来価値はいくらでしょうか？

「その 2」からの利率 I はゼロではないため、新しい I を計算するときに DIVIDE BY 0 エラーが発生しません。

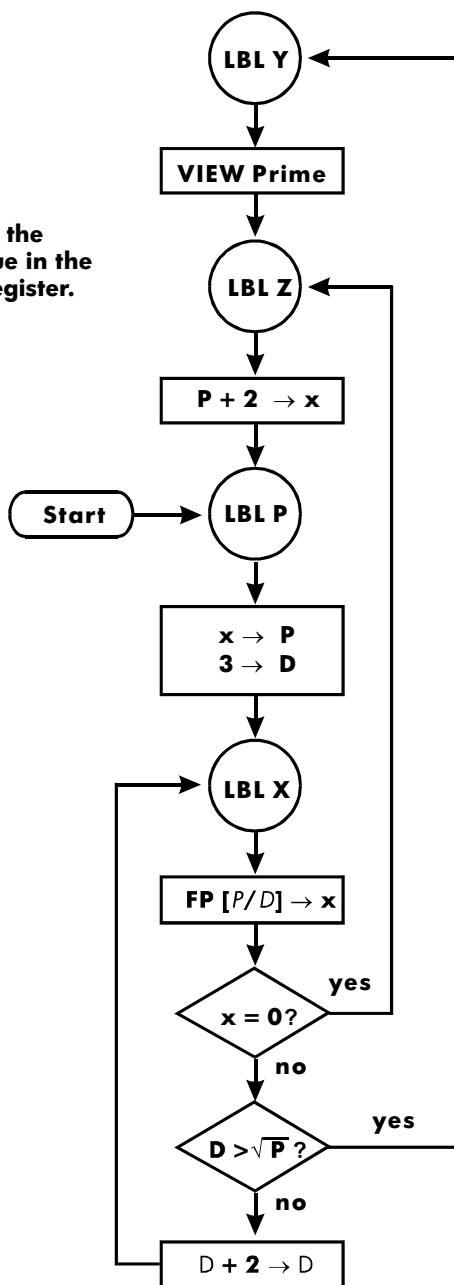
キー:	ディスプレイ:	説明:
(RPN モード)		
[EQN]	$P \times 100 \times (1 - (1 + I)^{\rightarrow})$	TVM 方程式の左端の部分を表示します。
[2] [SOLVE] [F]	P? -176.89	F を選択し、 P の入力を求めます。
[R/S]	I? 0.56	P を保持し、 I の入力を求めます。
[R/S]	N? 36.00	0.56 を I に保持し、 N の入力を求めます。
[2] [4] [R/S]	B? 5,750.00	24 を N に格納し、 B の入力を求めます。
[R/S]	SOLVING F= -2,047.05	5750 を B に保持し、将来価値 F を計算します。符号が負であるため、支払残高があることを示しています。
[<] [DISPLAY] [1]		FIX 4 表示形式を設定します。
(FIX) [4]		

素数生成器

このプログラムは、4 以上の正の整数の入力を受け付けます。それが素数（その数自身と 1 でのみ割り切れる数値）の場合、プログラムが入力値を戻します。入力値が素数でない場合、プログラムが、入力値より大きい最初の素数を戻します。

プログラムは、素数でない数値を識別するため、すべての可能な因数を試します。素数でない場合、2 を加算し（奇数であると仮定して）素数が見つかるかテストします。素数が見つかるまで、この工程が繰り返されます。

Note: x is the value in the X-register.



プログラム：

プログラム行： (ALG モード)

説明

Y001 LBL Y このルーチンは素数 P を表示します。

Y002 VIEW P

チェックサムと長さ：2CC5 6

Z001 LBL Z このルーチンは、2 を P に加算します。

Z002 2+ P

チェックサムと長さ：EFB2 9

P001 LBL P このルーチンは、 P に対する入力値を格納します。

P002 LASTx▶ P

P003 FP($P \div 2$)

P004 x<>v

P005 0

P006 x=v? 偶数かどうか判定します

P007 1+P▶P 入力が偶数であれば、 P を増加します。

P008 3▶D 3 を テストの除数 D に格納します。

チェックサムと長さ：EA89 47

X001 LBL X このルーチンは、 P が素数かどうかを判定します。

X002 FP($P \div D$) $P \div D$ の小数部を求めます。

X003 x=0? 剰余がゼロかどうかを判定します（ゼロであれば素数
ではありません）。

X004 GTO Z001 素数でないときは、次の可能性を試します。

X005 SQRT(P)

X006 x<>v

X007 0

X008 x>v? すべての可能な因数を試行したかどうかを判定します。

X009 GTO Y001 すべての因数を試行したら、表示ルーチンに分岐
します。

X010 2+D▶D

X011 GTO X001 新しい因数で可能な素数をテストするため分岐します。

チェックサムと長さ：C6B5 53

フラグの使用：

使用されません。

プログラム利用手順：

1. プログラムルーチンを入力します。終了したら **[C]** を押します。
2. 4 以上の正の整数を入力します。
3. **[XEQ] [P] [ENTER]** を押してプログラムを実行します。素数 P が表示されます。
4. 次の素数を表示するには、**[R/S]** を押します。

変数の使用：

P	素数、または素数の可能性のある数値
D	P の現在の値をテストするための除数

注意点：

入力が 3 を超えるかどうかのテストは行われていません。

例：

789 の次の素数は何でしょうか？ さらにその次の素数は何でしょうか？

キー：

ディスプレイ：

説明：

(ALG モード)

[7] [8] [9] [XEQ]

P=

789 の次の素数を計算します。

[P] [ENTER]

797.0000

[R/S]

P=

797 の次の素数を計算します。

809.0000

ベクトルのクロス積

この例では、クロス積を計算するプログラム関数の使い方を示します。

クロス積

$$\mathbf{v}_1 \times \mathbf{v}_2 = (YW - ZV)\mathbf{i} + (ZU - XW)\mathbf{j} + (XV - YU)\mathbf{k}$$

ここで、

$$\mathbf{v}_1 = X\mathbf{i} + Y\mathbf{j} + Z\mathbf{k}$$

および

$$\mathbf{v}_2 = U\mathbf{i} + V\mathbf{j} + W\mathbf{k}$$

プログラム行 : (RPN モード)

説明

R001 LBL R	直交入力／表示ルーチンの開始を定義します。
R002 INPUT X	Xの入力を表示するか、受け付けます。
R003 INPUT Y	Yの入力を表示するか、受け付けます。
R004 INPUT Z	Zの入力を表示するか、受け付けます。
R005 GTO R001	ベクトルを入力するため R001 に移動します
チェックサムと長さ : D82E 15	
E001 LBL E	ベクトル入力ルーチンの開始を定義します。
E002 RCL X	X 、Y、Zの値をそれぞれ、U、V、W にコピーします。
E003 STO U	
E004 RCL Y	
E005 STO V	
E006 RCL Z	
E007 STO W	
E008 GTO R001	ベクトルを入力するため R001 に移動します
チェックサムと長さ : B6AF 24	

**プログラム行 :
(RPN モード)**

説明

C001 LBL C	クロス積ルーチンの開始を定義します。
C002 RCL Y	
C003 RCL×W	
C004 RCL Z	
C005 RCL×V	
C006 -	X 成分である $(YW - ZV)$ を計算します。
C007 STO A	
C008 RCL Z	
C009 RCL×U	
C010 RCL X	
C011 RCL×W	
C012 -	Y 成分である $(ZU - WX)$ を計算します。
C013 STO B	
C014 RCL X	
C015 RCL×V	
C016 RCL Y	
C017 RCL×U	
C018 -	
C019 STO Z	Z 成分である $(XV - YU)$ を格納します。
C020 RCL A	
C021 STO X	X 成分を格納します
C022 RCL B	
C023 STO Y	Y 成分を格納します
C024 GTO R001	ベクトルを入力するため R001 に移動します
チェックサムと長さ : 838D 72	

例：

2 つのベクトル $v_1=2i+5j+4k$ と $v_2=i-2j+3k$ のクロス積を計算します

キー：	ディスプレイ：	説明：
XEQ R ENTER	$\times?$	R ルーチンを実行してベクトル値
1 R/S	$\nu?$	値を入力します x 成分の ν_2 を入力します
2 +/- R/S	$z?$	値 y 成分の ν_2 を入力します
3 R/S	$\times?$	値 z 成分の ν_2 を入力します
XEQ E ENTER	$\times?$	1 E ルーチンを実行して、U、V、
2 R/S	$\nu?$	1 W 変数の ν_2 を交換します x 成分の ν_1 を入力します
5 R/S	$z?$	-2 y 成分の ν_1 を入力します
4 R/S	$\times?$	3 z 成分の ν_1 を入力します
XEQ C ENTER	$\times?$	2 C ルーチンを実行して、クロス
R/S	$\nu?$	23 積の x 成分を計算します クロス積の y 成分を計算します
R/S	$z?$	-2 クロス積の z 成分を計算します
		-9

第 3 部

付録

サポート、電池、およびサービス

サポート

電卓の使用方法については、電卓サポート部門にお問い合わせください。次の「一般的な質問への回答」に、製品についてのよくある質問をまとめて示します。その他の質問は、A-8 ページに示す電卓サポート部門にお問い合わせください。

一般的な質問への回答

Q：どうすれば電卓が適切に動作しているか判断できますか。

A：A-5 ページの診断セルフテストの説明を参照してください。

Q：小数点にピリオドではなくカンマが使用されています。どうすればピリオドに切り替えることができますか。

A： **DISPLAY**  (5.) 機能 (1-23 ページ) を使用してください。

Q：どうすればディスプレイの小数点以下の桁数を変更できますか。

A： **DISPLAY** メニュー (1-21 ページ) を使用してください。

Q：どうすればメモリの全部または一部をクリアできますか。

A： **CLEAR** を押すと CLEAR メニューが表示され、x (X レジスタの数値)、すべての直接変数、すべてのメモリ、すべての統計データ、すべてのスタックレベル、すべての間接変数をクリアできます。

Q：数値の 摘煤 ($2.51E-13$ など) は何を意味しますか。

A: 10 の指数です。この例では 2.51×10^{-13} です。

Q: 電卓に MEMORY FULL というメッセージが表示されました。どうすればいいでしょうか。

A: 次の操作の前にメモリの一部をクリアする必要があります (付録 B を参照してください)。

Q: π ラジアン の正弦または正接を計算すると、ゼロではなく非常に小さい数値が表示されるのはなぜですか。

A: π は、電卓の内部精度である 12 桁では正確に表現できません。

Q: 三角関数を使用すると計算結果が間違っているのはなぜですか。

A: 電卓が正しい角度モード (**MODE** 1DEG、2RAD、3GRD) を使用していることを確認してください。

Q: ディスプレイのインジケータは何を意味していますか。

A: 電卓の状態を示します。第 1 章の「インジケータ」を参照してください。

Q: 数値が分数として表示されます。小数にするにはどうすればよいですか。

A:  **FDISP** を押します。

環境限界

製品の信頼性を保つには、次の温度と湿度の限界を守ってください。

- 動作温度: 0 ~ 45 °C (32 ~ 113 °F)
- 保存温度: -20 ~ 65 °C (-4 ~ 149 °F)
- 動作湿度/保存湿度: 最大 90% の相対湿度、40 °C (104 °F)

電池の交換

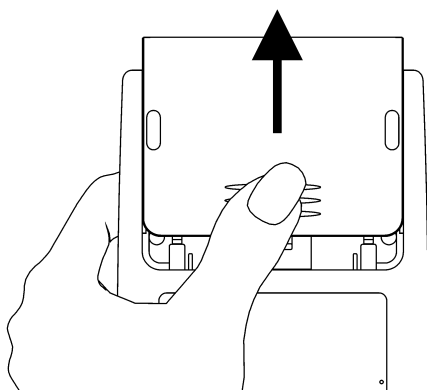
電卓は、3 V の CR2032 リチウムボタン電池を 2 個使用します。

電池容量低下記号（）が表示されたら、できるだけ早く電池を交換してください。電池の表示記号が出てディスプレイが暗くなってきたら、データが消失する可能性があります。消失した場合、メッセージ MEMORY CLEAR が表示されます。

電池を取り出したら、保存された情報が失われるのを防ぐため、2 分以内に新しい電池を入れてください（新しい電池を手元に用意してから電池カバーを開けてください）。

電池を装着するには：

1. 新しいボタン型電池を 2 個用意します。電池の電極に触れないよう、電池の縁をつかんでください。
2. 電卓がオフになっていることを確認します。**電池の入れ替えが完了するまで ON（）を押さないでください。**電池を取り出した状態で電卓をオンにすると、連続メモリの内容が消去されます。
☞ BK+B
3. 電卓を裏返して、電池カバーをずらして外します。



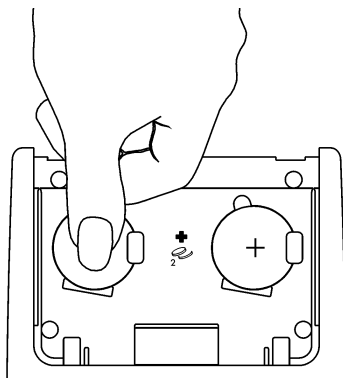
4. メモリ内のデータが失われないよう、古い電池 2 個を同時に取り外さないでください。電池の取り外しと交換は、必ず 1 個ずつ行います。

警告



電池を解体したり、穴をあけたり、火中に廃棄したりしないでください。電池が爆発または破裂し、有害な化学物質が放出される可能性があります。

5. プラス記号（+）を外側に向けて、新しい CR2032 リチウム電池を 1 個装着します。



6. 手順 4 と 5 に従い、もう一方の電池を取り外して装着します。両方の電池のプラス記号（+）が外側に向いていることを確認してください。
7. 電池ケースカバーを元に戻します。
8. **[C]** を押します。

電卓の動作テスト：

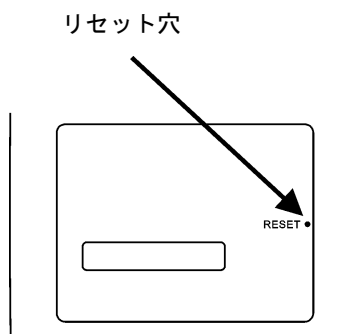
次のガイドラインを使用して、電卓が正しく動作しているかどうかを判断します。各手順の後に電卓をテストして、操作が復元されているか確認します。電卓にサービスが必要な場合は、A-8 ページを参照してください。

- 電卓の電源が入らない（手順 1 ～ 4）か、またはキーを押しても反応しない（手順 1 ～ 3）場合：

1. 電卓をリセットします。**[C]** キーを押しながら **[GTO]** を押します。必要に応じてリセットキー入力を複数回繰り返します。
2. メモリを消去します。**[C]** を押しながら、**[R/S]** と **[i]** を同時に押します。メモリがクリアされ、3 つのキーすべてを離すと、メッセージ MEMORY CLEAR が表示されます。

A-4 サポート、電池、およびサービス

3. 電池を取り外し（「電池の交換」を参照）、電卓内の両方の電池接点に硬貨を軽く押し当てます。電池を戻し、電卓をオンにします。MEMORY CLEAR が表示されるはずですが。
4. リセットしても電卓がキー入力に反応しない場合は、薄い先のとがったもので、リセット穴を押します。格納されたデータは通常、そのまま残ります。



これらの手順で電卓の動作が復元しない場合は、サービスが必要です。

■ **電卓がキー入力に反応するものの、動作不良が疑われる場合：**

1. 次節の説明に従ってセルフテストを実行します。電卓がセルフテストに失敗した場合は、サービスが必要です。
2. セルフテストに合格した場合は、操作ミスの可能性があります。マニュアルの関連部分を再読し、「一般的な質問への回答」（A-1 ページ）を確認してください。
3. A-8 ページに示す電卓サポート部門にお問い合わせください。

セルフテスト

ディスプレイをオンにできるものの、電卓が正しく動作しているようには見えない場合、次の診断セルフテストを実行します。

1. **[C]** キーを押しながら、**[XEQ]** を同時に押します。
2. 任意のキーを 8 回押し、表示されるさまざまなパターンを確認します。
キーを 8 回押すと、電卓が、著作権メッセージ © 2007 HP DEV CO., L.P. とメッセージ KBD 01 を表示します。

3. 次の順番でキーを押します。

[R/S] → **[GTO]** → **[XEQ]** → **[MODE]** → **[^]** → **[<]** → **[>]** → **[RCL]** → **[R↓]** → **[x↔y]** → **[i]** → **[v]** → **[SIN]** → **[COS]** → **[TAN]** → **[√x]** → **[y^x]** → **[1/x]** → **[ENTER]** → **[+/-]** → **[E]** → **[()]** → **[←]** → **[EQN]** → **[7]** → **[8]** → **[9]** → **[÷]** → **[↵]** → **[4]** → **[5]** → **[6]** → **[X]** → **[→]** → **[1]** → **[2]** → **[3]** → **[=]** → **[C]** → **[0]** → **[.]** → **[Σ+]** → **[+]**

- 順番通りにキーを押し、キーが正しく機能している場合、KBD の後に 2 桁の数値が表示されます（電卓は、16 進数を使用してキーをカウントしています）。
 - キー入力が順番通りではなかった場合や、キーが正しく機能していない場合、次のキー入力の際に失敗のメッセージが表示されます（ステップ 4 を参照してください）。
4. 表示されるセルフテストの結果には、次の 2 種類あります
 - セルフテストに合格した場合は、35S-OK と表示されます。ステップ 5 に進んでください。
 - セルフテストに失敗した場合は、35S-FAIL と 1 桁の数値が表示されます。順番通りにキーを入力しなかったことがメッセージの原因であれば、電卓をリセットし（**[C]** を押しながら、**[GTO]** を押す）、セルフテストを再実行します。キーを順番通りに入力してもこのメッセージが表示された場合は、セルフテストを繰り返して結果を確認します。電卓が再度失敗した場合は、サービスが必要です（A-8 を参照してください）。サービスのために電卓を送り返すときは、失敗メッセージのコピーを添付してください。
 5. セルフテストを終了するには、電卓をリセットします（**[C]** を押しながら、**[GTO]** を押します）。

[C] と **[MODE]** を押すと、工場で使用されている連続セルフテストが開始します。任意のキーを押すと、工場テストを終了できます。

A-6 サポート、電池、およびサービス

保証

HP 35s 関数電卓、保証期間：12 か月

1. HP は、お客様が HP から購入された HP ハードウェア製品、付属品、およびサプライ品について、上記の保証期間中、材質および製造過程での問題がないことを保証します。HP は、保証期間中に欠陥品である旨の連絡を受けた場合、自社の判断に基づいて欠陥品を修理または交換します。交換用の製品は、新品もしくは新品に近い製品とします。
2. HP は、お客様が HP から購入された HP ソフトウェア製品について、上記の保証期間中、正しくインストールし使用している場合に限り、品質および製造過程での問題により、プログラム命令の正常な実行が妨げられないことを保証します。HP は、保証期間中に欠陥品である旨の連絡を受けた場合、自社の判断に基づいて、このような欠陥によりプログラム命令を正常に実行できないソフトウェア メディアを交換します。
3. HP は、この製品の動作に中断が生じないこと、またはエラーがないことを保証しません。HP は、保証対象の欠陥品を妥当な期間内に適切に修理または交換できない場合、お客様から購入証明付きで当該製品が遅滞なく返却されれば、製品の購入代金を払い戻します。
4. HP 製品には、新品と同等の性能を有する再生部品、または付帯的に使用されていた部品が含まれている可能性があります。
5. 上記の保証は、(a) 誤った、または不適切な保守やキャリブレーション、(b) HP 以外のソフトウェア製品、インタフェース、部品、またはサプライ品、(c) 当社が認めていない改造または誤使用、(d) 製品の公表された動作仕様範囲外の操作、(e) 不適切な場所での準備や保守から生じた欠陥については適用されません。

6. 書面または口頭にかかわらず、当社は本製品に関して上記以外の明示的保証または条件提示は一切行いません。地域の法律で認められる範囲において、商品性、許容品質、特定用途としての適合性に対するすべての暗示的保証は、上記に明示された保証期間に限定されます。国、州、または地域によっては、暗示的保証期間に対する制限を認めていない場合もあるため、上記の制限事項または免責事項が適用されないことがあります。本保証規定は、お客様に特定の法律上の権利を与えるものです。また、お客様は、他の権利を有する場合がありますが、その権利は国、州、または地域によって異なります。
7. 地域の法律で認められる範囲内において、本保証に記載された救済措置が唯一の排他的な救済措置となります。上記を除き、HP またはそのサプライヤは、データの損失もしくは直接的、特殊、偶発的、結果的（利益の損失を含む）またはその他の損害につき、契約、不法行為その他いかなる理由によってでも、責任を負いません。国、州、または地域によっては、偶発的または結果的損害に対する免責または制限を認めていない場合もあるため、上記の制限事項または免責事項が適用されないことがあります。
8. HP 製品およびサービスに関する保証は、当該製品およびサービスに付帯している明示的保証規定に記載されている事項に限定されます。本保証規定に含まれている技術的または編集上の誤り、もしくは欠落について、HP は責任を負いません。

オーストラリアおよびニュージーランドでは、法律で許可される範囲を除き、本保証規定の条項によって、本製品の販売に対して適用される義務的な法的権利を排除、制限、変更することはできません。この保証の条件は、これらの法的権利に加えて適用されます。

カスタマサポート

AP

国：	電話番号
オーストラリア	1300-551-664 または 03-9841-5211
中国	010-68002397

香港	2805-2563
インドネシア	+65 6100 6682
日本	+852 2805-2563
マレーシア	+65 6100 6682
ニュージーランド	09-574-2700
フィリピン	+65 6100 6682
シンガポール	6100 6682
韓国	2-561-2700
台湾	+852 2805-2563
タイ	+65 6100 6682
ベトナム	+65 6100 6682

EMEA

国：	電話番号
オーストリア	01 360 277 1203
ベルギー	02 620 00 86
ベルギー	02 620 00 85
チェコ共和国	296 335 612
デンマーク	82 33 28 44
フィンランド	09 8171 0281
フランス	01 4993 9006
ドイツ	069 9530 7103
ギリシャ	210 969 6421
オランダ	020 654 5301
アイルランド	01 605 0356
イタリア	02 754 19 782
ルクセンブルク	2730 2146
ノルウェー	23500027
ポルトガル	021 318 0093
ロシア	495 228 3050
南アフリカ	0800980410
スペイン	913753382
スウェーデン	08 5199 2065
スイス(フランス語)	022 827 8780
スイス(ドイツ語)	01 439 5358

スイス（イタリア語）	022 567 5308
イギリス	0207 458 0161

LA

国：	電話番号
アンギラ	1-800-711-2884
アンティグア	1-800-711-2884
アルゼンチン	0-800- 555-5000
アルーバ	800-8000 ♦ 800-711-2884
バハマ	1-800-711-2884
バルバドス	1-800-711-2884
バミューダ	1-800-711-2884
ボリビア	800-100-193
ブラジル	0-800-709-7751
英国領バージン諸島	1-800-711-2884
ケイマン諸島	1-800-711-2884
キュラソー	001-800-872-2881 + 800-711-2884
チリ	800-360-999
コロンビア	01-8000-51-4746-8368 (01-8000-51- HP INVENT)
コスタリカ	0-800-011-0524
ドミニカ	1-800-711-2884
ドミニカ共和国	1-800-711-2884
エクアドル	1-999-119 ♦ 800-711-2884 (Andinatel) 1-800-225-528 ♦ 800-711-2884 (Pacifitel)
エルサルバドル	800-6160
フランス領アンティル諸島	0-800-990-011 ♦ 800-711-2884
フランス領ギアナ	0-800-990-011 ♦ 800-711-2884
グレナダ	1-800-711-2884
グアドループ	0-800-990-011 ♦ 800-711-2884
グアテマラ	1-800-999-5105
ガイアナ	159 ♦ 800-711-2884
ハイチ	183 ♦ 800-711-2884

ホンジュラス	800-0-123 ♦ 800-711-2884
ジャマイカ	1-800-711-2884
マルティニク	0-800-990-011 ♦ 877-219-8671
メキシコ	01-800-474-68368 (800 HP INVENT)
モントセラート	1-800-711-2884
オランダ領アンティル諸島	001-800-872-2881 ♦ 800-711-2884
ニカラグア	1-800-0164 ♦ 800-711-2884
パナマ	001-800-711-2884
パラグアイ	(009) 800-541-0006
ペルー	0-800-10111
プエルトリコ	1-877 232 0589
セントルシア	1-800-478-4602
セントビンセント	01-800-711-2884
セントクリストファーネイビス	1-800-711-2884
セントマーチン	1-800-711-2884
スリナム	156 ♦ 800-711-2884
トリニダードトバゴ	1-800-711-2884
タークスカイコス	01-800-711-2884
米国領バージン諸島	1-800-711-2884
ウルグアイ	0004-054-177
ベネズエラ	0-800-474-68368 (0-800 HP INVENT)

NA

国 :	電話番号
カナダ	800-HP-INVENT
USA	800-HP INVENT

最新のサービスとサポートに関する情報については、<http://www.hp.com> をご覧ください。

Regulatory information

Federal Communications Commission Notice

This equipment has been tested and found to comply 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. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment 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.
- Increase the separation between the equipment and the receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio or television technician for help.

Modifications

The FCC requires the user to be notified that any changes or modifications made to this device that are not expressly approved by Hewlett-Packard Company may void the user's authority to operate the equipment.

Declaration of Conformity for Products Marked with FCC Logo, United States Only

This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions: (1) this device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.

If you have questions about the product that are not related to this declaration, write to

Hewlett-Packard Company
P. O. Box 692000, Mail Stop 530113
Houston, TX 77269-2000

For questions regarding this FCC declaration, write to
Hewlett-Packard Company

P. O. Box 692000, Mail Stop 510101

Houston, TX 77269-2000

or call HP at 281-514-3333

To identify your product, refer to the part, series, or model number located on the product.

Canadian Notice

This Class B digital apparatus meets all requirements of the Canadian Interference-Causing Equipment Regulations.

Avis Canadien

Cet appareil numérique de la classe B respecte toutes les exigences du Règlement sur le matériel brouilleur du Canada.

European Union Regulatory Notice

This product complies with the following EU Directives:

- Low Voltage Directive 2006/95/EC
- EMC Directive 2004/108/EC

Compliance with these directives implies conformity to applicable harmonized European standards (European Norms) which are listed on the EU Declaration of Conformity issued by Hewlett-Packard for this product or product family.

This compliance is indicated by the following conformity marking placed on the product:

 This marking is valid for non-Telecom products and EU harmonized Telecom products (e.g. Bluetooth).	 This marking is valid for EU non-harmonized Telecom products. *Notified body number (used only if applicable - refer to the product label)
--	--

Hewlett-Packard GmbH, HQ-TRE, Herrenberger Strasse 140, 71034 Boeblingen, Germany

Japanese Notice

この装置は、情報処理装置等電波障害自主規制協議会（VCCI）の基準に基づくクラス B 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。

欧州連合における一般家庭内でのユーザーによる廃棄対象機器の廃棄



製品やパッケージ上に見られる本記号は、本製品をその他の家庭ごみと一緒に廃棄してはならないことを示します。代わりに、所有者は廃棄電気および電子機器のリサイクル用に指定された収集地にまで廃棄機器を持参して、廃棄機器を廃棄する責任を負います。廃棄時における廃棄機器の分別収集とリサイクルにより、自然資源が保護され、人間の健康と環境を保護する方法で廃棄機器がリサイクルされます。リサイクル用の廃棄機器の収集地について

では、お住まいの地区の市役所、廃棄物処理業者、または製品を購入された店舗にお問い合わせください。

Perchlorate Material - special handling may apply

This calculator's Memory Backup battery may contain perchlorate and may require special handling when recycled or disposed in California.

メモリとスタック

この付録では、次の項目について説明します。

- ユーザメモリの割当と要件
- メモリに影響を与えずにリセットする方法
- すべてのユーザメモリをクリアし、システムデフォルトをリセットする方法
- スタックの上昇に影響を与える操作

電卓メモリの管理

HP 35s には 30KB のユーザメモリが搭載されており、データの任意の組合せ（変数、方程式、プログラム行）を格納できます。SOLVE、 $\int$ FN、統計計算にもユーザメモリが必要です（ $\int$ FN は特に多くのメモリを消費します）。

格納したデータは、ユーザが明示的にクリアするまでメモリに残ります。メッセージ MEMORY FULL は、その直前の操作のためのメモリが不足していることを示しています。一部（または全部）のユーザメモリをクリアする必要があります。例えば以下を実行できます。

- 一部またはすべての方程式をクリアします（第 6 章の「方程式の編集とクリア」を参照してください）。
- 一部またはすべてのプログラムをクリアします（第 13 章の「プログラムのクリア」を参照してください）。
- すべてのユーザメモリをクリアします（ CLEAR  (3ALL) を押します）。

利用可能なメモリ量を調べるには、 MEM を押します。利用できるバイト数が表示されます。

方程式リスト内の特定の方程式のメモリ要件を確認するには：

1. **[EQN]** を押して、方程式モードをアクティブにします (EQN LIST TOP または現在の方程式の左端が表示されます)。
2. 必要に応じて、目的の方程式が表示されるまで方程式リストをスクロールします (**[↑]** または **[↓]** を押します)。
3. **[☐] [SHOW]** を押して、方程式のチェックサム (16 進数) と長さ (バイト数) を表示します。例、CK=382E LN=41。

特定のプログラムの合計メモリ要件を確認するには：

1. **[☐] [MEM] [2]** (2PGM) を押して、プログラムリストの最初のラベルを表示します。
2. プログラムリストをスクロールします (目的のプログラムラベルとサイズが表示されるまで、**[↑]** または **[↓]** を押します)。例、LBL F LN=57。
3. オプション：**[☐] [SHOW]** を押して、プログラムのチェックサム (16 進数) と長さ (バイト数) を表示します。例、CK=9CC9 LN=57。

プログラム内の方程式のメモリ要件を確認するには：

1. 方程式を含むプログラム行を表示します。
2. **[☐] [SHOW]** を押して、チェックサムと長さを表示します。
例、CK=AB71 LN=15。

電卓のリセット

電卓がキー入力に対して反応しなくなったり、異常な動作を示す場合には、電卓のリセットを試みてください。リセットにより、現在の計算が停止し、プログラム入力、数値入力、プログラム実行、SOLVE の計算、 $\int$ FN の計算、VIEW 表示、INPUT 表示がキャンセルされます。格納されたデータは通常、そのまま残ります。

電卓をリセットするには、**[C]** キーを押しながら、**[GTO]** を押します。電卓をリセットできない場合は、新しい電池に交換してみてください。それでもリセットできないときや、リセットしても正しく動作しないときは、次節の特別の手順でメモリをクリアしてみてください。

それでも電卓がキー入力に反応しない場合は、薄い先のとがったもので、リセット穴を押します。
電卓を落としたり、電力が途切れると、電卓がリセットされることがあります。

B-2 メモリとスタック

メモリのクリア

ユーザメモリをクリアするには通常、 **CLEAR**  (3ALL) を押します。ただし、その他の情報をリセットでき、キーボードが正しく機能しない場合に有効な、より強力なクリア方法もあります。

電卓がキーに反応しないときや、リセットや電池交換を行っても操作を復元できない場合は、次に示す MEMORY CLEAR 手順を実行してください。キー入力によりすべてのメモリがクリアされ、電卓がリセットされ、さらにすべての表示形式とモード設定がデフォルト設定に戻ります（次を参照）。

1.  キーを押し続けます。
2.  を押し続けます。
3.  を押します（3 個のキーを同時に押すことになります）。3 つのキーすべてを離すと、操作が成功した場合に MEMORY CLEAR が表示されます。

カテゴリ	CLEAR ALL	MEMORY CLEAR (デフォルト)
角度モード	変更なし	度
基数モード	変更なし	10 進数
コントラスト設定	変更なし	Medium
小数点	変更なし	"."
桁区切り	変更なし	"1,000"
分母（/c 値）	変更なし	4095
表示形式	変更なし	FIX 4
フラグ	変更なし	クリア
複素数モード	変更なし	xiy
分数表示モード	変更なし	オフ
乱数シード値	変更なし	ゼロ
方程式ポインタ	EQN LIST TOP	EQN LIST TOP
方程式リスト	クリア	クリア
FN= ラベル	ヌル	ヌル
プログラムポインタ	PRGM TOP	PRGM TOP
プログラムメモリ	クリア	クリア
スタック上昇	有効	有効
スタックレジスタ	ゼロにクリア	ゼロにクリア

変数 間接変数 論理	ゼロにクリア 未定義 変更なし	ゼロにクリア 未定義 RPN
------------------	-----------------------	----------------------

電卓を落とした場合、または電力が途切れた場合、メモリがクリアされることがあります。

スタック上昇の状態

4 個のスタックレジスタは常にメモリに存在し、スタックには「スタック上昇の状態」が定義されています。つまり、X レジスタに次の数値が格納されたときの動作に対して、スタック上昇が**有効**または**無効**になります（第 2 章「自動メモリスタック」を参照してください）。

次の 2 つのリスト内の関数を除くすべての関数により、スタック上昇が有効になります。

無効になる操作

5 つの操作 **ENTER**、 **$\Sigma+$** 、 **$\Sigma-$** 、** CLEAR 1** (1 $\times$)、** CLEAR 5** (5STK) は、スタック上昇を無効にします。無効になる操作の後に入力された数値は、X レジスタの現在の数値に上書きされます。Y レジスタ、Z レジスタ、T レジスタはそのまま、変更されません。

さらに、**C** と **** が Clx のように動作するとき、これらもスタック上昇を無効にします。

INPUT 機能は、入力を求めるためにプログラムを停止する際、スタック上昇を**無効**にしますが（したがって入力した数値は X レジスタに上書きされますが）、プログラムが再開するときにスタック上昇を**有効**にします。

中立的操作

次の操作は、スタック上昇の状態に対して影響を及ぼしません。

DEG, RAD, GRAD	FIX, SCI, ENG, ALL	DEC, HEX, OCT, BIN	CLVARS
PSE	SHOW	RADIX . RADIX ,	CLΣ
OFF RCL +	R/S と STOP	^ と ∇	C * と ← *
MEM 1	MEM 2	GTO . .	GTO . ラベル nnn
(1VAR)**	(2PGM)**		
EQN	FDISP	エラー	PRGM とプログラム入力
2 進数のウィン ドウの切り替え	数値入力	xiy pθα	UNDO
* CLx のように使用する場合があります			
** スタック上昇を有効にする {VAR} ENTER と {PGM} XEQ を除いて、 カタログが表示されているあいだに実行されたすべての操作を含みます。			

LAST X レジスタの状態

RPN モードにおける次の操作により、 x が LAST X レジスタに格納されます。

$+, -, \times, \div$	$\sqrt{x}, x^2,$	$e^x, 10^x$
LN, LOG	$y^x, \sqrt[x]{y}$	$1/x, \text{INT}\div, \text{Rmdr}$
SIN, COS, TAN	ASIN, ACOS, ATAN	$\hat{x} \hat{y}$
SINH, COSH, TANH	ASINH, ACOSH, ATANH	IP, FP, SGN, INTG, RND, ABS
%, %CHG	$\Sigma+, \Sigma-$ HMS $\rightarrow$, $\rightarrow$ HMS	RCL+, $-, \times, \div$ $\rightarrow$ DEG, $\rightarrow$ RAD
nCr nPr	!	ARG
CMPLX $+, -, \times, \div$	CMPLX $e^x, \text{LN}, y^x, 1/x$	CMPLX SIN, COS, TAN
$\rightarrow$ kg, $\rightarrow$ lb	$\rightarrow$ $^{\circ}\text{C}$, $\rightarrow$ $^{\circ}\text{F}$	$\rightarrow$ cm, $\rightarrow$ in
$\rightarrow$ l, $\rightarrow$ gal	$\rightarrow$ KM $\rightarrow$ MILE	

$\angle c$ は LAST X レジスタに影響しません。

呼び出し算術シーケンス **[X] [RCL] [+]** 変数が x を LASTx に格納し、**[X] [RCL]** 変数 **[+]** が呼び出した数値を LASTx に格納します。

ALG モードでは、LAST X レジスタはスタックの付随レジスタです。最後の式の計算結果である数値を保持します。これにより、ALG モードで前の式の結果を再利用できます。

スタックレジスタの内容へのアクセス

X、Y、Z、T の 4 つのスタックレジスタに保持されている値は、RPN モードの方程式、またはプログラムの REGX、REGY、REGZ、REGT コマンドで呼び出すことができます。

これらの命令を使用するには、まず **[EQN]** を押します。**[R↓]** を押すと、ディスプレイに X、Y、Z、T レジスタを示すメニューが表示されます。**[>]** または **[<]** を押してアンダーライン記号を移動し、どのレジスタが現在選択されているかを示します。**[ENTER]** を押して、今後使用するため選択したスタックレジスタの値を呼び出す命令を、プログラムまたは方程式に入れます。これらは、REGX、REGY、REGZ、REGT と表示されます。

例えば、最初に **[EQN]** を押し、命令 REGX x REGY x REGZ x REGT を入力することによって入力されるプログラム行は、4 つのスタックレジスタ内の値の積を計算し、結果を X レジスタに格納します。X、Y、Z レジスタの前の数値がそれぞれ、Y、Z、T スタックレジスタに残ります。

HP35s でスタック内の値を効率良くさまざまに活用するには、この方法を使用する必要があります。

ALG: まとめ

ALG について

この付録では、次の ALG モード固有の機能についてまとめます。

- 2つの引数をとる演算
- 指数関数と対数関数 ( 10^x 、 LOG、 e^x 、 LN)
- 三角関数
- 数値の部分
- スタックの確認
- 複素数の計算
- 方程式の積分
- 基数 2、8、16 の四則演算
- 2変数の統計データの入力

MODE **4** (4ALG) を押して、電卓を ALG モードに設定します。電卓が ALG モードにある場合、ALG インジケータがオンになります。

ALG モードの演算は、次の順番で行われます。

1. 括弧の中の式
2. 階乗 (!) 関数。  を押す前に数値を入力する必要があります。
3. 関数キーの後に数値を入力する必要がある関数。例えば、COS、SIN、TAN、ACOS、ASIN、ATAN、LOG、LN、 x^2 、 $1/x$ 、 $\sqrt{x}$ 、 π 、 $\sqrt[3]{x}$ 、%、RND、RAND、IP、FP、INTG、SGN、nPr、nCr、%CHG、INT÷、Rmdr、ABS、 e^x 、 10^x 、単位変換。
4. $\sqrt[y]{x}$ と y^x 。

5. 単項マイナス +/-

6. $\times$, $\div$

7. $+$, $-$

8. $=$

ALG モードにおける 2 つの引数をとる演算

ALG を使用した演算の説明では、ALG モードによって影響を受ける次の部分を取り上げます。2 つの引数をとる演算は、ALG モードによって影響を受けます。

- 単純な四則計算
- 累乗関数 (y^x 、 $\sqrt[y]{x}$)
- 百分率計算 ($\%$ または $\boxed{\text{2nd}} \boxed{\text{\%CHG}}$)
- 順列と組合せ ($\boxed{\text{2nd}} \boxed{\text{nCr}}$, $\boxed{\text{2nd}} \boxed{\text{nPr}}$)
- 除算の商と剰余 ($\boxed{\text{2nd}} \boxed{\text{INTG}} \boxed{\text{2}} (\text{2INTG}\div)$, $\boxed{\text{2nd}} \boxed{\text{INTG}} \boxed{\text{3}} (\text{3Rmdr})$)

単純な四則計算

簡単な四則計算の例をいくつか挙げます。

ALG モードでは、最初の数値を入力して、演算子 ($+$ 、 $-$ 、 $\times$ 、 $\div$) を押し、2 番目の数値を入力して、最後に $\boxed{\text{ENTER}}$ キーを押します。

計算内容：

$12 + 3$

$12 - 3$

12×3

$12 \div 3$

以下のキーを押します。

$\boxed{\text{1}} \boxed{\text{2}} \boxed{+} \boxed{\text{3}} \boxed{\text{ENTER}}$

$\boxed{\text{1}} \boxed{\text{2}} \boxed{-} \boxed{\text{3}} \boxed{\text{ENTER}}$

$\boxed{\text{1}} \boxed{\text{2}} \boxed{\times} \boxed{\text{3}} \boxed{\text{ENTER}}$

$\boxed{\text{1}} \boxed{\text{2}} \boxed{\div} \boxed{\text{3}} \boxed{\text{ENTER}}$

ディスプレイ：

$12+3$

15.0000

$12-3$

9.0000

12×3

36.0000

$12\div 3$

4.0000

累乗関数

ALG モードで数値 y の x 乗を計算するには、 y y^x x を入力し、 [ENTER] を押し
ます。

計算内容：	以下のキーを押します。	ディスプレイ：
12^3	$\text{[1] [2] } y^x \text{ [3] [ENTER]}$	12^3 1,728.0000
$64^{1/3}$ (立方根)	$\text{[6] } \frac{1}{y} \text{ [3] [>] [6] [4] [ENTER]}$	$\text{XROOT}(3,64)$ 4.0000

百分率計算

百分率機能 [%] キーは、数値を 100 で除算します。

計算内容：	以下のキーを押します。	ディスプレイ：
200 の 27%	$\text{[2] [%] [2] [0] [0] [>] [2] [7] [ENTER]}$	%(200,27) 54.0000
200 の 27% 減	$\text{[2] [0] [0] [-] [2] [%] [2] [0] [0] [>] [2] [7] [ENTER]}$	$200 - \text{%(200,27)}$ 146.0000
25 の 12% 増	$\text{[2] [5] [+] [2] [%] [2] [5] [>] [1] [2] [ENTER]}$	$25 + \text{%(25,12)}$ 28.0000

計算内容：	以下のキーを押します。
y の $x\%$	$\text{[2] [%] } y \text{ [>] } x \text{ [ENTER]}$
y から x への変化率 ($y \neq 0$)	$\text{[6] [%CHG] } y \text{ [>] } x \text{ [ENTER]}$

例：

昨年、価格が \$16.12 だった商品が今年は \$15.76 となりました。昨年から今年
の価格の変化率はいくらでしょうか。

キー：	ディスプレイ：	説明：
$\text{[6] [%CHG] [1] [6] [.] [1] [2] [>] [1] [5] [.] [7] [6] [ENTER]}$	$\text{[%CHG}(16.12,15.7$ -2.2333	今年の価格は昨年の価格 から、約 2.2% 下落しま した。

順列と組合せ

例：従業員の組合せ

女性従業員 14 名、男性従業員 10 名の会社で、6 名から成る安全委員会を設立中です。異なる組合せがいくつ可能でしょうか。

キー：	ディスプレイ：	説明：
 nCr 2 4 >	nCr(24,6)	可能な組合せの総数。
6 ENTER	134,596.0000	

除算の商と剰余

 **INTG** **2**(2INTG÷) と  **INTG** **3**(3Rmdr) を使用して、2 個の整数の除算から商または剰余を求めることができます。

 **INTG** **2**(2INTG÷) 整数 **1** **>** 整数 **2** **ENTER**

 **INTG** **3**(3Rmdr) 整数 **1** **>** 整数 **2** **ENTER**

例：

58 ÷ 9 の商と剰余を表示します

キー：	ディスプレイ：	説明：
 INTG 2 (2INTG÷)	IDIV(58,9)	商を表示します。
5 8 > 9 ENTER	6.0000	
 INTG 3 (3Rmdr)	RMDR(58,9)	剰余を表示します。
5 8 > 9 ENTER	4.0000	

括弧を含む計算

さらに数値を入力するまで、中間結果を計算させないようにするには、括弧を使用します。例えば、次の式を計算する場合を考えます。

$$\frac{30}{85-12} \times 9$$

3 **0** **÷** **8** **5** **−** **1** **2** **×** **9** と入力すると、その結果は -107.6471 となります。これは、希望する計算とは異なります。85 から 12 を差し引くまで除算しないようにするには、括弧を使用します。

キー：	ディスプレイ：	説明：
3 0 ÷ () 8 5 −	30÷(85−)	計算は実行されません。
1 2 >	30÷(85−12) _	85 − 12 を計算します。
× 9	30÷(85−12)×9 _	30/73 を計算します。
ENTER	30÷(85−12)×9 3.6986	30/(85 − 12) × 9 を 計算します。

左括弧の前の乗算記号 (×) は省略可能です。方程式モードでは、暗黙の乗算は使用できません。例えば、式 $2 \times (5 - 4)$ は、**2** **()** **5** **−** **4** のように、2 と左括弧の間に **×** キーを挿入しない形で入力できます。

指数関数と対数関数

計算内容：	以下のキーを押します。	ディスプレイ：
自然対数（底 e）	$\boxed{\text{LN}} \boxed{1} \boxed{\text{ENTER}}$	LN(1) 0.0000
常用対数（底 10）	$\boxed{\text{LOG}} \boxed{1} \boxed{0} \boxed{\text{ENTER}}$	LOG(10) 1.0000
e を底とする指数	$\boxed{\text{e}^x} \boxed{2} \boxed{\text{ENTER}}$	EXP(2) 7.3891
10 を底とする指数（真数）	$\boxed{10^x} \boxed{2} \boxed{\text{ENTER}}$	ALOG(2) 100.0000

三角関数

角度の単位が $\boxed{\text{MODE}} \boxed{1}$ (1 DEG) であるとします。

計算内容：	以下のキーを押します。	ディスプレイ：
x の正弦	$\boxed{\text{SIN}} \boxed{3} \boxed{0} \boxed{\text{ENTER}}$	SIN(30) 0.5000
x の余弦	$\boxed{\text{COS}} \boxed{6} \boxed{0} \boxed{\text{ENTER}}$	COS(60) 0.5000
x の正接	$\boxed{\text{TAN}} \boxed{4} \boxed{5} \boxed{\text{ENTER}}$	TAN(45) 1.0000
x の正弦の逆関数	$\boxed{\text{ASIN}} \boxed{1} \boxed{\text{ENTER}}$	ASIN(1) 90.0000
x の余弦の逆関数	$\boxed{\text{ACOS}} \boxed{0} \boxed{\text{ENTER}}$	ACOS(0) 90.0000
x の正接の逆関数	$\boxed{\text{ATAN}} \boxed{0} \boxed{\text{ENTER}}$	ATAN(0) 0.0000

双曲線関数

計算内容：	以下のキーを押します。
x の双曲線正弦 (SINH)	 HYP SIN 、数値を入力、 ENTER を押します
x の双曲線余弦 (COSH)	 HYP COS 、数値を入力、 ENTER を押します
x の双曲線正接 (TANH)	 HYP TAN 、数値を入力、 ENTER を押します
x の双曲線正弦の逆関数 (ASINH)	 HYP  ASIN 、数値を入力、 ENTER を押します
x の双曲線余弦の逆関数 (ACOSH)	 HYP  ACOS 、数値を入力、 ENTER を押します
x の双曲線正接の逆関数 (ATANH)	 HYP  ATAN 、数値を入力、 ENTER を押します

数値の部分

計算内容：	以下のキーを押します。	ディスプレイ：
2.47 の整数部	 INTG 6 (6IP) 2 . 4 7 ENTER	IP(2.47) 2.0000
2.47 の小数部	 INTG 5 (5FP) 2 . 4 7 ENTER	FP(2.47) 0.4700
-7 の絶対値	 ABS +/- 7 ENTER	ABS(-7) 7.0000
9 の符号値	 INTG 1 (1SGN) 9 ENTER	SGN(9) 1.0000
-5.3 以下の最大整数	 INTG 4 (4INTG) +/- 5 . 3 ENTER	INTG(-5.3) -6.0000

スタックの確認

 または   キーを押すと、X レジスタ、Y レジスタ、Z レジスタ、T レジスタのメニューが表示され、スタック全体の内容を確認できます。 と   キーでは、ディスプレイにおけるアンダーラインの位置が異なります。  を押すと、T レジスタにアンダーラインが表示され、 を押すと、Y レジスタにアンダーラインが表示されます。

 を押すと、次のメニューが表示されます。

X Y Z T

値

  を押すと、次のメニューが表示されます。

X Y Z T

値

 と   を ( または  と一緒に) 押すと、スタック全体の内容を確認し、呼び出すことができます。スタックのどの部分を呼び出したかに応じて、REGX、REGY、REGZ、または REGT として表示され、次の計算に利用できます。

ALG モードにおける X、Y、Z、T レジスタの値は、RPN モードの場合と同じです。通常の計算、SOLVE、プログラム実行、積分実行の後の 4 つのレジスタの値は、RPN モードでも ALG モードでも同じで、ALG と RPN の論理モードを切り替えたときに保持されます。

方程式の積分

1. 方程式を入力します（第 6 章「方程式リストへの方程式の入力」を参照してください）。方程式モードを終了します。
2. 積分の範囲を入力します。下限を入力し、 を押して、上限を入力します。
3. 方程式を表示します。 を押し、必要に応じて方程式リストをスクロールし ( または  を押します)、目的の方程式を表示します。
4. 積分の変数を選択します。  変数を押します。これにより計算が開始します。

複素数の計算

複素数を入力するには：

形式： $x \cdot i^y$

1. 実数部を入力します。
2. **[i]** を押します。
3. 虚数部を入力します。

形式： $x + y \cdot i$

1. 実数部を入力します。
2. **[+]** を押します。
3. 虚数部を入力します。
4. **[i]** を押します。

形式： $r \theta a$

1. r の値を入力します。
2. **[$\angle$]** **[θ]** を押します。
3. θ の値を入力します。

1 個の複素数を持つ演算を実行するには：

1. 関数を選択します。
2. 複素数 z を入力します。
3. **[ENTER]** を押して計算します。
4. 第 2 行に計算結果が表示されます。表示形式は **[MODE]** で設定した形式になります。

2 個の複素数を持つ四則演算を実行するには：

1. 最初の複素数 z_1 を入力します。
2. 四則演算を選択します。
3. 第 2 の複素数 z_2 を入力します。
4. **[ENTER]** を押して計算します。
5. 第 2 行に計算結果が表示されます。表示形式は **[MODE]** で設定した形式になります。

複素数の例をいくつか挙げます。

例：

$\sin(2 + 3i)$ を評価します

キー：	ディスプレイ：	説明：
 DISPLAY 9 ($9 \times i$)		表示形式を設定します。
SIN 2 + 3 i	$\text{SIN}(2+3i)$	
ENTER	$\text{SIN}(2+3i)$	結果は
	$9.1545i-4.1689$	$9.1545i-4.1689$

例：

式を評価します

$$z_1 \div (z_2 + z_3),$$

ただし、 $z_1 = 23 + 13i$, $z_2 = -2 + i$, $z_3 = 4 - 3i$

キー：	ディスプレイ：	説明：
 DISPLAY . 1		表示形式を設定します
$(10 \times + \times i)$		
()	$\leftarrow i \div (-2 + i + 4 - 3i)$	
2 3 + 1 3 i		
> ÷ () +/- 2 +		
i + 4 - 3 i		
ENTER	$(23+13i) \div (-2+...)$	結果は
	$2.5000+9.0000i$	$2.5000 + 9.0000i$

例:

$(4 - 2/5) \div (3 - 2/3)$ を評価します。

キー:	ディスプレイ:	説明:
() 4 - . 2 .	$\leftarrow 5 \div \times (3 - 2/3)$	
5 i > x () 3		
- . 2 . 3 i		
ENTER	$(4 - 2/5) \div (3 - 2/3)$ 11.7333 $\div$ 3.8667	結果は 11.7333 $\div$ 3.8667

基数 2、8、16 の四則演算

16 進数、8 進数、2 進数モードでの四則演算の例をいくつか挙げます。

例:

キー:	ディスプレイ:	説明:
MODE BASE 2 (2HEX)		基数 16 を設定します。 HEX インジケータがオンになります。
1 2 RCL F MODE BASE 6 (6h) + RCL E 9 RCL A MODE BASE 6 (6h) ENTER	12Fh+E9Ah FC9h	結果。
MODE BASE 3 (3OCT)		8 進数モードにします。 OCT インジケータがオンになります。
7 7 6 0 MODE BASE 7 (7o) - 4 3 2 6 MODE BASE 7 (7o) ENTER	7760o-4326o 3432o	表示数値を 8 進数に変換します。

$$100_8 \div 5_8 = ?$$

1 0 0 BASE 7 100₈ ÷ 5₈
 (7₈) 5
 BASE 7 (7₈) ENTER

結果の整数部。

14₈

$$5A0_{16} + 10011000_2 = ?$$

BASE 2 (2HEX)

5A0h +

16 進数モードにしま

5 A 0

す。HEX インジケータ

BASE 6 (6h) +

がオンになります。

1 0 0 1 1 0 0 ← A0h + 10011000₂

0 BASE 8 (8b)

ENTER

5A0h + 10011000₂

16 進数での結果。

638h

BASE 1 (1DEC)

5A0h + b10011000₂

10 進数に戻します。

1,592,000₁₀

2 変数の統計データの入力

ALG モードでは、(x, y) ペアを逆順 (y x または y ENTER x) に入力して、y が Y レジスタに、x が X レジスタに格納されるようにします。

1. CLEAR 4 (4Σ) を押して、既存の統計データをクリアします。
2. y 値を最初に入力し、 を押します。
3. 対応する x の値を入力し、 を押します。
4. ディスプレイには蓄積した統計データペアの数 n が表示されます。
5. x, y ペアの入力を続けます。入力のたびに n が更新されます。

入力直後の間違った値を削除するには、 を押します。間違った統計データを削除した後、電卓は、入力された最後の統計データを第 1 行（ディスプレイの一番上の行）に、n を第 2 行に表示します。統計データがない場合は、第 2 行に n=0 と表示します。

例：

次の表の左側の x, y 値を入力してから、右側のように訂正します。

x, y の初期値	x, y の訂正值
20, 4	20, 5
400, 6	40, 6

キー：

ディスプレイ：

説明：

CLEAR **4** (4Σ)

既存の統計データをクリアします。

X↔Y **2** **0** **Σ+**

20 Σ+
1.0000

最初の新しいデータペアを入力します。

6 **X↔Y** **4** **0** **0**

入力したデータペアの数 n が表示されます。

Σ+

400 Σ+
2.0000

LASTx

LASTx
400.0000

最後の x 値を呼び出します。最後の y は、Y レジスタに残っています。

Σ-

400 Σ-
1.0000

最後のデータペアを削除します。

6 **X↔Y** **4** **0** **Σ+**

40 Σ+
2.0000

最後のデータペアを再入力します。

4 **X↔Y** **2** **0** **Σ-**

20 Σ-
1.0000

最初のデータペアを削除します。

Σ-

5 **X↔Y** **2** **0** **Σ+**

20 Σ+
2.0000

最初のデータペアを再入力します。
統計レジスタには合計 2 組のデータペアがあります。

線形回帰

線形回帰 L.R. (線形推定とも呼ばれます) は、 x, y データセットに一番適合する直線を求めるための統計手法です。

- x (または y) の推定値を求めるには、 y (または x) の所定仮想値を入力し、 **ENTER**、次に **L.R.** ($\hat{x}$) (または **L.R.** **Y** ($\hat{y}$)) を押します。
- データに一番適合する線を定義する値を求めるには、 **L.R.** を押し、次に (r)、(m)、または (b) を押します。

SOLVE の詳細

この付録では、第 7 章で説明した SOLVE 操作についてさらに詳しく説明します。

SOLVE が根を見つける方法

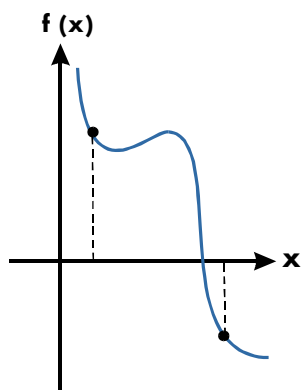
SOLVE は最初に、方程式で未知の変数を直接解くことを試みます。それができなかった場合は、SOLVE が反復（繰り返し）プロセスに変わります。**反復**操作では、指定した方程式が繰り返し実行されます。方程式が返す値は、未知の変数 x の関数 $f(x)$ です（ $f(x)$ は、未知の変数 x を使って数学的に簡素化して表現された関数です）。SOLVE は、未知変数 x の推定値を関数 $f(x)$ に当てはめ、関数を繰り返し実行することでその推定値を洗練させていくという方法をとります。

関数 $f(x)$ の 2 つの連続する推定値が異なる符号を持てば、SOLVE は、関数 $f(x)$ が 2 つの推定値間において x 軸と 1 箇所以上で交差したと推定します。根が発見できるまで、この間隔を規則的に狭めていきます。

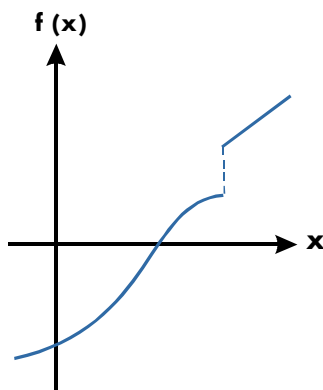
SOLVE が根を発見するためには、根は、電卓が認識できる数値の範囲に存在している必要があります。また、反復検索が発生する場所に、関数が数学的に定義されている必要があります。根がオーバーフロー境界の内側に存在する場合、次の条件の 1 つ以上に適合すれば、SOLVE は必ず根を発見します。

- 2 つの推定値が異なる符号を持つ $f(x)$ 値を生成し、関数のグラフが、これらの推定値間で x 軸と 1 箇所以上で交差している（図 a）。
- x の増加に伴い、 $f(x)$ が常に増加または減少する（図 b）。
- $f(x)$ のグラフがすべての領域で凸型か凹型である（図 c）。

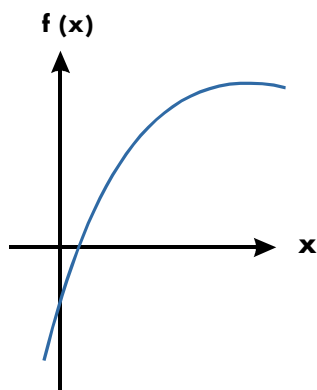
- $f(x)$ に 1 つ以上の極小値または極大値がある場合、それぞれが $f(x)$ の隣接根間に単独で発生している (図 d)。



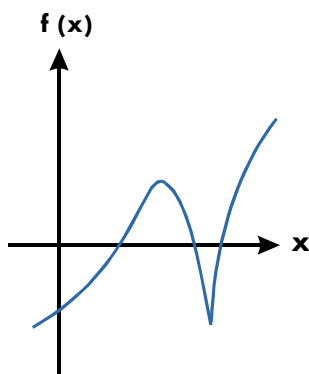
a



b



c



d

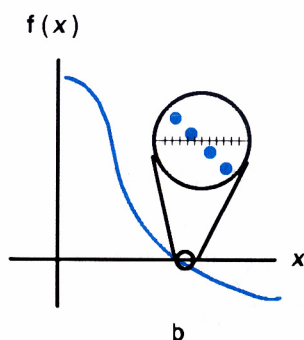
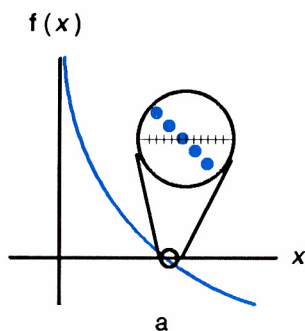
Function Whose Roots Can Be Found

ほとんどの場合、計算で求めた根は、方程式の非常に高精度の理論的な根の、正確な推定値になります。理想の解は、 $f(x) = 0$ の解です。しかし通常は、 $f(x)$ の非常に小さいゼロ以外の値も許容されます。原因が、精度が 12 桁に制限された状態で数値を近似することにあるからです。

結果の解釈

SOLVE は、次のどちらかの条件下で解を生成します。

- $f(x)$ がゼロに等しくなる推定値を見つけた場合（図 a を参照してください）。
- $f(x) = 0$ でない推定値であるが、計算された根が、関数のグラフが x 軸と交差する場所に隣接する 12 桁数値である場合（図 b を参照してください）。これは、2 つの最終推定値が隣り合っており（12 桁目の数値の差が 1 である状態）、一方の推定値に対する関数の値が正で、もう一方に対する値が負であるときに発生します。すなわち、 $(0, 10^{-499})$ または $(0, -10^{-499})$ です。ほとんどの場合、 $f(x)$ は相対的にゼロに近い値になります。



- ✓ 結果に関する追加情報を取得するには、**[R↓]** を押して、Y レジスタに残されていた、根 (x) の直前の推定値を確認します。**[R↓]** を再度押して、Z レジスタに残されていた、 $f(x)$ の値を確認します。 $f(x)$ の値がゼロか、または相対的に十分小さいものであれば、無事に解を発見したとみなすことができます。これに対し、 $f(x)$ の値が相対的に大きかった場合、結果の解釈には注意を要します。

例：根が 1 つの方程式

方程式の根を求めます。

$$-2x^3 + 4x^2 - 6x + 8 = 0$$

方程式を式として入力します。

キー:

ディスプレイ:

説明:

EQN
 $\frac{+}{-}$ 2 $\times$
 RCL X y^x 3
 $+$ 4 $\times$
 RCL X y^x 2
 $-$ 6 $\times$ RCL X
 $+$ 8 ENTER
 $\rightarrow$ SHOW
 C

$-2 \times X^3 + 4 \times X^2 - 6 \Rightarrow$
 CK=B9AD
 LN=18

方程式モードを選択します。
 方程式を入力します。

チェックサムと長さ。

方程式モードをキャンセルします。

方程式を解いて根を求めます。

キー:

ディスプレイ:

説明:

0 $\rightarrow$ STO X
 ENTER 1 0
 EQN
 $\rightarrow$ SOLVE X

10_{-}
 $-2 \times X^3 + 4 \times X^2 - 6 \Rightarrow$
 SOLVING
 $X=$
 1.6506
 1.6506
 $-4.00000E-11$

根の初期推定値。

方程式モードを選択し、方程式の左端を表示します。
 X を求め、結果を表示します。

最後の 2 つの推定値は、小数点以下 4 桁まで同じです。
 $f(x)$ は非常に小さいので、近似値は良い根です。

例: 根が 2 つの方程式

放物型方程式の 2 つの根を求めます。

$$x^2 + x - 6 = 0.$$

方程式を式として入力します。

キー:

ディスプレイ:

説明:

EQN

方程式モードを選択します。

RCL **X** **y^x** **2** **+**

方程式を入力します。

RCL **X** **-** **6**

$X^2 + X - 6$

ENTER

⇐ **SHOW**

CK=3971

チェックサムと長さ。

LN=7

C

方程式モードをキャンセルします。

方程式を解いて、正の根と負の根を求めます。

キー:

ディスプレイ:

説明:

0 **⇐** **STO** **X**

10_

正の根の初期推定値。

ENTER **1** **0**

EQN

$X^2 + X - 6$

方程式モードを選択し、方程式を表示します。

⇐ **SOLVE** **X**

SOLVING

推定値 0 と 10 で正の根を計算

$X =$

2.0000

します。

2.0000

最後の 2 つの推定値は同じです。

✓ **R↓**

✓ **R↓** **⇐** **SHOW**

0.000000000000 $f(x) = 0.$

0 **⇐** **STO** **X**

-10_

負の根の初期推定値。

ENTER **1** **0** **+/-**

EQN

$X^2 + X - 6$

方程式を再表示します。

⇐ **SOLVE** **X**

SOLVING

推定値 0 と -10 で負の根を計算

$X =$

-3.0000

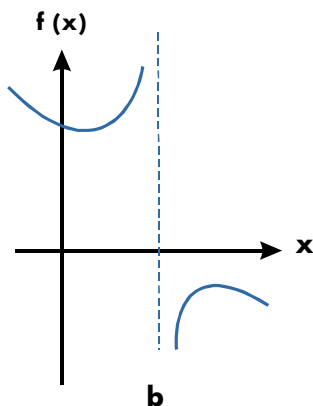
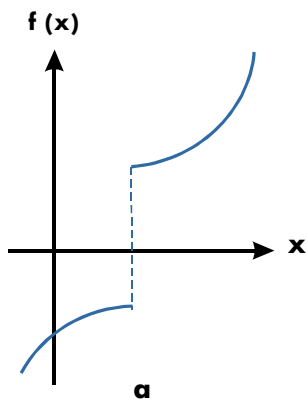
します。

✓ **R↓** **R↓** **⇐** **SHOW**

0.000000000000 $f(x) = 0.$

次のような場合は注意してください。

- 関数のグラフに x 軸と交差する不連続部があり、SOLVE 演算が不連続部に隣接する値を返す場合（図 a を参照してください）。この場合、 $f(x)$ は相対的に大きくなります。
- $f(x)$ の値が、グラフの符号が変化する点で無限大に向かっている可能性のあるとき（図 b を参照してください）。この状態は極と呼ばれています。SOLVE は、隣り合う 2 つの x 値間に符号の変化があると判断し、可能性のある根を返します。しかし、 $f(x)$ の値は相対的に大きくなります。もし極が、12 桁精度で表現できる x の値で発生しているときは、計算が停止し、エラーメッセージが表示されます。



Special Case: A Discontinuity and a Pole

例：不連続な関数

方程式の根を求めます。

$$IP(x) = 1.5$$

方程式を入力します。

キー：

ディスプレイ：

説明：

EQN

INTG **6** (6IP)

RCL **X** **>** **=**

1 **.** **5** **ENTER**

IP(X)=1.5

方程式モードを選択します。
方程式を入力します。

SHOW

CK=D2C1
LN=9

チェックサムと長さ。

C

方程式モードをキャンセル
します。

方程式を解いて根を求めます。

キー:

ディスプレイ:

説明:

0 **STO** **X**

5_

根の初期推定値。

ENTER **5**

EQN

IP(X)=1.5

方程式モードを選択し、方程式
を表示します。

SOLVE **X**

SOLVING

推定値 0 と 5 を使用して根を求
めます。

X=

2.0000

SHOW

1.9999999999

根を小数点以下 11 桁まで表示
します。

✓ **R↓** **SHOW**

2.000000000000

直前の推定値は、わずかに大き
くなります。

✓ **R↓**

-0.5000

f(x) は相対的に大きくなります。

最終の 2 つの推定値に差がある点と、 $f(x)$ の値が相対的に大きい点に注目
してください。問題は、 $f(x)$ がゼロになる x の値がないことです。しかし
 $x = 1.9999999999$ に、 $f(x)$ を反対の符号にする x の隣接値が存在します。

例:

方程式の根を求めます。

$$\frac{x}{x^2 - 6} - 1 = 0$$

x が $\sqrt{6}$ に近づくと、 $f(x)$ は、非常に大きい正または負の数値になります。

方程式を式として入力します。

キー:

ディスプレイ:

説明:

EQN
RCL **X** **÷** **()**
RCL **X** **y^x** **2**
← **6** **>**
← **1** **ENTER**

方程式モードを選択します。
方程式を入力します。

↵ **SHOW**

$X \div (X^2 - 6) - 1$
CK=7358
LN=11

チェックサムと長さ。

C

方程式モードをキャンセル
します。

方程式を解いて根を求めます。

キー:

ディスプレイ:

説明:

2 **·** **3** **↵**
STO **X** **ENTER**
2 **·** **7**
EQN
↵ **SOLVE** **X**

2.7_

根の初期推定値。

$X \div (X^2 - 6) - 1$

方程式モードを選択し、方程式を表示します。
 $f(x)$ の根は見つかりません。

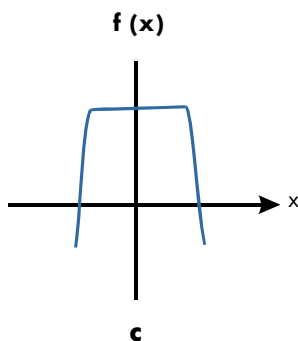
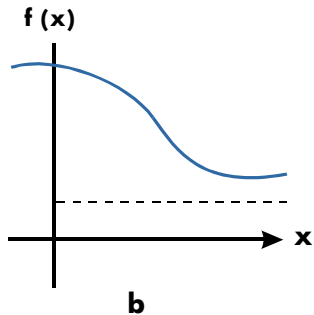
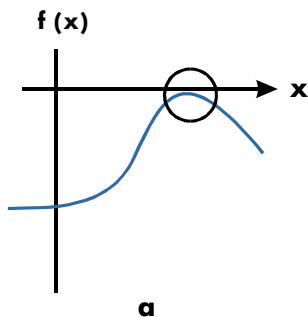
NO ROOT FND

SOLVE が根を発見できなかったときは

SOLVE は常に根を発見できるわけではありません。次のような場合にはメッセージ NO ROOT FND が表示されます。

- 極小値または極大値の付近で根の検索が終了してしまう場合（図 a を参照してください）。
- $f(x)$ が広範囲の x に対して基本的にならざるエリアである、水平漸近線上で SOLVE が動作しているため、検索が途中終了してしまう場合（図 b を参照してください）。
- 検索が、関数の「平坦な」局所領域に集中している場合（図 c を参照してください）。

これらの場合、スタックの値が、SOLVE 実行前の値と同じになります。



Case Where No Root Is Found

例：極小

放物型方程式の根を計算します。

$$x^2 - 6x + 13 = 0.$$

$x = 3$ で最小値になります。

方程式を式として入力します。

キー：

ディスプレイ：

説明：

[EQN]

方程式モードを選択します。

[RCL] [X] y^x [2]

方程式を入力します。

[−] [6] [×] [RCL] [X]

[+] [1] [3] [ENTER]

$X^2 - 6X + 13$

[SHOW]

CK=EC74
LN=10

チェックサムと長さ。

[C]

方程式モードをキャンセルします。

方程式を解いて根を求めます。

キー:

ディスプレイ:

説明:

[0] **[$\frac{\square}{\square}$]** **[STO]** **[X]**

根の初期推定値。

[ENTER] **[1]** **[0]**

10_

[EQN]

$X^2 - 6X + 13$

方程式モードを選択し、方程式を表示します。

[$\frac{\square}{\square}$] **[SOLVE]** **[X]**

NO ROOT FND

推定値 0 と 10 を使用すると、検索に失敗します。

例：漸近線

方程式の根を求めます。

$$10 - \frac{1}{X} = 0$$

方程式を式として入力します。

キー:

ディスプレイ:

説明:

[EQN]

方程式モードを選択します。

[1] **[0]** **[$\frac{\square}{\square}$]** **[1/x]**

方程式を入力します。

[RCL] **[X]** **[ENTER]**

10-INV(X)

[$\frac{\square}{\square}$] **[SHOW]**

チェックサムと長さ。

LN=9

[C]

方程式モードをキャンセルします。

[.] **[0]** **[0]** **[5]** **[$\frac{\square}{\square}$]**

根の正の推定値。

[STO] **[X]** **[ENTER]** **[5]**

5_

[EQN]

10-INV(X)

方程式モードを選択し、方程式を表示します。

[$\frac{\square}{\square}$] **[SOLVE]** **[X]**

X=

推定値 0.005 と 5 を使用して x を求めます。

0.1000

直前の推定値は同じです。

0.1000

0.000000000000 $f(x) = 0$

D-10 SOLVE の詳細

推定値に負の値を使用するとどうなるか確認します。

キー:	ディスプレイ:	説明:
$\pm/\square$ 1 $\rightarrow$ STO X	-1.0000	根の負の推定値。
ENTER		
$\pm/\square$ 2 EQN	10-INV(X)	方程式モードを選択し、方程式を表示します。
$\rightarrow$ SOLVE X	X= 0.1000	Xを求め、結果を表示します。

例：方程式の根を求めます。

$$\sqrt{[x \div (x + 0.3)]} - 0.5 = 0$$

方程式を式として入力します。

キー:	ディスプレイ:	説明:
EQN		方程式モードを選択します。
$\sqrt{\square}$ RCL X $\div$ ()		方程式を入力します。
RCL X + . 3		
> > - . 5		
ENTER	SQRT(X÷(X+0.3))➡	
$\leftarrow$ SHOW	CK=9F3B LN=19	チェックサムと長さ。
C		方程式モードをキャンセルします。

最初に正の根を探します。

キー:	ディスプレイ:	説明:
0 $\rightarrow$ STO X		根の正の推定値。
ENTER 1 0	10_	
EQN	SQRT(X÷(X+0.3))➡	方程式モードを選択し、方程式の左端を表示します。
$\rightarrow$ SOLVE X	X= 0.1000	推定値 0 と 10 で根を計算します。

推定値 0 と -10 で負の根を探します。関数は、0 から -0.3 の間の x 値に対して未定義です。これらの値は分母を正、分子を負にするため、負の平方根になるからです。

キー:	ディスプレイ:	説明:
	-10_	
	SQRT(X?(X+0.3))	方程式モードを選択し、方程式の左端を表示します。
	NO ROOT FND	f(x) の根は見つかりません。

例：局所で平坦な関数

関数の根を求めます。

$f(x) = x + 2$ if $x < -1$,
 $f(x) = 1$ for $-1 \leq x \leq 1$ (局所で平坦な領域),
 $f(x) = -x + 2$ if $x > 1$.

RPN モードでは、この関数をプログラムとして入力します。

```
J001 LBL J
J002 1
J003 2
J004 RCL+ X
J005 x<y?
J006 RTN
J007 4
J008 -
J009 +/-
J010 x>y?
J011 R↓
J012 RTN
```

チェックサムと長さ：9412 39

初期推定値 10^{-8} と -10^{-8} を使用して X を求めます。

キー：
(RPN モード)

ディスプレイ：

説明：

1 **E** **8**

推定値を入力します。

+/- **STO** **X** **1**

$-1E-8$

+/- **E** **8** **+/-**

↵ **FN=** **J**

$-1.00000E-8$

関数としてプログラム "J" を選択
します。

↵ **SOLVE** **X**

$\times=$

X を求め、結果を表示します。

-2.00000

丸め誤差

電卓の有効桁が 12 桁で有限なので、丸めによって誤差が発生することがあります。また、丸めは、SOLVE と積分の繰り返し計算で増幅されることもあります。例えば、

$$[(|x|+1)+10^{15}]^2 \cdot 10^{30} = 0$$

には根がありません。 $f(x)$ が常にゼロより大きいからです。しかし初期推定値 1 と 2 を指定すると、SOLVE は、丸め誤差により 1.0000 を返します。

丸め誤差により、根が発見できなくなることもあります。方程式

$$|x^2 - 7| = 0$$

は、 $\sqrt{7}$ に根があります。しかし 12 桁の数値が $\sqrt{7}$ と正確に等しくなることは
ないため、電卓は関数をゼロにできません。さらに、関数が符号を変化させる
こともないため、SOLVE がメッセージ NO ROOT FND を返します。

積分の詳細

この付録では、第 8 章で説明した積分についてさらに詳しく説明します。

積分の評価方法

積分操作 $\int f(x) dx$ で使用されるアルゴリズムは、積分区間内の多数の x 値（サンプル点と呼ばれます）に対する関数の値を加重平均することで、関数 $f(x)$ の積分を計算します。こうしたサンプリング処理の結果の確度は、考慮するサンプル点の数に依存します。通常、サンプル点が多いほど、確度が高くなります。 $f(x)$ を無限数のサンプル点で評価できれば、アルゴリズムは、計算される関数 $f(x)$ 内の不確定値による限界を無視して、常に正確な答えを提供できます。

無限のサンプル点での関数の評価には、無限の時間がかかってしまいます。しかし、計算される積分の最大確度は計算される関数値の確度によって制限されるため、これは不要です。有限のサンプル点のみでも、 $f(x)$ の固有の不確定値を考慮した場合と同じ確度で積分を計算できます。

この積分アルゴリズムでは最初に、2 ～ 3 個のサンプル点のみを使ってそれほど確度の高くない近似値を求めます。もし近似値の確度が $f(x)$ の許容確度ほど高くない場合、サンプル点を増やして繰り返し計算を行います。結果の近似値の確度が $f(x)$ の固有の不確定値を考慮した場合の確度と同じになるまで、サンプル点を約 2 倍ずつ増加しながら反復を続行します。

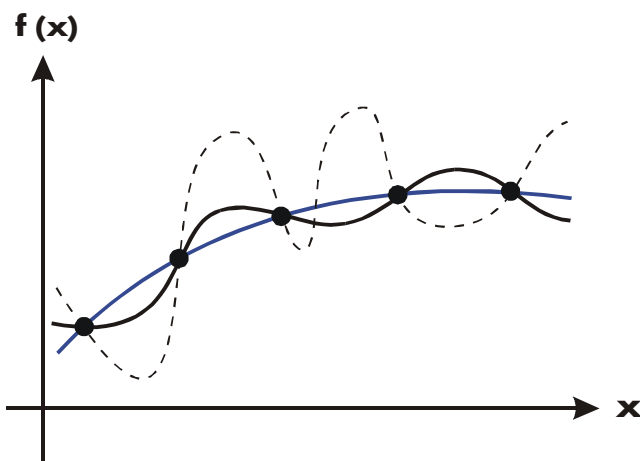
第 8 章で説明したように、最終的な近似値の不確定値は表示形式から導出された数値で、これにより関数の不確定値が決まります。それぞれの繰り返しの最後で、アルゴリズムは、最後に得られた近似値と、その直前の 2 つの反復中に計算された近似値を比較します。これら 3 つの近似値それぞれと、残り 2 つの近似値との差が、最後の近似値に対する不確定値の許容誤差より小さくなったら、積分計算を終了します。このとき、最後の近似値が X レジスタに、不確定値が Y レジスタに格納されます。

3 つの連続する近似値それぞれの誤差（厳密解と近似値の差）がすべて、近似値間の差異より大きくなることは、ほとんどありません。したがって、最後の近似値の誤差は不確定値よりも小さくなります（ $f(x)$ には急激な変化がないものとします）。最後の近似値の誤差を計算することはできませんが、その誤差は、近似値の表示された不確定値を超えないと見なすことができます。すなわち、Y レジスタに格納された不確定値は、得られた近似値と厳密解との差の「最大値」と考えることができます。

不正確な結果を引き起こす条件

HP 35s に内蔵の積分アルゴリズムは、最良のアルゴリズムの 1 つですが、他の数値積分アルゴリズムと同様、特定の条件で不正確な結果を出力します。**ただし、その条件はほとんど発生しません。**このアルゴリズムは、ほぼすべての**スムーズな**関数で正確な結果を出力するように設計されています。**急激に変化する関数**の場合のみ、不正確な結果を出力するリスクが高くなります。こうした関数を実際の物理学的な状況に関連する問題で発生することは稀であり、発生しても、計算前に認識し、簡単な方法で対処できます。

アルゴリズムが $f(x)$ について認識できるのは各サンプル点における関数値のみであり、 $f(x)$ と、 $f(x)$ とすべてのサンプル点で一致するその他の関数を区別することができません。この状況を図に示します。示された 3 つの関数のグラフ（積分区間の一部）には、共通のサンプル点が多数含まれています。



図のサンプル点を使って、アルゴリズムは、すべての関数の積分で同じ近似値を計算します。青と黒の実線で示された関数の厳密解はほぼ同じであるため、 $f(x)$ がこれらの関数の 1 つであったとき、その推定値はかなり正確になります。しかし、破線で示された関数の厳密解はその他の関数の解とまったく異なるため、 $f(x)$ がこの関数であったとき、現在の推定値は不正確なものとなります。

このアルゴリズムでは、サンプル点をもっと増やすことによって、関数の一般的な動作を理解できるようになります。ある積分区間における関数の変動が、それ以外の区間における変動と大きく異なる場合、ある反復で、アルゴリズムがその変動を検出します。こうした場合、継続した反復によって最も高速の（ただし**特性的な**）変動の存在を考慮した近似値が得られるまで、サンプル点の数を増加します。

例えば、次の近似値について考えてみます。

$$\int_0^{\infty} x e^{-x} dx.$$

数値積分を行うので、積分の上限は、この電卓で入力可能な最大値である 10^{499} とします。

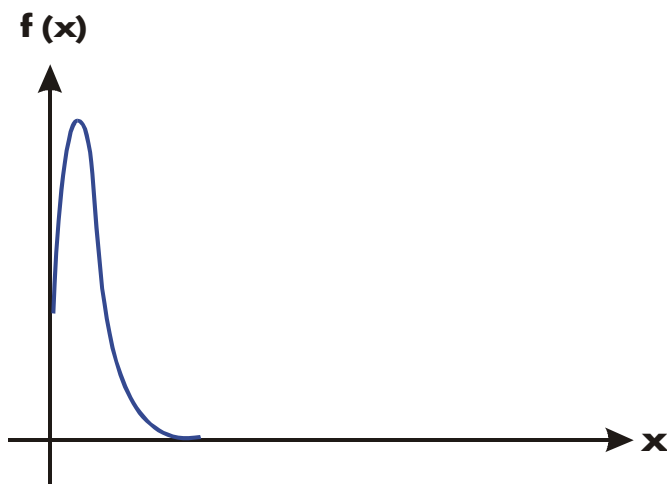
入力して何が起こるか試してみましょう。関数 $f(x) = xe^{-x}$ を入力します。

キー:	ディスプレイ:	説明:
EQN		方程式モードを選択します。
RCL X X → e^x	$XXEXP($	方程式を入力します。
+/- RCL X ENTER	$XXEXP(-X)$	方程式を終了します。
↵ SHOW	CK=2FE6 LN=9	チェックサムと長さ。
C		方程式モードをキャンセルします。

表示形式を SCI 3 に設定し、下限と上限をそれぞれゼロと 10499 に指定し、積分を開始します。

キー:	ディスプレイ:	説明:
✓ ↵ DISPLAY 2 (2SCI)		積分の確度レベルと限界を指定します。
3 ENTER 1 E 4	1E499_	
9 9		
EQN	$XXEXP(-X)$	方程式モードを選択し、方程式を表示します。
↵ ∫ X	INTEGRATING ∫ = 0.000E0	積分の近似値

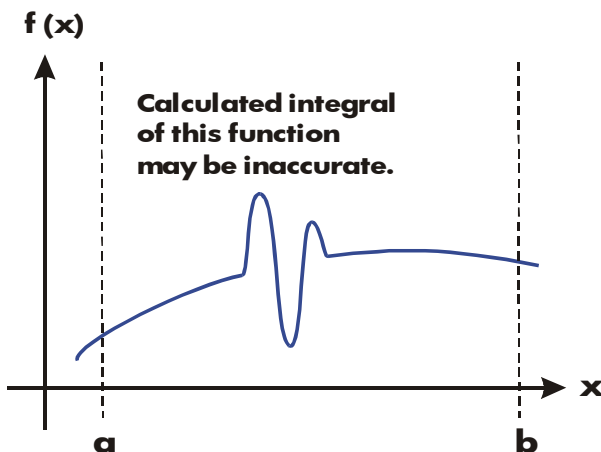
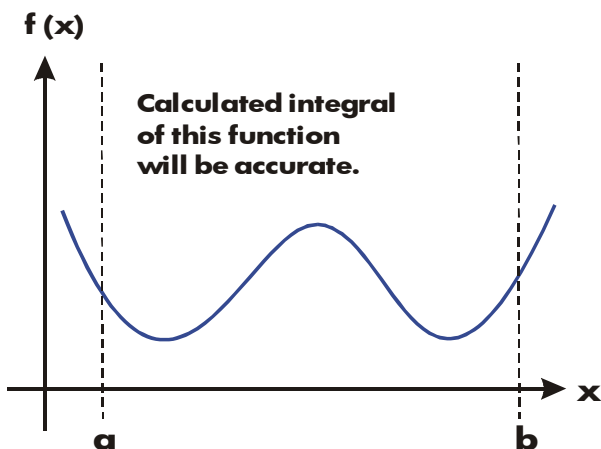
$f(x) = xe^{-x}$ のゼロから ∞ における積分の厳密解は 1 なので、得られた近似値は不正確です。ただし、 ∞ を 10499 で表現したことが問題なのではありません。この関数のゼロから 10499 の積分の厳密解は 1 に非常に近いからです。不正確になった理由は、 $f(x)$ の積分区間におけるグラフではっきりします。



このグラフは原点に非常に近い位置で大きく変動しています。この変動を検出するサンプル点が無いため、アルゴリズムは、 $f(x)$ が積分区間においてずっとゼロであるとみなしました。SCI 11 または ALL 形式で積分を計算することでサンプル点数を増やしても、この特定の関数在这个積分区間、積分したとき、追加サンプル点が変動をとらえません（このような場合の対処法については、次節の「計算時間が長くなる条件」を参照してください）。

このような異常（他の区間における関数の動作とは特性の異なる変動）を示す関数は、ほとんど存在しません。結果が不正確となる可能性のある関数は、関数とその低次の導関数が積分区間でどれだけ急激に変化するかを調べるという簡単な方法で特定できます。基本的に、関数または導関数の変化が急激であればあるほど、また急激に変化する導関数の次数が低いほど、計算時間が長くなり、得られた近似値の信頼性が低下します。

関数値や低次の導関数が急激に変化するとき、積分区間の幅に注意する必要があります。例えば、ある関数 $f(x)$ が積分区間で 3 回の変動を持つとします。3 回の変動が積分区間全体に渡っている方が、積分区間の狭い領域に限定されるよりも、変動の特性をサンプルでより良く表現できます（次の図に、これらの 2 つの状況を示します）。変動を関数の振動としてとらえると、その振動の期間と積分区間の幅の比を、信頼性の判断基準にすることができます。この比が大きいほど、計算速度が速くなり、得られる近似値の信頼性が増します。



多くの場合、積分する関数に十分慣れてくれば、関数が積分区間で急激に変動するかどうか予測できるようになります。関数に慣れておらず、しかもその関数が問題の原因であると疑われるときは、専用の方程式またはプログラムを使用して関数を評価することで、いくつかの点をすばやくプロットできます。

何らかの理由で積分に対する近似値の取得後に、その妥当性が疑われる場合、妥当性を確認するための簡単な手順があります。積分区間を複数に分割して、それぞれを新たな積分区間として関数を積分し、得られた近似値を加算します。これにより、関数がまったく新しいサンプル点でサンプリングされるので、前の積分では隠れていた変動が明らかになる可能性があります。初期近似値が妥当なものであれば、部分区間の近似値の総和と等しくなります。

E-6 積分の詳細

計算時間が長くなる条件

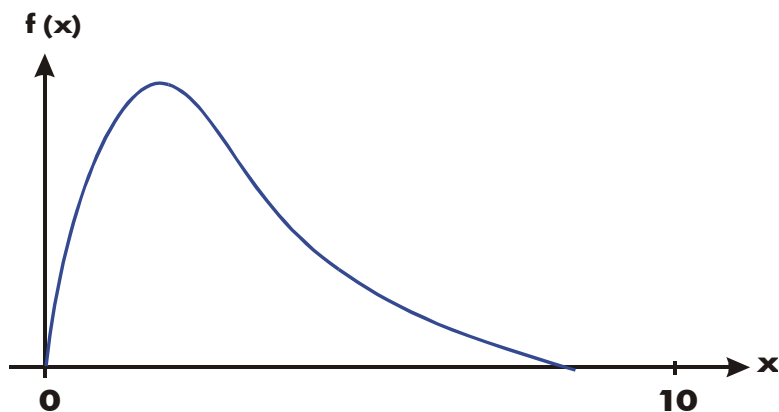
前の例では、関数の変動を検出できず、不正確な計算結果となっていました。これは、関数値の変動が積分区間の幅に比べて急激すぎたためです。積分区間の幅が狭まるほど、正しい結果が得られますが、それでもまだ区間が広い場合、非常に長い計算時間がかかります。

積分区間が広すぎるため過大な計算時間がかかるものの、区間が十分に広くないため計算結果が不正確になる積分を考えます。 $f(x) = xe^{-x}$ は、 x が ∞ に近づくとき急速にゼロに近づくため、大きな x の値では積分値に対する寄与度が無視できるほど小さくなります。このため、積分の上限の ∞ を 10^{499} ほど大きくない、 10^3 などで置き換えて積分を評価できます。

新しい積分の限界値を使用して前の積分問題を再実行します。

キー:	ディスプレイ:	説明:
0 ENTER 1 E	1E3_	新しい上限値。
3		
EQN	$X \times \text{EXP}(-X)$	方程式モードを選択し、方程式を表示します。
↵ ∫ X	INTEGRATING ∫ = 1.0000E0	積分（計算には 1 ~ 2 分かかります）。
x↔y	1.0000E-3	近似値の不確定値。

計算結果は妥当ですが、計算に非常に長い時間がかかります。原因を理解するには、関数の $x = 0$ と $x = 10^3$ 間のグラフ（前の例で示したものと同一ものを探します）を、 $x = 0$ と $x = 10$ 間のグラフと比較します。



グラフが示すとおり、この関数に変化があるのは、 x が小さな値のときだけです。 x の値が大きくなるにつれ、関数値が滑らかに次第に予測可能な方法で減少します。

アルゴリズムは、連続する複数の近似値の差が十分に小さくなるまで、サンプル点の密度を高めながら関数をサンプリングします。関数が変動する領域の狭い区間を対象にすると、この重要な密度に達するまでの時間が速まります。

同じサンプル点の密度を実現するには、大きな区間に必要なサンプル点の総数は、小さな区間に必要な数よりもはるかに大きくなります。つまり積分区域が広い場合に同じ確度の近似値を実現するには、数倍の繰り返し計算が必要となり、その結果、計算時間が非常に長くなります。

積分の計算時間は、関数が変動する領域で特定のサンプル点の密度をいかにすばやく達成できるかに依存します。したがって、積分区間に関数値の変動しない領域が多く含まれていれば、どんな関数でも計算時間が長くなります。そのような関数を計算するときは、計算時間が減少するように条件を調整することもできます。条件の調整には、積分区間を分割する手法と、変数を調整する手法があります。これらの手法により、関数または積分の上限と下限を変更することで、積分区間における被積分関数の動作が処理しやすくなります。

メッセージ

特定の状態になったときや特定のキーを押したとき、メッセージが表示されます。メッセージに対する注意を促すため ▲ 記号が表示されます。重要なメッセージは、クリアされるまで表示され続けます。[C] または [←] を押すとメッセージがクリアされ、前の表示内容に戻ります。その他のキーを押してもメッセージはクリアされますが、キーの機能は実行されません。

[FN ACTIVE	実行中のプログラムが、積分の計算中にプログラムラベル (FN= ラベル) を選択しようとしてしました。
[<[FN]	実行中のプログラムが、別の積分計算を実行中にプログラム ([FN ≠ 変数]) を積分しようとしてしました。
[<[SOLVE]	実行中のプログラムが、積分計算中にプログラムの解を求めようとしてしました。
ALL VARS=0	変数カタログ ([↵] [MEM] [1] (1VAR)) に値が格納されていません。
BAD GUESS	ある式の変数について SOLVE を実行したとき、誤った推定値 (複素数、ベクトルなど) を設定しました。
CALCULATING	計算時間がかかる機能を実行しています。
CLR ALL? Y N	メモリのすべての内容をクリアするかどうかを確認できます。
CLR EQN? Y N	編集集中の方程式をクリアするかどうかを確認できます (方程式入力モードでのみ発生します)。
CLR PGMS? Y N	メモリのすべてのプログラムをクリアするかどうかを確認できます (プログラム入力モードでのみ発生します)。
DIVIDE BY 0	ゼロで除算しようとしてしました (Y レジスタにゼロが保存されている状態で [↵] [%CHG] を押したときも表示されます)。

DUPLICAT LBL	別のプログラムルーチンに対してすでに存在するプログラムラベルを入力しようとした。
EQN LIST TOP	方程式メモリの「トップ」を示しています。メモリ方式は循環型であるため、EQN LIST TOP は、方程式メモリの最後の方程式の後の方程式でもあります。
INTEGRATING	方程式またはプログラムを積分しています。しばらく時間がかかります。
INTERRUPTED	ALG、RPN、EQN、PGM モードで [C] または [R/S] が押されたことにより、実行中の計算、SOLVE、f FN が中断されました。
INVALID DATA	データエラー： ■ エラーのあるデータを保存または計算しようとした。 ■ 組合せまたは順列を、$r > n$ であるか、r または n が整数でないか、$n \geq 10^{16}$ の状態で計算しようとした。 ■ 複素数またはベクトルを統計データに格納しようとした。 ■ 基数 n の最大許容桁数を超える桁を含んだ n 進数を保存しようとした。 ■ [x↔y] 操作を使用して、不正なデータを統計レジスタに格納しようとした。 ■ 複素数またはベクトルを比較しようとした。 ■ 三角関数または双曲線関数を不正な引数と一緒に使用しようとした。 ■ [TAN] と 90° の奇数倍の x ■ [ ACOS] または [ ASIN] と $x < -1$ または $x > 1$ ■ [ HYP  ATAN] と $x \leq -1$ または $x \geq 1$ ■ [ HYP  ACOS] と $x < 1$
INVALID VAR	方程式を解くときに、不正な変数名を入力しようとした。
INVALID x!	階乗またはガンマ関数を、 x を負の整数として計算しようとした。

INVALID ψx	指数エラー : ■ 0 の 0 乗または 0 の負の累乗を計算しようとしてしました。 ■ 負数を非整数乗しようとしてしました。 ■ 複素数 ($0 + i0$) を、負の実数部を持つ数値で累乗しようとしてしました。
INVALID (I)	不正な間接値を使って計算しようとしてしました ((I) が定義されていません)。
INVALID (J)	不正な間接値を使って計算しようとしてしました ((J) が定義されていません)。
LOG(0)	ゼロまたは ($0 + i0$) の対数を計算しようとしてしました。
LOG(NEG)	負の数値の対数を計算しようとしてしました。
MEMORY CLEAR	すべてのユーザメモリが消去されました (ページを参照してください)。
MEMORY FULL	指定した操作を実行するのに、十分な空きメモリがありません (付録 B を参照してください)。
NO	テスト手順でチェックされた状態が、真ではありません (キーボードから実行された場合のみ発生します)。
NONEXISTENT	[GTO]、[XEQ]、FN で、存在しないプログラムラベル (または行番号) を参照しようとしてしました。エラーメッセージ NONEXISTENT は次の場合に表示されます。 ■ 明示的に (キーボードから) 存在しないプログラムラベルを呼び出した場合 ■ 呼び出したプログラムが他のラベルを参照しており、そのラベルが存在しない場合 積分の結果が存在しない場合
NO LABELS	プログラムのカタログ ([] [MEM] [2] (2PGM)) にプログラムラベルが格納されていません。
NO SOLUTION	この線形方程式の解が見つかりませんでした。
MULT SOLUTION	この線形方程式で複数の解が見つかりました。

NO ROOT FND	SOLVE (EQN モードと PGM モードを含む) が、現在の初期推定値では方程式の根を発見できません (ページを参照してください)。状況として、推定値が不適切な場合、解で目的の点が見つからない場合、右辺と左辺が等しくない場合などがあります。プログラムの中で SOLVE が実行された場合、このエラーは表示されず、次のプログラム行 (SOLVE 変数) に続く行) にスキップします。
OVERFLOW	瞬間的に表示される警告です。計算結果の大きさが、電卓で取り扱い可能な範囲を超えています。電卓は、現在の表示形式で $\pm 9.999999999999999E499$ を返します (ページ「数値の範囲とオーバーフロー」を参照してください)。この状態になるとフラグ 6 がセットされます。フラグ 5 がセットされている場合、オーバーフローが発生すると、キーを押すまで実行中のプログラムが中断され、このメッセージが表示され続けます。
PRGM TOP	プログラムメモリの「トップ」を示しています。メモリ方式は循環型であるため、PRGM TOP は、プログラムメモリの最後の行の後の「行」でもあります。
RUNNING B	電卓が、方程式またはプログラムを実行しています (SOLVE と fFN ルーチン以外)。
SELECT FN	プログラムラベルを選択せずに SOLVE 変数または fFN d 変数を実行しようとした。これは、メッセージ MEMORY CLEAR が表示された後 SOLVE または fFN を最初に使用するときに発生します。または、現在のラベルがすでに存在しない場合に発生します。
SOLVE ACTIVE	実行中のプログラムが、SOLVE 実行中にプログラムラベル (FN= ラベル) を選択しようとした。
SOLVE<SOLVE>	実行中のプログラムが、SOLVE 実行中にプログラムの解を求めようとした。
SOLVE<fFN>	実行中のプログラムが、SOLVE 実行中にプログラムを積分しようとした。
SOLVING	電卓が、方程式またはプログラムの根を計算しています。しばらく時間がかかります。
SQRT<NEG>	負の数値の平方根を計算しようとした。

STAT ERROR

統計エラー :

- $n = 0$ で統計計算を実行しようとした。
- $n = 1$ で s_x , s_y , $\hat{x}$, $\hat{y}$, m , r , b を計算しようとした。
- x データのみ (すべての y 値がゼロ) で r , $\hat{x}$, $\bar{xw}$ を計算しようとした。
- すべての x 値が等しい状態で $\hat{x}$, $\hat{y}$, r , m , b を計算しようとした。

SYNTAX ERROR

方程式、式、**[SOLVE]**、**[/]** の評価中に、文法エラーが見つかりました。**[←]** または **[C]** を押すと、エラーメッセージをクリアし、エラーを修正できます。

TOO BIG

基数を HEX、OCT、BIN にするには数値が大きすぎます。数値は、次の範囲に存在する必要があります。
 $-34,359,738,368 \leq n \leq 34,359,738,367$ 。

XEQ OVERFLOW

実行中のプログラムが、21 番目の入れ子になった「XEQ ラベル」を実行しようとした (サブルーチンの入れ子は 20 までです)。SOLVE と「FN はそれぞれ 1 レベルを使用するので、これらによってもエラーが発生します。

YES

テスト手順でチェックされた状態が、真です (キーボードから実行された場合のみ発生します)。

セルフテストメッセージ

3SS-OK

セルフテストとキーボードテストに合格しました。

3SS-FAIL n

セルフテストまたはキーボードテストに失敗しました。電卓にはサービスが必要です。

© 2007 HP DEV CO., L. P.

セルフテストが無事に完了したときに表示される著作権メッセージです。

操作インデックス

このセクションは、すべての関数、演算、公式のクイックリファレンスです。リストは、関数名のアルファベット順になっています。この場合の関数名は、プログラム行で使用される名前です。例えば、FIX n という名前の関数は、   (1FIX) n として実行します。

プログラムで利用できない機能の名前は、キーボックスで示されています。例、.

リストでは、英字以外の文字とギリシア文字が、英字よりも先に来いています。関数名の前に矢印がある場合 (→DEG など)、アルファベット順に並べる際に矢印は無視されています。

最後の列 (*) には、表の最後にある注記の参照番号が示されています。

名前	キーと詳細	ページ	*
+/-	 数値の符号を変更します。	1-15	1
+	 加算。 $y + x$ を返します。	1-19	1
-	 減算。 $y - x$ を返します。	1-19	1
×	 乗算。 $y \times x$ を返します。	1-19	1
÷	 除算。 $y \div x$ を返します。	1-19	1
^	 累乗。指数を示します。	6-16	1
	入力した最後の数字の削除、 x のクリア、メニューのクリア、方程式に入力した最後の関数の削除、方程式の削除、プログラムステップの削除を行います。	1-4 1-8 6-3 13-7	
	カタログ内の直前のエントリの表示、方程式リスト内の前の方程式への移動、プログラムポインタの前のステップへの移動を行います。	1-28 6-3 13-11 13-20	

名前	キーと詳細	ページ	*
	カタログ内の次のエントリの表示、方程式リスト内の次の方程式への移動、プログラムポインタの次の行への移動（プログラム入力中）、現在のプログラム行の実行（プログラムの入力中以外の場合）を行います。	1-28 6-3 13-11 13-20	
または	カーソルを移動します。内容を削除しません。	1-14	
または	左や右に表示しきれない桁があるときにディスプレイをスクロールします。方程式や 2 進数の残りを表示します。CONST メニューと SUMS メニューで次のメニューページに移動します。	1-11 6-4 11-8	
	プログラムモードにおいて、方程式の先頭行、または最後のラベルの先頭行に移動します。	6-3	
	プログラムモードにおいて、方程式の最終行、または次のラベルの先頭行に移動します。	6-3	
,	関数の 2 つまたは 3 つの引数を分離します。	6-5	1
1/x	逆数	1-18	1
10 ^x	常用指数。 10 の $\times$ 乗を返します。	4-2	1
%	百分率。 $(y \times x) \div 100$ を返します。	4-6	1
%CHG	変化率。 $(x - y)(100 \div y)$ を返します。	4-6	1
π	近似値 3.14159265359 (12 桁) を返します。	4-3	1
$\Sigma+$	(y, x) を統計レジスタに累積します。	12-2	
$\Sigma-$	(y, x) を統計レジスタから削除します。	12-2	
Σx	(Σx) x 値の総和を返します。	12-11	1

名前	キーと詳細	ページ	*
Σx^2	 SUMS    (Σx^2) x 値の 2 乗の総和を返します。	12-11	1
Σxy	 SUMS      (Σxy) x 値と y 値の積の総和を返します。	12-11	1
Σy	 SUMS   (Σy) y 値の総和を返します。	12-11	1
Σy^2	 SUMS     (Σy^2) y 値の 2 乗の総和を返します。	12-11	1
σx	 S.σ   (σx) x 値の母標準偏差を返します。 $\sqrt{\sum (x_i - \bar{x})^2 \div n}$	12-7	1
σy	 S.σ    (σy) y 値の母標準偏差を返します。 $\sqrt{\sum (y_i - \bar{y})^2 \div n}$	12-7	1
$\int$ FN d 変数	  ($\int$ FN d _) 変数 表示中の方程式または FN= で選択されたプログラムを積分します。 Y レジスタの値を積分変数の下限、 X レジスタの値を積分変数の上限とします。	8-2 15-7	
()	 括弧。  を押して、今後の計算のために括弧を残します。	6-6	1
[]	  : ベクトル演算を実行するためのベクトル記号	10-1	1
θ	  : 複素数演算を実行するための複素数記号	9-1	1

名前	キーと詳細	ページ	*
A から Z	変数 名前付き変数の値	6-4	1
ABS	ABS 絶対値。 $ x $ を返します。	4-17	1
ACOS	ACOS 余弦の逆関数。 $\cos^{-1}x$ を返します。	4-4	1
ACOSH	ACOSH 双曲線余弦の逆関数。 $\cosh^{-1}x$ を返します。	4-6	1
(4ALG)	代数モードをアクティブにします。	1-9	
ALOG	10^x 常用指数。 10 の指定累乗 (真数) を返します。	6-16	1
ALL	(4ALL) すべての有効桁を表示します。必要に応じて右にスクロールして () 残りの桁を表示させます。	1-23	
AND	(1AND) 論理演算子	11-4	1
ARG	ARG 複素数を引数 $\theta =$ と置換します。	4-17	1
ASIN	ASIN 正弦の逆関数 $\sin^{-1}x$ を返します。	4-4	1
ASINH	ASINH 双曲線正弦の逆関数。 $\sinh^{-1}x$ を返します。	4-6	1
ATAN	ATAN 正接の逆関数。 $\tan^{-1}x$ を返します。	4-4	1
ATANH	ATANH 双曲線正接の逆関数。 $\tanh^{-1}x$ を返します。	4-6	1
b	(b) 回帰線の y 切片を返します。 $\bar{y} - m\bar{x}$.	12-11	1

名前	キーと詳細	ページ	*
b	 BASE 8 (8b) 2 進数を示します	11-2	1
 BASE	基数変換メニューを表示します。	11-1	
BIN	 BASE 4 (4BIN) 2 進数 (base 2) モードを選択します。	11-1	
 C	電卓の電源投入、x のクリア、メッセージとプロンプトのクリア、メニューのキャンセル、カタログのキャンセル、方程式入力のキャンセル、プログラム入力のキャンセル、方程式の実行の中断、プログラムの実行の中断を行います。	1-1 1-4 1-8 1-29 6-3 13-7 13-19	
/c	  分母。 表示する分数に対する分母の最大値を x に設定します。x = 1 の場合、現在の /c 値を表示します。	5-4	
→°C	  華氏を摂氏に変換します。	4-14	
CF n	 FLAGS 2 (2CF) n フラグ n (n = 0 ~ 11) をクリアします。	14-12	
 CLEAR	数値またはメモリの一部をクリアするためのメニューを表示します。指定された変数やプログラムを MEM カタログからクリアします。表示中の方程式をクリアします。	1-5 1-28	
 CLEAR 3 (3ALL)	格納されているデータ、方程式、プログラムをすべてクリアします。	1-29	
 CLEAR 3 (3PGM)	すべてのプログラムをクリアします (プログラムモードの電卓)。	13-23	
 CLEAR 3 (3EQN)	表示されている方程式をクリアします (方程式モードの電卓)。	13-7	
CLS	 CLEAR 4 (4Σ) 統計レジスタをクリアします。	12-1	
CLVARS	 CLEAR 2 (2VARS) すべての変数をゼロにクリアします。	3-6	

名前	キーと詳細	ページ	*
CLx	CLEAR 1 (1%) x (X レジスタ) をゼロにクリア します。	2-3 2-7 13-7	
CLVARx	CLEAR 6 (6CLVARx) アドレスが x アドレスより大きい 間接変数をゼロにクリアします。	1-4	
CLSTK	CLEAR 5 (5STK) すべてのスタックレベルをゼロに クリアします。	2-7	
→CM	→cm インチをセンチメー トルに変換します。	4-14	1
nCr	nCr n 個のアイテムから一度 に r 個を取り出したときの 組合せ 。 $n! \div (r! (n-r)!)$ を返します。	4-15	1
COS	COS 余弦。 $\cos x$ を返します。	4-3	1
COSH	HYP COS 双曲線余弦。 $\cosh x$ を返します。	4-6	1
CONST	41 個の物理定数にアクセスします。	4-8	
d	BASE 5 (5d) 10 進数を示します	11-1	1
DEC	BASE 1 (1DEC) 10 進数モードを選択します。	11-1	
DEG	MODE 1 (1DEG) 角度モードとして度を選択します。	4-4	
→DEG	→DEG ラジアンから度への 変換。 $(360/2\pi) x$ を返します。	4-13	1
DISPLAY	表示形式、区切り文字 (・ と 、)、 桁区切り、複素数の表示形式を設定 するためのメニューを表示します。	1-21	
DSE 変数	DSE 変数減少、以下であれば スキップ。変数に制御値 ccccccc.ffff が格納されている場合、ii (増減値) を ccccccc (カウンタ値) から減算 します。結果 ≤fff (最終値) であ れば、次のプログラム行をスキップ します。	14-18	

名前	キーと詳細	ページ	*
E	指数の入力を開始し、入力中の数値に "E" を追加します。10 の累乗が続くことを示します。	1-15	1
ENG n	 DISPLAY 3 (3ENG) _n 最初の数値の後に n 桁が続く工学表示を選択します (n = 0 ~ 11)。	1-22	
 ENG と ENG 	表示中の数値の指数表示が、3 の倍数で変化します。	1-22	
ENTER	連続して入力する 2 つの数値を分離します。方程式の入力を完了します。表示中の方程式を評価します (適切な場合は結果を格納します)。	1-19 6-4 6-11	
ENTER	ENTER x を Y レジスタにコピーします。 y を Z レジスタに、 z を T レジスタに移動します。 t は失われます。	2-6	
EQN	方程式入力モードをアクティブまたはキャンセルに切り替えます。	6-3 13-7	
e ^x	 e^x e の指数関数。 e の x 乗を返します。	4-1	1
EXP	 e^x e の指数関数。 e の指定累乗を返します。	6-16	1
→°F	 →°F 摂氏を華氏に変換します。	4-14	1
 FDISP	分数表示モードのオンとオフを切り替えます。	5-1	
FIX n	DISPLAY 1 (1FIX) _n 小数点以下 n 桁の固定表示を選択します。0 ≤ n ≤ 11。	1-21	
 FLAGS	フラグを設定、クリア、テストするためのメニューを表示します。	14-12	
FN = ラベル	 FN= ラベル ラベル付きプログラムを現在の関数として選択します (SOLVE と「FN で使用)。	15-1 15-7	
FP	 INTG 5 (5FP) x の小数部。	4-17	1

名前	キーと詳細	ページ	*
FS? <i>n</i>	FLAGS 3 (3FS?) <i>n</i> フラグ <i>n</i> (<i>n</i> = 0 ~ 11) がセットされている場合、次のプログラム行を実行します。フラグ <i>n</i> がクリアのときは、次のプログラム行をスキップします。	14-12	
→GAL	gal リットルをガロンに変換します。	4-14	1
GRAD	MODE 3 (3GRD) 角度モードをグラジアンに設定します。	4-4	
ラベル <i>nnn</i>	プログラムポインタをプログラムラベルの <i>nnn</i> 行に設定します。	13-21	
.	プログラムポインタを PRGM TOP に設定します。	13-21	
h	BASE 6 (6h) 16 進数を示します	11-1	1
HEX	BASE 2 (2HEX) 16 進数 (base 16) モードを選択します。	11-1	
HYP	双曲線関数の HYP_ prefix を表示します。	4-6	
→HMS	HMS 時間を、時間、分、秒に変換します。 <i>x</i> を、小数から時間、分、秒形式に変換します。	4-13	1
HMS→	HMS→ 時間、分、秒を、時間に変換します。 <i>x</i> を、時間、分、秒形式から小数に変換します。	4-13	1
i	複素数の入力に使用します	9-2	1
(I)/(J)	RCL (I) / (J) , STO (I) / (J) . 変数 I / J に格納された数値に対応する文字が付いた変数の値。	6-4 14-21	1
→IN	in センチメートルをインチに変換します。	4-14	1

名前	キーと詳細	ページ	*
IDIV	INTG 2 (2INT÷) 2つの整数に対する除算の商を生成します。	6-16	1
INT÷	INTG 2 (2INT÷) 2つの整数に対する除算の商を生成します。	4-2	1
INTG	INTG 4 (4INTG) 与えられた数値以下の最大整数を取得します。	4-18	1
INPUT 変数	INPUT 変数 変数を X レジスタに呼び出します。変数の名前と値を表示します。プログラム実行を停止します。 R/S (プログラム実行を再開するため) または (現在のプログラム行を実行するため) を押して、入力を変数に格納します (プログラムでのみ使用されます)。	13-13	
INV	引数の逆数。	6-16	1
IP	INTG 6 (6IP) x の整数部。	4-17	1
ISG 変数	ISG 変数 増加、より大きければスキップ。 変数に制御値 cccccc.fffj が格納されている場合、jj (増減値) を cccccc (カウンタ値) に加算します。結果 >fff (最終値) であれば、次のプログラム行をスキップします。	14-18	
→KG	ポンドをキログラムに変換します。	4-14	1
→KM	マイルをキロメートルに変換します。	4-14	1
→L	ガロンをリットルに変換します。	4-14	1
LASTx	LASTx LAST X レジスタに格納されている数値を返します。	2-8	
→LB	キログラムをポンドに変換します。	4-14	1

名前	キーと詳細	ページ	*
LBL ラベル	LBL ラベル XEQ、GTO、FN= で参照される、 英字 1 文字のプログラムラベルを 定義します (プログラムでのみ使 用されます)。	13-3	
LN	LN 自然対数。 $\log_e x$ を返します。	4-1	1
LOG	LOG 常用対数。 $\log_{10} x$ を返します。	4-1	1
L.R.	線形回帰メニューを表示します。	12-4	
m	L.R. (m) 回帰線の傾きを返します。 $[\sum(x_i - \bar{x})(y_i - \bar{y})] \div \sum(x_i - \bar{x})^2$	12-7	1
→MILE	→MILE キロメートルをマイ ルに変換します。	4-14	1
MEM	使用可能なメモリ量とカタログメ ニューを表示します。	1-28	
MEM 2 (2PGM)	プログラムのカタログを開始 します。	13-22	
MEM 1 (1VAR)	変数のカタログを開始します。	3-4	
MODE	ALG または RPN モード、または角 度モードを設定するためのメ ニューを表示します。	1-7 4-4	
n	SUMS (n) データ点セットの数を返します。	12-11	1
NAND	LOGIC 5 (5NAND) 論理演算子	11-4	1
NOR	LOGIC 6 (6XOR) 論理演算子	11-4	1
NOT	LOGIC 4 (4NOT) 論理演算子	11-4	1
o	BASE 7 (7o) 8 進数を示します	11-2	1

名前	キーと詳細	ページ	*
OCT	BASE (3OCT) 8 進数 (base 8) モードを選択します。	11-1	
OR	LOGIC (3OR) 論理演算子	11-4	1
OFF	電卓の電源をオフにします。	1-1	
nPr	nPr n 個のアイテムから一度に r 個を取り出したときの順列。 $n! \div (n-r)!$ を返します。	4-15	1
PRGM	プログラム入力モードをアクティブまたはキャンセルに切り替えます。	13-6	
PSE	PSE 中断。 プログラムの実行を中断して、 x 、変数、または方程式を表示してから、すぐに実行を再開します (プログラムでのみ使用されます)。	13-18 13-19	
r	L.R. (r) x 値と y 値間の相関係数を返します。 $\frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2 \times \sum (y_i - \bar{y})^2}}$	12-7	1
$\rho\theta\alpha$	DISPLAY (10 $\rho\theta\alpha$) 複素数の表示を変更します。	1-25	
RAD	MODE (1RAD) 角度モードとしてラジアンを選択します。	4-4	
→RAD	→RAD 度からラジアンへの変換。 $(2\pi/360) \times$ を返します。	4-13	1
RADIX ,	DISPLAY (6,) 基数記号 (小数点) としてカンマを選択します。	1-23	
RADIX .	DISPLAY (5.) 基数記号 (小数点) としてピリオドを選択します。	1-23	

名前	キーと詳細	ページ	*
RANDOM	 RAND RANDOM 関数を実行します。0 から 1 の間の乱数を返します。	4-15	1
RCL 変数	 変数 呼び出し。 変数を X レジスタにコピーします。	3-7	
RCL+ 変数	  変数 $x +$ 変数を返します。	3-7	
RCL- 変数	  変数 $x -$ 変数を返します。	3-7	
RCLx 変数	  変数 $x \times$ 変数を返します。	3-7	
RCL÷ 変数	  変数 $x \div$ 変数を返します。	3-7	
RMdR	 INTG  (3Rmdr) 2 つの整数に対する除算の剰余を生成します。	6-16	1
RND	 RND 丸め 。 FIX n 表示モードでは、 x を小数点以下 n 桁に丸めます。SCI n または ENG n 表示モードでは、 $n + 1$ の有効桁に丸めます。分数表示モードでは、表示中の分数に一番近い 10 進数に丸めます。	4-18 5-8	1
 RPN	 MODE  (5RPN) 逆ポーランド記法をアクティブにします。	1-9	
RTN	 RTN リターン 。 プログラムの終了をマークします。プログラムポインタが、トップまたは呼び出したルーチンに戻ります。	13-4 14-1	
R↓	 ロールダウン 。 RPN モードにおいて t を Z レジスタに、 z を Y レジスタに、 y を X レジスタに、 x を T レジスタに移動します。 ALG モードでスタックを確認するための X,Y,Z,T メニューを表示します。	2-3 C-7	

名前	キーと詳細	ページ	*
R↑	R↑ ロールアップ。 RPN モードにおいて t を X レジスタに、 z を T レジスタに、 y を Z レジスタに、 x を Y レジスタに移動します。	2-3 C-7	
S.σ	ALG モードでスタックを確認するための X,Y,Z,T メニューを表示します。 標準偏差メニューを表示します。	12-4	
SCI n	DISP 2 (2SCI) n 小数点以下 n 桁の科学表記を選択します ($n = 0 \sim 11$)。	1-22	
SEED	SEED シード $ X $ を使って乱数シーケンスを再度開始します。	4-15	
SF n	FLAG 1 (1SF) n フラグ n ($n = 0 \sim 11$) をセットします。	14-12	
SGN	INTG 1 (1SGN) x の符号を示します。	4-17	1
SHOW	x (または現在のプログラム行の数値) の仮数 (12 桁) をすべて表示します。方程式とプログラムの 16 進数チェックサムと 10 進数バイト長を表示します。	6-19 13-23	
SIN	SIN 正弦。 $\sin x$ を返します。	4-3	1
SINH	HYP SIN 双曲線正弦。 $\sinh x$ を返します。	4-6	1
SOLVE 変数	SOLVE 変数 表示中の方程式または FN= で選択されたプログラムの解を求めます。 変数 と x の初期推定値を使用します。	7-1 15-1	
SPACE	方程式入力中に空白文字を挿入します。	14-14	1
SQ	x² 引数の 2 乗。	6-16	1

名前	キーと詳細	ページ	*
SQRT	x の平方根。	6-16	1
STO 変数	STO 変数 格納。x を変数にコピーします。	3-2	
STO + 変数	STO 変数 変数 + x を変数に格納します。	3-6	
STO - 変数	STO 変数 変数 - x を変数に格納します。	3-6	
STO × 変数	STO 変数 変数 × x を変数に格納します。	3-6	
STO ÷ 変数	STO 変数 変数 ÷ x を変数に格納します。	3-6	
STOP	R/S 実行／停止。 現在のプログラム行からプログラムの実行を開始します。実行中のプログラムを停止し、X レジスタを表示します。	13-19	
SUMS	総和メニューを表示します。	12-4	
sx	(Σx) x 値の標本標準偏差を返します。 $\sqrt{\sum (x_i - \bar{x})^2 \div (n - 1)}$	12-6	1
sy	(Σy) y 値の標本標準偏差を返します。 $\sqrt{\sum (y_i - \bar{y})^2 \div (n - 1)}$	12-6	1
TAN	TAN 正接。tan x を返します。	4-3	1
TANH	双曲線正接。 tanh x を返します。	4-6	1
VIEW 変数	VIEW 変数 スタックに値を呼び出すことなく、 変数のラベル付き内容を表示 します。	3-4 13-15	
XEQ	表示中の方程式を評価します。	6-12	
XEQ ラベル	XEQ ラベル ラベルによって識別されたプログラムを実行します。	14-1	

名前	キーと詳細	ページ	*
x^2	x^2 x の 2 乗。	4-2	1
$\sqrt{x}$	$\sqrt{x}$ x の平方根。	4-2	1
$\sqrt[x]{y}$	$\sqrt[x]{y}$ y の x 乗根。	4-2	1
$\bar{x}$	$\bar{x}, \bar{y}$ ($\bar{\bar{x}}$) x 値の平均を返します。 $\Sigma x_i \div n$	12-4	1
$\hat{x}$	$L.R.$ ($\hat{x}$) X レジスタの y 値が与えられた場合、回帰線に基づいた x 推定値を返します。 $\hat{x} = (y - b) \div m$	12-11	1
!	$!$ 階乗 (ガンマ関数)。 $(x)(x-1) \dots (2)(1)$ 、または $\Gamma(x+1)$ を返します。	4-15	1
XROOT	$\sqrt[x]{y}$ 引数 2 の引数 1 乗根。	6-16	1
$\bar{x}w$	$\bar{x}, \bar{y}$ $\bar{w}$ ($\bar{\bar{x}}w$) x 値の加重平均を返します。 $(\Sigma y_i x_i) \div \Sigma y_i$	12-4	1
$\bar{x}, \bar{y}$	平均 (算術平均) メニューを表示します。	12-4	
$x <> \text{変数}$	$X \leftrightarrow$ x の交換。 x と変数を交換します。	3-8	
$x <> y$	$x \leftrightarrow y$ x と y の交換。 x を Y レジスタに、 y を X レジスタに移動します。	2-4	
$X?Y$	" $x?y$ " 比較テストメニューを表示します。	14-7	
$x \neq y$	$X?Y$ ($\neq$) $x \neq y$ の場合、次のプログラム行を実行します。 $x = y$ の場合、次のプログラム行をスキップします。	14-7	
$x \leq y?$	$X?Y$ $\leq$ ($\leq$) $x \leq y$ の場合、次のプログラム行を実行します。 $x > y$ の場合、次のプログラム行をスキップします。	14-7	

名前	キーと詳細	ページ	*
$x < y?$	    (<) $x < y$ の場合、次のプログラム行を実行します。 $x \geq y$ の場合、次のプログラム行をスキップします。	14-7	
$x > y?$	     (>) $x > y$ の場合、次のプログラム行を実行します。 $x \leq y$ の場合、次のプログラム行をスキップします。	14-7	
$x \geq y?$	      ($\geq$) $x \geq y$ の場合、次のプログラム行を実行します。 $x < y$ の場合、次のプログラム行をスキップします。	14-7	
$x = y?$	       (=) $x = y$ の場合、次のプログラム行を実行します。 $x \neq y$ の場合、次のプログラム行をスキップします。	14-7	
 	"x?0" 比較テストメニューを表示します。	14-7	
$x \neq 0?$	  ($\neq$) $x \neq 0$ の場合、次のプログラム行を実行します。 $x = 0$ の場合、次のプログラム行をスキップします。	14-7	
$x \leq 0?$	   ($\leq$) $x \leq 0$ の場合、次のプログラム行を実行します。 $x > 0$ の場合、次のプログラム行をスキップします。	14-7	
$x < 0?$	    (<) $x < 0$ の場合、次のプログラム行を実行します。 $x \geq 0$ の場合、次のプログラム行をスキップします。	14-7	

名前	キーと詳細	ページ	*
$x>0?$	$x?0$ ($>$) $x>0$ の場合、次のプログラム行を実行します。 $x\leq 0$ の場合、次のプログラム行をスキップします。	14-7	
$x\geq 0?$	$x?0$ ($\geq$) $x\geq 0$ の場合、次のプログラム行を実行します。 $x<0$ の場合、次のプログラム行をスキップします。	14-7	
$x=0?$	$x?0$ (=) $x=0$ の場合、次のプログラム行を実行します。 $x\neq 0$ の場合、次のプログラム行をスキップします。	14-7	
XOR	LOGIC ($\oplus$) 論理演算子	11-4	1
$x \cdot y$	DISPLAY ($\otimes$) 複素数の表示を変更します。	4-11	
$x + y \cdot i$	DISPLAY ($11x + y \cdot i$) 複素数の表示を変更します。 ALG モードのみ。	1-25	
$\bar{y}$	$\bar{x} \cdot \bar{y}$ ($\bar{y}$) y 値の平均を返します。 $\Sigma y_i \div n$ 。	12-4	1
$\hat{y}$	L.R. ($\hat{y}$) X レジスタの x 値が与えられた場合、回帰線に基づいた y 推定値を返します。 $\hat{y} = m x + b$	12-11	1
y^x	累乗。 y の x 乗を返します。	4-2	1

注記：

1. 方程式でも使用可能な関数

索引

特殊文字

FN。積分を参照

% 関数 4-6

1-15

分数 1-26

π 4-4, A-2

▲ ▼ インジケータ

分数 5-2, 5-3

↔ インジケータ

方程式 6-7

2 進数 11-8

方程式 13-7

. バックスペースキーを参照

. 数値入力カーソルを参照

. 積分を参照

インジケータ 1-2

インジケータ 1-1, A-3

数字

1 変数の統計 12-2

10 の指数 1-15, 1-16

10 進数モード。基数モードを参照

16 進数。数値を参照

四則演算 11-4

入力 11-1

範囲 11-7

変換 11-2

2 の補数 11-4, 11-6

2 変数の統計 12-2

2 進数。数値を参照

四則演算 11-4

スクロール 11-8

全桁の表示 11-8

入力 11-1

範囲 11-7

変換 11-

8 進数。数値を参照

四則演算 11-4

入力 11-1

範囲 11-7

変換 11-2

A

A..Z インジケータ 1-3, 3-2, 6-4

ALG 1-9

プログラム 13-4

方程式との比較 13-4

B

BIN インジケータ 11-1

C

%CHG の引数 4-6, C-3

VIEW のキャンセル 3-4

X レジスタのクリア 2-3, 2-7

オンとオフ 1-1

カタログの終了 1-4

コントラストの調整 1-1

操作 1-4

プロンプトのキャンセル 1-4

メッセージのクリア 1-4

メニューの終了 1-4, 1-9

/c の値 5-4

SOLVE の停止 7-8, 15-1

積分の中止 8-2, 15-8

プログラムの中断 13-19

プログラムモードの終了 13-7

プロンプトのキャンセル

6-14, 13-15

方程式モードの終了 6-3, 6-4

/c の値 B-4

/c の値 B-6

CLEAR メニュー 1-5

D

DSE 14-18

E

E (数値内) 1-15, 1-21, A-1
ENG 表記 1-22 表示形式も参照
ENTER

数値の区切り 1-17, 2-6
数値の複製 2-6
スタック操作 2-6
スタックのクリア 2-6
表示された変数のコピー 13-15
方程式の終了 6-4, 6-8, 13-7
方程式の評価 6-10, 6-11

E (指数) 1-16
EQN LIST TOP 6-7, F-2

EQN インジケータ
プログラムモード 13-7
方程式リスト 6-4, 6-7

F

FN. 積分を参照

FDISP
表示モードの切り替え 5-1, A-2
フラグの切り替え 14-9
プログラム不能 5-10

FIX 表記 1-21 表示形式も参照
FN=

プログラム 15-6, 15-10
プログラムの積分 15-8
プログラムの SOLVE 15-1

G

GTO
PRGM TOP の検索
13-6, 13-21, 14-6
ガンマ関数 4-15
プログラム行の検索 13-22, 14-5
プログラムラベルの検索
13-11, 13-21, 14-6
GTO 14-5, 14-16

H

HEX インジケータ 11-1

I

i 3-10, 14-20, 14-21
(i) 14-19, 14-20, 14-21
INPUT
SOLVE プログラム 15-2
応答 13-14
積分プログラム 15-8
プログラムデータの入力 13-12
プロンプトを常に表示 14-11
ISG 14-16

L

LAST X レジスタ 2-8, B-6
LASTx 関数 2-8
Łukasiewicz 2-1

M

MEM
プログラムのカタログ
1-28, 13-22
変数カタログ 1-28
メモリの確認 1-28
MODES メニュー
角度モード 4-4

O

OFF 1-1
OCT インジケータ 11-1, 11-5

P

π A-2
PRGM TOP 13-4, 13-6, 13-21, F-4
PSE
プログラム停止の防止 14-10
プログラムの一時停止
13-18, 15-10

R

R/S
SOLVE の停止 7-9, 15-1
積分の中止 8-2, 15-8
プログラムの再開 13-16, 13-19
プログラムの実行 13-22
プログラムの中断 13-19
プロンプトの終了 6-11, 6-14,
7-2, 13-15

R↓ と R↑ 2-3, C-8
RCL 3-2, 13-14
RCL 四則演算 3-8
return (プログラム)。プログラムを

参照

RPN

起源 2-1
プログラム 13-4
方程式との比較 13-4

S

SCI 表記。表示形式を参照
設定 1-21
プログラム 13-7

SHOW

数値の桁 1-25, 13-7
プログラムのチェックサム
13-23, B-2
プログラムの長さ 13-22, B-2
プロンプト桁 6-14
方程式のチェックサム 6-19, B-2
方程式の長さ 6-19, B-2

SOLVE

極 D-6
結果の確認 7-8, D-3
根が発見できない 7-8, 15-6, D-8
再開 15-1
極小値または極大値 D-8
使用 7-1
初期推定値
7-2, 7-8, 7-9, 7-12, 15-6
スタック上の結果 7-2, 7-7, D-3
制約 15-11
中断 7-2, 7-9
動作 7-8, D-1
端数処理 D-13
複数の根 7-10
不連続性 D-6
プログラムの評価 15-2
プログラム 15-6
平坦な領域 D-8
方程式の評価 7-1, 7-7
目的 7-1

SPACE

14-14
STO 3-2, 13-12
STO 四則演算 3-7
STOP 13-19

T

T レジスタ 2-5
TVM 17-1

V

VIEW

スタックへの影響なし 13-15
プログラムデータの表示 13-15,
13-18, 15-6
プログラムの停止 13-15
変数の表示 3-4

X

X ROOT の引数 6-17

X レジスタ

VIEW による影響なし 13-15
Y レジスタとの交換 2-4
クリア 1-5, 2-3, 2-7
クリアなし 2-5
スタックの一部 2-1
テスト 14-7
表示 2-3
プログラムでのクリア 13-7
プログラム一時停止中 13-19
プロンプトによる影響 6-14
変数との交換 3-8
変数による四則計算 3-7

XEQ

プログラムの実行 13-10, 13-22
方程式の評価 6-10, 6-13

あ

アドレッシング

間接 14-19, 14-20, 14-21,

い

移動。GTO を参照

一般的な質問への回答 A-1

インジケータ

英字 1-3
シフトキー 1-2
電池容量低下 1-1, A-3
電池 1-1, A-3
フラグ 14-11
リスト 1-13

う

ウィンドウ (2 進数) 11-8

え

英字 1-3

エラー

クリア 1-4

修正 2-8, F-1

エラーメッセージ F-1

お

オーバーフロー

応答の設定 14-9, F-4

計算の結果 1-17, 11-5

発生のテスト 14-9

フラグ 14-9, F-4

温度

単位の変換 4-14

電卓に対する制限 A-2

オンとオフ 1-1

か

カーブフィッティング 12-9, 16-1

回帰 (線形) 12-8, 16-1

階乗関数 4-15

角度

暗黙の単位 4-4, A-2

形式の変換 4-13

単位の変換 4-13

ベクトル間 10-5

角度モード 4-4, A-2, B-3

確率

関数 4-15

正規分布 16-11

貸し手 (金融) 17-1

加重平均 12-5

仮数 1-25

傾き (カーブフィット) 12-8, 16-1

カタログ

終了 1-4

使用 1-29

プログラム 1-29, 13-22

変数 1-29, 3-4

括弧

四則計算内の括弧 2-12

方程式内の括弧 6-5, 6-6, 6-15

貨幣 (金融) 17-1

借り手 (金融) 17-1

関数

2 つの引数 1-19, 2-9, 9-3

実数 4-1

単一引数 1-18, 2-9

表示される名前 13-8

プログラム不能 13-24

方程式 6-5, 6-16

リスト G-1

関数の極 D-6

関数の極小値 D-8

関数の極大値 D-8

関数の不連続性 D-5

間接アドレッシング 14-19, 14-20, 14-21

カンマ (数値内) 1-23, A-1

き

キー

英字 1-3

シフト 1-3

文字 1-3

基数

四則演算 11-4

設定 11-1

デフォルト B-3

表示への影響 11-6

プログラム 11-8, 13-25

変換 11-2

基数マーク A-1

基数モード

設定 13-25

デフォルト B-4

プログラム 13-25

方程式 6-5, 6-11, 13-25

逆関数 9-3

逆三角関数 4-6, C-6

逆正規分布 16-11

逆双曲線関数 4-6

逆ポーランド記法。RPN を参照

キャッシュフロー 17-1

極座標と直交座標の変換 4-10, 9-5

虚数部 (複素数) 9-1, C-9

貨幣の時間的価値 17-1

金融計算 17-1

く

組合せ 4-15

グラジアン (角度の単位) 4-4, A-2

クリア

X レジスタ 2-3, 2-7

一般的な情報 1-4

数値 1-17

統計レジスタ 12-1

プログラム 1-29, 13-23

変数 1-28

方程式 6-9

メモリ 1-29, A-1

グループ化された標準偏差 16-18

け

計算

一般的な手順 1-18

計算の順番 2-14

実数 4-1

スタック操作 2-5, 9-2

中間結果 2-12

長い計算 2-12

複素数 9-1

現在の値。金融計算を参照

こ

根。SOLVE を参照

確認 7-8, D-3

発見できない 7-8, D-8

複数 7-9

プログラム内の根 15-6

プログラムの根 15-1

方程式 7-1

根関数 4-3

コントラストの調整 1-1

さ

最大の整数 4-18

最良の回帰 12-8, 16-1, C-13

座標

変換 4-10

サブルーチン。ルーチンを参照

三角関数 4-4, 9-4, C-6

四則演算

16 進数 11-4

2 進数 11-4

8 進数 11-4

一般的な手順 1-18

計算の順番 2-14

スタック操作 2-5, 9-2

中間結果 2-12

長い計算 2-12

四則演算の格納 3-7

四則演算の呼び出し 3-8

残高 (金融) 17-1

し

シード値 (乱数) 4-15

時間の表示形式 4-13

指数カーブフィッティング 16-1

指数関数 1-16, 4-1, 9-3,

実数

演算 4-1

実数部 (複素数) 9-1

質問 A-1

質量の単位変換 4-14

重量の単位変換 4-14

支払い (金融) 17-1

シフトキー 1-2

順列 4-15

条件テスト 14-6, 14-7, 14-9, 14-12,
14-17

乗算、除算 10-2

小数点 A-1

小数部関数 4-18

将来価値 (金融) 17-1

除算の商と剰余 4-2

シングルステップ実行 13-11

真なら実行 14-6, 15-6

す

推定値 (SOLVE の場合) 7-2, 7-8, 7-9,
7-13, 15-6

数値。2 進数、16 進数、8 進数、

変数を参照

E 1-15, A-1

大きな数字と小さな数字
1-15, 1-17

格納 3-2

基数 10-1, 13-25

切り捨て 11-6

クリア 1-4, 1-5, 1-17

交換 2-4
再利用 2-6, 2-10
算術計算の実行 1-18
実数 4-1
精度 D-13
全桁の表示 1-25
素数 17-6
内部表現 11-6
入力 1-15, 1-16, 11-1
範囲 1-17, 11-7
表示形式 1-21, 11-6
ピリオドとカンマ 1-23, A-1
負 1-15, 9-3, 11-6
複素数 9-1
符号の変更 1-15, 9-3
部分の検索 4-17
プログラム内の数値 13-7
分数 1-26, 5-1
編集 1-4, 1-17
方程式内の数値 6-5
丸め 4-18
呼び出し 3-2
数値の符号の変更 1-15, 9-3
スクロール
2 進数 11-8
方程式 6-7, 13-7, 13-16
スタック。スタック上昇を参照
VIEW による影響なし 13-15
X と Y の交換 2-4
影響 12-6
大きさの限界 2-4, 9-2
確認 2-3, C-8
操作 2-1, 2-5, 9-2
定数の格納 2-7
長い計算 2-12
複素数 9-2
プログラムの計算 13-14
プログラムの出力 13-12
プログラムの入力 13-12
プロンプトによる影響 6-14, 13-14
変数からの分離 3-2
変数での交換 3-9
方程式の使い方 6-11
目的 2-1, 2-2
レジスタ 2-1

ロール 2-3, C-7
スタック上昇。スタックを参照
影響なし B-5
操作 2-5
デフォルト状態 B-3
無効 B-4
有効 B-4
スタックのロール 2-3, C-6

せ

正規分布 16-11
正弦（三角関数） 4-5, 9-3, A-2, C-6
整数部関数 4-18
正接（三角関数） 4-5, 9-3, A-2, C-6
精度（数値） 1-25, D-13
積分区間 8-2, 15-8, C-8
積分
確度 8-2, 8-6, E-1
区間 8-2, 15-8, C-8, E-7
計算困難な関数 E-2, E-7
結果の不確定値 8-2, 8-6, E-2
サブ区間 E-7
スタックへの結果の保存 8-2, 8-6
制約 15-12
中止 8-2, 15-8
使い方 8-2, C-8
動作 E-1
必要な時間 8-6, E-7
表示形式 8-3, 8-7, 8-8
プログラム 15-10
プログラムの評価 15-7
変数 8-2, C-8
変数の変換 E-8
メモリの使用 8-2
目的 8-1
絶対値（実数） 4-17
切片（カーブフィット） 12-9, 16-1
セルフテスト（電卓） A-4
線形回帰（推定値） 12-8, 16-1
全表記（ALL）。表示形式を参照
設定 1-23
プログラム内の全表記（ALL）
13-7
方程式内の全表記（ALL） 6-5

そ

相関係数 12-7, 16-1
双曲線関数 4-6, C-7
素数生成器 17-6

た

対数カーブフィッティング 16-1
対数関数 4-1, 9-3, C-6
代数モード 1-9
体積の変換 4-14
代入方程式 6-9, 6-11, 6-12, 7-1
多項式 13-26
単位の変換 4-14

ち

チェックサム
 プログラム 13-22
 方程式 6-19, 13-7, 13-23
中間結果 2-12
直交座標と極座標の変換 4-10, 9-5

て

停止。PSE を参照
定数（スタックへの格納） 2-7
ディスプレイ
 コントラストの調整 1-1
 表示される X レジスタ 2-3
テストメニュー 14-7
電源インジケータ 1-1, A-2
電源カーブフィッティング 16-1
電卓
 オンとオフ 1-1
 環境限界 A-2
 コントラストの調整 1-1
 質問 A-1
 接点のショート A-5
 セルフテスト A-5
 デフォルト設定 B-3
 動作テスト A-4, A-5
 リセット A-4, B-2
電卓に関するヘルプ A-1
電卓の湿度限界 A-2
電卓のテスト A-4, A-5
電卓のリセット A-4, B-2
電池 1-1, A-3

と

度
 角度の単位 4-4, A-2
 ラジアンへの変換 4-14
等価方程式 6-9, 6-11, 7-1
統計データ。統計レジスタを参照
統計
 1 変数 12-2
 1 変数データ 12-2
 2 変数 12-2
 2 変数データ 12-2
 演算 12-1
 カーブフィッティング 12-9, 16-1
 クリア 1-4, 12-1
 グループ化されたデータ 16-18
 計算 12-4
 修正 12-3
 初期化 12-3
 精度 12-11
 入力 12-1
 分布 16-11
 変数の総和 12-12
統計変数の総和 12-12
統計メニュー 12-1, 12-4
統計レジスタ。統計データを参照
 アクセス 12-13
 クリア 1-4, 12-1
 初期化 12-3
 総和 12-1, 12-12, 12-13
 データの修正 12-3
 表示 12-12
 分数なし 5-2
トラブルシューティング A-4, A-5

な

長さ変換 4-14

に

入力カーソル
 意味 1-17
 バックスペース 1-4

は

端数処理

SOLVE D-13

三角関数 4-4

積分 8-6

統計 12-11

分数 5-9

バックスペースキー

VIEW のキャンセル 3-4

X レジスタのクリア 2-3, 2-7

操作 1-4

プログラム行の削除 13-20

方程式の入力 1-4

メッセージのクリア 1-4

メニューの終了 1-4, 1-9

ひ

比較テスト 14-7

評価 (統計) 12-8, 16-1

表示形式

積分への影響 8-2, 8-6, 8-7

設定 1-21, A-1

デフォルト B-3

ピリオドとカンマ 1-23, A-1

丸めへの影響 4-18

標準偏差

グループ化されたデータ 16-18

計算 12-6, 12-7

正規分布 16-11

標準偏差メニュー 12-6, 12-7

標本標準偏差 12-6

ピリオド (数値内) 1-23, A-1

ふ

不確定値 (積分) 8-2, 8-6

複素数

演算 9-3

座標系 9-5

スタック 9-2

入力 9-1

引数値 4-17

表示 9-2

符号 (数値の) 1-15, 9-3, 11-6

符号値 4-17

符号の規則 (金融) 17-1

物理定数 4-8

負の数値 1-15, 9-3, 11-6

フラグ

意味 14-9

インジケータ 14-11

演算 14-11

オーバーフロー 14-9

クリア 14-12

セット 14-12

テスト 14-9, 14-12

デフォルト状態 14-9

分数表示 14-10

方程式の評価 14-10

方程式のプロンプト 14-10

割り当てられていない 14-9

フロー図 14-2

プログラム。プログラムラベルを参照

1 行ずつ実行 13-11

ALG 演算 13-4

RPN 演算 13-4

SOLVE 15-1, D-1

SOLVE の使用 15-6

移動 13-11

エラー 13-19

カタログ 1-29, 13-22

間接アドレッシング 14-19,
14-20, 14-21

基数モード 13-25

行の削除 13-20

行の挿入 13-6, 13-20

行番号 13-22

クリア 13-6, 13-22, 13-23

計算 13-13

再開 13-16

削除 1-28

実行 13-10

終了でリターン 13-4

条件テスト 14-7, 14-9, 14-12,
14-16, 15-6

数値 13-7

すべてのクリア 13-6, 13-23

すべての削除 1-5

積分 15-8

積分の使用 15-10

設計 13-3, 14-1

チェックサム 13-22, 13-23, B-2

中断 13-19

- 停止 13-14, 13-16, 13-19
- 停止させない 13-18
- データ出力 13-5, 13-14, 13-18
- データ入力 13-5, 13-13, 13-14
- データのプロンプト 13-13
- テクニック 14-1
- テスト 13-11
- 長い数値の表示 13-7
- 長さ 13-22, 13-23, B-2
- 入力 13-6
- 比較テスト 14-7
- フラグ 14-9, 14-12
- 分岐 14-2, 14-5, 14-6, 14-16
- 分数 5-8, 13-15, 14-9
- 編集 1-4, 13-7, 13-20
- 変数 13-12, 15-1, 15-8
- 方程式 13-4, 13-7
- 方程式の削除 13-7, 13-20
- 方程式の評価 14-11
- 方程式のプロンプト 14-10
- 方程式の編集 13-7, 13-20
- メッセージ 13-16, 13-18
- メモリの使用 13-22
- 目的 13-1
- プログラムできない機能 13-24
- ルーチン 14-1
- ルーチンの呼び出し 14-1, 14-2
- ループ 14-15, 14-16
- ループカウンタ 14-16
- プログラム行。プログラムを参照
- プログラム入力モード 1-4, 13-6
- プログラムのカタログ 1-29, 13-22
- プログラムの実行 13-10
- プログラムポインタ 13-6, 13-11, 13-19, 13-21, B-3
- プログラム名。プログラムラベルを参照
- プログラムラベル
 - 移動 13-22
 - 間接アドレッシング 14-19, 14-20, 14-21
 - クリア 13-6
 - 実行 13-10
 - チェックサム 13-23
 - 名前への入力 1-3
 - 入力 13-4, 13-6
 - 表示 13-22
 - 複製 13-6
 - 分岐 14-2, 14-5, 14-15
 - 目的 13-4
- プロンプト
 - INPUT 13-12, 13-14, 15-2, 15-9
 - 応答 6-14, 13-14
 - 隠れている数字の表示 6-14
 - クリア 1-4, 6-14, 13-15
 - スタックへの影響 6-14, 13-14
 - プログラムでの方程式 14-10, 15-1, 15-9
 - 方程式 6-14
- 分岐 14-2, 14-15, 15-7
- 分数
 - 確度インジケータ 5-2, 5-3
 - 形式 5-6
 - 形式の設定 5-6, 14-10, 14-13
 - 統計レジスタ以外 5-2
 - 入力 1-26
 - 端数処理 5-9
 - 表示 5-2, 5-4, A-2
 - フラグ 14-9
 - プログラム 5-10, 13-15, 14-9
 - 分母 1-26, 5-4, 14-10, 14-13
 - 方程式 5-10
 - 丸め 5-9
 - 約分 5-2, 5-6
- 分数表示モード
 - VIEW への影響 13-15
 - 設定 5-1, A-2
 - 丸めへの影響 5-9
- 分母
 - 最大値の設定 5-4
 - 制御 5-4, 14-10, 14-13
 - 範囲 1-26, 5-2
- 文法（方程式）6-15, 6-19, 13-16
- 平均（統計）
 - 計算 12-4
 - 正規分布 16-11
- 平均メニュー 12-5
- へ
- ベクトル
 - 2 つのベクトル間の角度 10-5
 - 加算、減算 10-1

クロス積 17-10
座標の変換 4-10, 9-5
絶対値 10-3
ドット積 10-4
プログラム 10-6
変数またはレジスタからの
ベクトルの作成 10-7
方程式 10-6
ベッセル関数 8-3
変化率機能 4-6
変換
温度の単位 4-14
角度形式 4-13
角度の単位 4-14
座標 4-10
時間の表示形式 4-13
質量の単位 4-14
数値の基数 10-1, 11-2
体積の単位 4-14
長さの単位 4-14
変換関数 4-10
変数
X との交換 3-8
内側の四則演算 3-6
格納 3-2
カタログ 1-29, 3-5
間接アドレッシング 14-20,
14-21
クリア 1-28
数値の格納 3-1
スタックからの分離 3-2
スタックレジスタの内容への
アクセス B-7
すべてのクリア 1-5
積分 8-2, 15-6, C-8
全桁の表示 13-15
多項式 13-26
解く 7-1, 15-1, 15-6, D-1
名前 3-1
名前の入力 1-3
表示 3-4, 13-15, 13-18
表示中のクリア 13-15

プログラム 13-12, 15-1, 15-7
プログラムの出力 13-15, 13-18
プログラムの入力 13-14
方程式 6-3, 7-1
方程式からの格納 6-11
呼び出し 3-2, 3-4
変数カタログ 1-29, 3-5

ほ

方程式。方程式リストを参照
(I)/(J) 14-21
ALG との比較 13-4
RPN との比較 13-4
TVM 方程式 17-1
値のプロンプト 6-11, 6-14
方程式内の数値 6-5
演算子の優先順位 6-15
応用問題 17-1
括弧 6-6, 6-7, 6-15
関数 6-5, 6-16, G-1
基数モード 6-5, 6-11, 13-25
根 7-1
根なし 7-8
削除 1-5, 6-9
種類 6-9
数値 6-10, 6-11, 7-1, 7-8, 13-4
スクロール 6-7, 13-7, 13-16
スタックの使い方 6-11
積分 8-2
操作のまとめ 6-3
チェックサム 6-19, 13-7, 13-23
使い方 6-1
解く 7-1, D-1
長い式 6-7
長さ 6-18, 13-7, B-2
入力 6-4, 6-7
評価 6-10, 6-11, 6-12, 7-7, 13-4,
14-10
評価の制御 14-13
表示 6-6
複数の根 7-10
プログラムでの方程式 13-4,
13-7, 13-24, 14-11
プログラムでの削除 13-20
プログラムでの入力 13-7

- プログラムでの表示 13-16,
13-18, 14-10
- プログラムでのプロンプト 14-10
- プログラムでの編集 13-7, 13-20
- 分数 5-9
- 文法 6-15, 13-16
- 編集 1-4, 6-8
- 変数 6-4, 7-1
- 変数値の格納 6-12
- メモリ 13-16
- 方程式モード
 - 開始 6-3, 6-7
 - 終了 1-4, 6-3
 - バックスペース 1-4, 6-8
 - プログラム入力中 13-7
 - 方程式リストの表示 6-3
- 方程式リスト
 - EQN インジケータ 6-4
 - 操作のまとめ 6-3
 - 追加 6-4
 - 表示 6-6
 - 編集 6-8
 - 方程式モード 6-3
- 方程式を式として入力 6-10, 6-11, 7-1
- ホーナーの方法 13-26
- 母標準偏差 12-7

ま

丸め

- 数値 4-18
- 分数 5-9, 13-18

め

メッセージ

- 応答 1-28, F-1
- クリア 1-4
- 表示 13-16, 13-18
- 方程式 13-16
- まとめ F-1

メニュー

- 一般的な操作 1-6
- 終了 1-4, 1-9
- 使用例 1-8
- リスト 1-6

メニューキー 1-6

メモリ

- 大きさ 1-28, B-1
- クリア 1-5, 1-29, A-1, A-4,
B-1, B-3
- スタック 2-1
- 使い方 B-1
- 電源オフ時の保持 1-1
- 統計レジスタのクリア 12-3
- フル A-1
- プログラム 13-21, B-2
- プログラムのクリア 1-29, 13-6,
13-22
- 変数のクリア 1-28
- 方程式のクリア 6-8
- 空き容量 1-28
- メモリのクリア A-4, B-3, F-3
- メモリフル B-1, F-3

も

- モード。角度モード、基数モード、
方程式モード、分数表示モード、
プログラム入力モードを参照
- 文字キー 1-2

ゆ

- 優先順位（方程式演算子）6-14

よ

- 余弦（三角関数）4-4, 9-3, C-6

ら

ラジアン

- 角度の単位 4-4
- 角度の単位 A-2
- 度への変換 4-14

- 乱数 4-16, B-3

り

- 利率（金融）17-3

る

ルーチン

- 入れ子 14-3, 15-11
- プログラムの部分 14-1
- 呼び出し 14-1

- ルーチンの入れ子 14-3, 15-11

- ループ 14-16, 14-17

- ループカウンタ 14-16, 14-22

累乗（指数）関数 1-16, 4-3, 9-3

れ

連鎖計算 2-12

連続メモリ 1-1

ろ

論理

AND 11-4

NAND 11-4

NOR 11-4

NOT 11-4

OR 11-4

XOR 11-4