HEWLETT-PACKARD

HP-65

Manuel d'utilisation

HEWLETT hp PACKARD

HP-65

Manuel d'utilisation

SOMMAIRE

INTRODUCTION

Les trois modes d'utilisation du HP-65 3
1. Calcul sans programme 5
2. Utilisation d'un programme préenregistré 10
3. Rédaction de vos propres programmes 12

1. GÉNÉRALITÉS

Effacement 17
Affichage 18
Introduction des grands et des petits nombres 22
Registre Last X 23
Mouvements de la pile opérationnelle 25

2. REGISTRES

Registres mémoire 27
Touches [R↑], [R↓], [x⇄y] 31
Pi (π) 33
Utilisation originale du HP-65 – Evolution d'un placement 33

3. FONCTIONS

Fonctions angulaires 37
Conversions 42
Fonctions de **x** et fonction exponentielle (y^x) 44

4. PROGRAMMATION

Un programme ultra-simple 50
La mémoire-programme 58
Branchement inconditionnel 68

Sous-programmes 74
Décisions logiques 82
Exemples complets de programmation 100
Quelques «trucs» pour améliorer vos programmes 114
Définitions 121

ANNEXES

Annexe A: Limites opérationnelles 125
Annexe B: Accessoires 128
Annexe C: Service et maintenance 130
Annexe D: Erreurs courantes 141

LISTE DES TABLEAUX

Tableau 1-1: Opérations illicites (affichage clignotant) . . . 21
Tableau 3-1: Fonctions angulaires 38
Tableau 3-2: Conversions 43
Tableau 3-3: Fonctions de **x** et fonction exponentielle (y^x) 45

INTRODUCTION

LES TROIS MODES D'UTILISATION DU HP-65

Félicitations pour l'achat de votre HP-65! Outre les fonctions de calcul qui ont fait le succès des calculateurs HP-35 et HP-45, le HP-65 possède une qualité que n'offre aucun autre calculateur de poche: la programmation.

En deux mots, être programmable pour un calculateur, c'est être capable d'apprendre, de retenir et d'exécuter de manière automatique et répétitive une succession d'opérations et d'instructions nécessaires à la résolution d'un problème donné.

L'intérêt de la programmation devient une réelle évidence dès que l'on songe au temps passé à faire et refaire fréquemment le même type de calcul.

Qu'il s'agisse de l'établissement d'un plan de vol, d'un relevé topographique, du calcul d'un rendement ou bien de la conception d'une alimentation électrique, il faut dans tous les cas effectuer des calculs longs et répétitifs qui diminuent notre productivité et diffèrent la réalisation de nos objectifs.

Bien qu'il existe également, et depuis peu, des calculateurs de bureau programmables, leur coût, leur complexité ainsi que l'impossibilité de les transporter facilement, les rendent d'utilisation parfois difficile, voire dans certain cas impossible. L'intérêt essentiel du HP-65 est qu'il libère de ces contraintes et permet pratiquement à tous de bénéficier des avantages d'un calculateur programmable: rapidité d'exécution, précision et souplesse d'utilisation.

Il y a trois manières d'utiliser votre HP-65:

1. CALCUL SANS PROGRAMME

Vous effectuerez chacune des étapes du calcul en appuyant sur les touches dans l'ordre d'exécution réel des opérations: intro-

duction des données, exécution des fonctions, stockage des résultats, affichage, etc.

2. UTILISATION D'UN PROGRAMME PRÉENREGISTRÉ

Par l'emploi de cartes magnétiques préenregistrées (telles que celles fournies avec la bibliothèque «Programmes types» livrée avec votre calculateur), vous pouvez immédiatement traiter des problèmes complexes, facilement et sans connaissance particulière du calculateur lui-même, ou de la programmation. Dans ce cas, il vous suffit simplement d'introduire dans le calculateur la carte préenregistrée correspondant au calcul à effectuer.

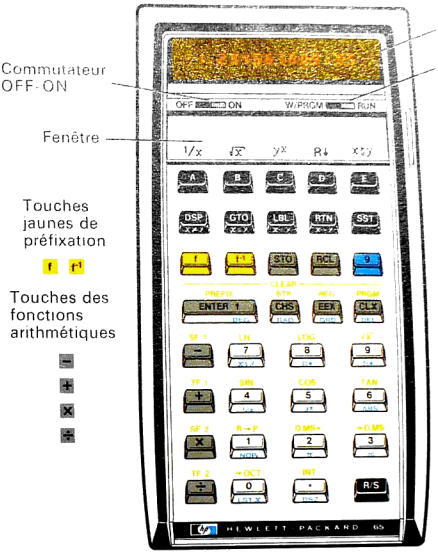
Il ne vous reste alors plus qu'à introduire les données et lancer l'exécution automatique du calcul. Le HP-65 s'arrête dès que le calcul est fini (et affiche le résultat) ou s'il est nécessaire d'introduire de nouvelles données.

3. RÉDACTION DE VOS PROPRES PROGRAMMES

Le HP-65 n'exige aucune connaissance particulière en matière de programmation. Vous pouvez facilement créer les fonctions dont vous avez besoin grâce aux cinq touches de la rangée supérieure, puis les regrouper ou non dans un programme. Un programme n'est rien d'autre que la succession de touches qu'il vous faudrait presser pour obtenir le résultat, plus quelques touches de contrôle de début et de fin de programme.

Pour programmer le HP-65, positionnez le commutateur W/PRGM-RUN sur W/PRGM, puis pressez une à une les touches de la séquence. Vous pouvez alors enregistrer votre programme sur carte magnétique pour l'utiliser ultérieurement.

La suite de l'introduction est consacrée à une étude rapide de ces trois modes de fonctionnement. Ils sont ensuite approfondis au cours des chapitres suivants. Nous vous conseillons d'exécuter sur votre HP-65 les exemples figurant dans ce manuel. En plus du contrôle de son bon fonctionnement, vous vous familiariserez immédiatement avec son emploi.



CLAVIER DU HP-65

1. CALCUL SANS PROGRAMME

OPÉRATIONS PRÉALABLES

Votre calculateur de poche HP-65 vous est livré complet (batterie en place). Si vous envisagez de l'utiliser immédiatement sur batterie, mettez-le préalablement en charge pendant 14 heures (voir Annexe C). Vous pouvez également utiliser votre calculateur pendant que la batterie se recharge. Pour mettre en service votre HP-65:

- Mettre le commutateur W/PRGM-RUN sur la position RUN.
- Mettre le commutateur OFF-ON sur la position ON.

La valeur 0.00 doit alors s'afficher sur l'écran; si ce n'est pas le cas, voir Annexe C.

INTRODUCTION DES DONNÉES

Les chiffres apparaissent sur l'écran d'affichage, de gauche à droite, dans l'ordre où ils ont été introduits. Si le nombre à introduire comporte une virgule, appuyer sur la touche [•]. Par exemple: 314,32 s'introduira ainsi:

[3] [1] [4] [•] [3] [2]

En cas d'erreur pendant la frappe d'un nombre, appuyer sur la touche CLX, puis réintroduire le nombre correctement.

Introduction des nombres négatifs. Pour introduire un nombre négatif, frapper le nombre normalement, puis appuyer sur la touche CHS (changement de signe). Par exemple, pour introduire –12, appuyer sur les touches [1] [2] CHS.

Pour changer le signe d'un nombre négatif ou positif déjà affiché sur l'écran, appuyer sur la touche CHS. Ainsi, pour changer le signe de –12.00 actuellement affiché sur l'écran, appuyer sur la touche CHS.

OPÉRATIONS ARITHMÉTIQUES SIMPLES

Sur le HP-65, le résultat d'une opération arithmétique simple apparaît sur l'écran dès que l'on appuie sur l'une des touches +, –, × ou ÷. Dans tout calcul comportant deux nombres et un opérateur arithmétique, vous devez introduire le premier nombre et le mémoriser en appuyant sur la touche ENTER↑, puis introduire le second nombre et le faire suivre de l'opérateur arithmétique. Pour additionner 12 et 3, appuyer sur les touches

12 ENTER↑ 3 +

Le calculateur utilise le dernier nombre mémorisé et la valeur introduite: il additionne la valeur introduite au nombre mémorisé; il soustrait la valeur introduite du nombre mémorisé; il multiplie la valeur introduite par le nombre mémorisé; il divise le nombre mémorisé par la valeur introduite. Ainsi, pour retrancher 3 de 12, appuyer sur les touches

12 ENTER↑ 3 ×

Pour diviser 12 par 3, appuyer sur les touches

12 ENTER↑ 3 ÷

FONCTIONS NON ARITHMÉTIQUES

Pour utiliser une fonction inscrite en bleu sur le clavier, il est nécessaire d'appuyer préalablement sur la touche préfixe bleue g. De même, pour utiliser une fonction inscrite en jaune sur le clavier, il faut préalablement appuyer sur la touche préfixe jaune f. La touche préfixe jaune f-1 donne l'inverse de la fonction. Par exemple:

Calcul	Touches	Affichage
sin (90°) = 1	90 f SIN	1.00
arc sin (.5) = 30°	.5 f-1 SIN	30.00
1/5 = .2	5 g 1/x	0.20

PILE OPÉRATIONNELLE

Le calculateur HP-65 comporte quatre registres de stockage **X**, **Y**, **Z** et **T**. Ces registres sont disposés en «pile», le registre **X** occupant la position inférieure et le registre **T** la position supérieure.

Contenu (registre)	Nom (registre)
t	T
z	Z
y	Y
x	X

Pour éviter toute confusion entre le nom d'un registre et son contenu, le nom du registre est désigné tout au long de ce manuel par une lettre majuscule et son contenu par une lettre minuscule. Ainsi, **x**, **y**, **z** et **t** sont les contenus respectifs des registres **X**, **Y**, **Z** et **T**. Tout nombre introduit au clavier s'inscrit dans le registre **X** (registre du bas), qui est le seul à être affiché. Lorsque vous appuyez sur la touche ENTER↑, ce nombre est recopié dans le registre **Y** tandis que simultanément **y** passe dans **Z** et **z** dans **T** de la manière suivante:

Appuyer sur	Contenu	Nom
ENTER↑	t → T z → Z y → Y x → X	(perdu)

Le calculateur HP-65 peut conserver un nombre dans chacun des quatre registres.

La plupart des problèmes peuvent être résolus en introduisant les nombres dans l'ordre naturel de lecture de l'expression à calculer, c'est-à-dire de la gauche vers la droite. Pour effectuer un calcul comportant plusieurs nombres, plusieurs opérateurs ou fonctions, commencer par introduire le nombre le plus à gauche dans l'expression. Si une opération directe sur ce nombre est possible (par exemple, sin **x**), exécuter-la immédiatement. Sinon, appuyer sur ENTER↑ pour mémoriser le nombre. Introduire le nombre suivant, le premier nombre est alors en **Y** et le dernier introduit en **X**. De même si une opération est possible entre **X** et **Y**, il faut l'exécuter; sinon, mémoriser la dernière entrée à nouveau et continuer ce processus jusqu'à la fin du calcul.

Utilisation de la pile. Lorsque vous appuyez sur la touche +, les contenus de **X** et **Y** s'additionnent. La pile descend d'un cran: **t** est recopié de **T** en **Z**, **z** descend en **Y**, (**y**+**x**) est transféré dans **X**, et **x** est transféré dans le registre Last X (voir chapitre 1).

Appuyer sur	Contenu	Registre
+	t → T z → Z y → Y x → y+x → X Last X	

Le même processus se produit avec les touches −, × ou ÷ ; le résultat est toujours stocké en X.

Opérations arithmétiques combinées. Chaque fois qu'un nouveau nombre est introduit après une opération, le HP-65 effectue automatiquement une opération ENTER↑ avant d'enregistrer ce nouveau nombre. Ceci vous permet d'effectuer des calculs en série ainsi que des calculs en chaîne et en chaîne mixte, sans avoir à appuyer à chaque fois sur la touche ENTER↑.

Exemple : $[(4 \times 5) \div (2 + 3)] - 6 = -2$

Touches	Affichage
4 ENTER↑	4.00
5 ×	20.00
2 ENTER↑	2.00
3 +	5.00
÷	4.00
6 −	−2.00

Les nombres sont introduits dans l'ordre où ils se présentent dans le problème. Examinons le contenu de la pile au cours de ce calcul.

Registre de la pile

T						20	20				
Z					20	2	2	20		4	
Y		4	4		20	2	2	20		4	
X	4	4	5	20	2	2	3	5	4	6	−2
Touches	4	↑	5	×	2	↑	3	+	÷	6	−

Nota : ENTER↑ est représenté par ↑.

Exemple : $(12 \times 5) + (11 \times 4) + (10 \times 3) = ?$

Touches	Affichage
12 ENTER↑ 5 ×	60.00
11 ENTER↑ 4 × +	104.00
10 ENTER↑ 3 × +	134.00

REGISTRES MÉMOIRES

Le calculateur comporte, en plus des 4 registres en pile et du registre Last X, 9 registres mémoire destinés au stockage de nombre (constantes, résultats intermédiaires, etc.). Des calculs de grande complexité peuvent ainsi être faits sans avoir besoin de réintroduire les constantes ou résultats intermédiaires.

Vous vous sentez déjà plus à votre aise avec votre nouveau calculateur de poche. Nous reviendrons plus loin et plus longuement sur les possibilités de la machine utilisée comme un calculateur non programmable. Nous allons maintenant voir l'utilisation de votre HP-65 avec des cartes magnétiques préenregistrées.

2. UTILISATION D'UN PROGRAMME PRÉENREGISTRÉ

Un programme peut être conservé en permanence sur une carte magnétique. Il est introduit dans la mémoire programme du calculateur grâce au lecteur de carte incorporé à l'appareil. Une telle possibilité augmente immédiatement la puissance du calculateur et certains utilisateurs préfèreront n'employer que les programmes des bibliothèques Hewlett-Packard établis par des spécialistes.

Nous espérons que le 11e programme de notre bibliothèque «Programmes types» livrée avec votre calculateur pourra vous être utile quel que soit le domaine dans lequel vous travaillez. Il traite en effet du classique problème d'investissement à intérêts composés.

Certains programmes de la bibliothèque «Programmes types» sont extraits de bibliothèques également disponibles auprès de Hewlett-Packard.

Par exemple:
- Adaptation d'impédance d'un réseau en π (Bibliothèque Electronique I).
- Moyenne arithmétique, écart type, erreur moyenne (Bibliothèque Statistiques I).

Au fur et à mesure, vous pourrez vous familiariser avec tous ces programmes et exécuter les exemples proposés. Le programme «INTÉRÊTS COMPOSÉS» n'appartient pour le moment à aucune bibliothèque HP. Il a été écrit spécialement à votre intention pour faciliter vos calculs d'investissements. A chaque bibliothèque est joint un fascicule d'utilisation dans lequel sont donnés la description des programmes, les formules utilisées, leur mode opératoire, des exemples d'applications et le listing des pas de programmes.

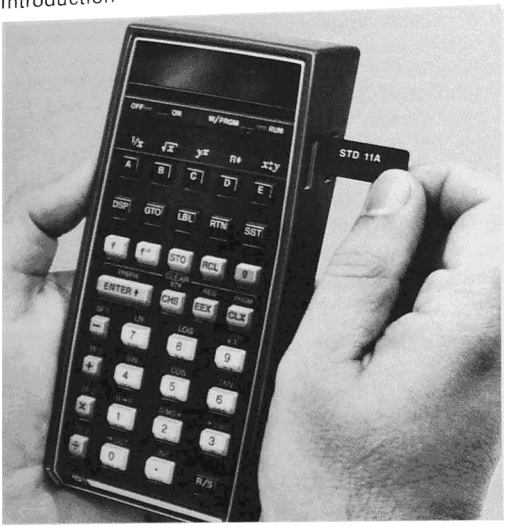
MISE EN MÉMOIRE D'UN PROGRAMME

1. Prendre dans l'étui de la bibliothèque «Programmes types» la carte STD 11A (programme: «INTÉRÊTS COMPOSÉS»).

2. Placer le commutateur W/PRGM-RUN sur la position RUN.
3. Introduire doucement la carte (côté imprimé au-dessus) dans la fente inférieure droite. Dès que la carte est partiellement engagée, le moteur l'entraîne automatiquement et la fait ressortir du côté gauche du calculateur. Si le moteur tourne sans entraîner la carte, enfoncer légèrement la carte dans la machine. Ne jamais forcer, ni empêcher son mouvement. (Voir photo page 12.)
4. En cas de lecture incorrecte, l'affichage clignote: appuyer alors sur la touche **R/S**, puis réintroduire la carte.
5. A l'arrêt du moteur, retirer la carte à gauche du calculateur et l'introduire dans la fente supérieure droite.

Le programme est maintenant mis en mémoire.

Exemple: Quelle somme doit-on investir maintenant pour posséder 15 000 francs au bout de 20 années avec taux d'intérêts de 7% et composition trimestrielle des intérêts?



Pour résoudre ce problème, reportez-vous au mode opératoire donné ci-après. Lire les «instructions» ligne par ligne, introduire les données nécessaires, puis appuyer sur la ou les touches indiquées.

N°	Instructions	Données	Touches		Résultats
1	Introduire le programme				
2	Initialiser		RTN	R/S	0.00
3	Introduire				
	n(n=20×4)	80	A		80.00
	i(i=7÷4)	1.75	B		1.75
	FV	15000	D		15000.00
4	Calcul de PV		E	C	3744.02

Résultat : 3744,02 francs

Avec le HP-65 est fourni un cahier de 50 feuilles de programmation à remplir lors de la création de vos propres programmes.

3. RÉDACTION DE VOS PROPRES PROGRAMMES

PROGRAMMATION – TOUCHES À DÉFINIR

Des calculs très complexes peuvent être programmés par la pression d'une succession de touches. Etant donné que le calculateur HP-65 est totalement programmable, et comporte notamment des fonctions de branchement et de test, il est tout à fait possible de créer un programme effectuant des itérations pendant toute une nuit. Un calcul peut occuper jusqu'à 100 pas de programme.

Nous avons vu comment les fonctions des cinq touches de la rangée supérieure pouvaient être définies en fonction d'une application particulière par la lecture d'une carte magnétique préenregistrée. La touche **A** commande l'exécution de la fonction programmée. Dans l'exemple qui suit, définir d'abord la fonction, puis l'introduire en mémoire pour la tester. Si les tests sont concluants, enregistrer la fonction sur une carte magnétique pour une éventuelle utilisation ultérieure.

ANALYSE DES OPÉRATIONS

La séquence suivante calcule k^3 (cube de toute valeur **k** se trouvant dans le registre **X**).

T				
Z		k		
Y	k	k	k	
X	k	k	k^2	k^3
Touche	↑	↑	×	×

Note: Dans cet exemple, la touche ENTER↑ est représentée par ↑.

Pour faire de cette séquence une fonction programmée qui puisse être exécutée par la touche A, il faut identifier le début de la séquence par les touches supplémentaires LBL A ainsi que la fin par la touche RTN.

MISE EN MÉMOIRE DE LA FONCTION

1. Mettre le commutateur W/PRGM-RUN sur la position W/PRGM et appuyer sur les touches f PRGM pour effacer la mémoire programme.

2. Appuyer sur les touches ci-dessous dans l'ordre indiqué:

Touches	Commentaires
LBL	Définition du début de la fonction A
A	
ENTER↑	
ENTER↑	Calcul de x^3
×	
×	
RTN	Définition de la fin de la fonction A

En cas d'erreur, effacer le programme, puis recommencer. Nous verrons par la suite qu'il est possible de corriger les erreurs et quelle est la signification des nombres apparaissant sur l'écran.

Votre HP-65 sait désormais calculer x^3, lorsque vous appuyer sur A en mode RUN.

VÉRIFICATION DE LA FONCTION

- Mettre le commutateur W/PRGM-RUN sur la position RUN.
- Introduire un nombre, puis appuyer sur **A** : affichage du cube de ce nombre.

Touches	Affichage	Commentaires
2 **A**	8.00	2^3
3 **A**	27.00	3^3
4 **A**	64.00	4^3
5 **CHS** **A**	−125.00	$(-5)^3$

ENREGISTREMENT SUR UNE CARTE MAGNÉTIQUE DE LA FONCTION

Pour enregistrer le programme, procéder comme suit:

1. Prendre une carte magnétique non protégée (coin non coupé).

Carte non protégée

Carte protégée

2. Mettre le commutateur W/PRGM-RUN sur la position W/PRGM (W signifie Write).
3. Introduire la carte dans la fente inférieure droite, exactement comme pour la mise en mémoire d'un programme (voir précédemment). Si la carte n'est pas protégée, le contenu de la mémoire programme du HP-65 est recopié sur la carte. Dans l'exemple ci-dessus, nous n'avons pas défini les touches **B** ... **E**, mais nous aurions pu également les définir.

Nous venons de voir comment on peut rédiger un programme pour une fonction simple et l'identifier au moyen de l'une des cinq touches définissables par l'utilisateur. La programmation de votre HP-65 peut également s'effectuer sans faire appel aux touches de la rangée supérieure. Nous en verrons par la

suite un exemple au chapitre programmation. Il est très facile de créer des fonctions simples, et guère plus difficile de créer des fonctions ou d'autres programmes complexes.

EN AVANT...

Cette introduction avait pour but de vous présenter succinctement les 3 modes d'utilisation de votre calculateur de poche HP-65. Dans les chapitres suivants, ces modes sont décrits en détail. Nous vous conseillons de lire les résumés encadrés: ceux-ci sont un guide précieux. Vous vous familiariserez rapidement avec le clavier et prendrez rapidement de l'assurance en effectuant les nombreux exemples de calcul présentés dans ce manuel.

Les chapitres **1** à **3** exposent l'utilisation du calculateur en mode manuel.

Chapitre 1: affichage et introduction des données en notation scientifique.
Chapitre 2: utilisation des registres mémoire et manipulation de la pile.
Chapitre 3: utilisation des fonctions préprogrammées.

Le chapitre **4** est consacré à la programmation: test, saut conditionné, processus répétitif, mise au point d'un programme, emploi des feuilles de programmation livrées avec le calculateur, etc.

L'AIDE-MÉMOIRE

Cet aide-mémoire résume les procédures les plus importantes exposées dans ce manuel et résume l'emploi des touches, par ordre alphabétique. Nous vous conseillons de mettre cet aide-mémoire dans l'étui de votre calculateur pour pouvoir le consulter à tout moment.

CHAPITRE 1

GÉNÉRALITÉS

Ce présent chapitre est consacré à la description des fonctions d'effacement, des différents modes d'affichage, de l'introduction de nombre (données), de la correction d'erreurs, ainsi que de l'emploi du registre Last X et de la pile opérationnelle.

EFFACEMENT

Le HP-65 dispose de quatre possibilités d'effacement distinctes, regroupées sous la touche préfixe **f**.

EFFACEMENT D'UN PRÉFIXE

Les touches **f** [PREFIX] annulent l'effet d'un préfixe. Supposons que vous ayez appuyé par erreur sur l'une des touches **f**, **f^{-1}** ou **g** avant d'introduire un nombre en **X**. Si vous appuyez ensuite sur une touche numérique, vous obtiendrez en fait l'exécution de la fonction de cette touche, au lieu de l'introduction d'un chiffre. Pour annuler l'effet de la touche préfixe, appuyez sur les touches **f** [PREFIX], puis introduisez alors le nombre. Par contre, si vous vous trompez de touche préfixe, rectifiez l'erreur en appuyant simplement sur la touche préfixe correcte: ceci annule l'effet de la première touche préfixé.

Les touches suivantes sont également des touches de préfixation:

STO, **RCL**, **DSP**, **GTO**, **LBL**

EFFACEMENT DES REGISTRES DE LA PILE OPÉRATIONNELLE

Les touches **f** [STK] effacent l'ensemble des quatre registres (**X**, **Y**, **Z** et **T**) de la pile opérationnelle. La touche **CLX** n'efface que le contenu du registre **X**.

EFFACEMENT DES REGISTRES MÉMOIRE

Les touches f REG effacent l'ensemble des neuf registres mémoire (voir chapitre 2).

EFFACEMENT GÉNÉRAL

Pour vider l'ensemble des mémoires et des registres du calculateur, mettre le commutateur OFF/ON sur la position OFF, puis sur la position ON. Toutefois, lors de la mise sous tension, les fonctions figurant au-dessus des touches A, B, C, D, E (1/x, √x, y^x, R↓, x⇄y) sont stockés automatiquement dans la mémoire programme.

EFFACEMENT DE LA MÉMOIRE PROGRAMME

Mettre le commutateur W/PRGM-RUN sur la position W/PRGM et appuyer sur les touches f PRGM : la mémoire programme est alors entièrement effacée (100 pas). Si ce commutateur est sur la position RUN, les touches f PRGM ont le même effet que la touche CLx.

AFFICHAGE

L'écran sert à l'affichage des résultats ainsi que des codes des pas programme stockés en mémoire (mettre le calculateur en mode W/PRGM). Il est également employé pour indiquer une erreur de fonctionnement, l'exécution d'une opération illicite, d'un programme en cours d'exécution ainsi que la fin de charge de la batterie.

MODES D'AFFICHAGE

L'écran du HP-65 affiche jusqu'à 15 caractères: signe de la mantisse, mantisse à 10 chiffres, point décimal, signe de l'exposant et exposant à 2 chiffres. Quel que soit le mode d'arrondi choisi, la précision de calcul n'est en aucun cas affecté. Le calculateur conserve toujours, de façon interne, les nombres et

valeurs calculés sur dix chiffres significatifs exacts pour la mantisse, et deux chiffres pour l'exposant.

Affichage en virgule flottante. Cet affichage s'obtient en appuyant sur les touches DSP •, puis sur la touche numérique correspondant au nombre de décimales (0 à 9) désiré. Cette notation permet d'obtenir tous les résultats avec le même nombre de chiffres significatifs voulu après la virgule. L'affichage est cadré de gauche à droite et donne les zéros non significatifs nécessaires pour atteindre le nombre de décimales voulu selon l'arrondi choisi. Lorsque le commutateur OFF/ON est mis successivement sur les positions OFF et ON, le calculateur se prépositionne toujours à deux décimales. Exemple:

Touches:	Affichage
S'assurer que le commutateur W/PRGM-RUN est sur la position RUN. Mettre le commutateur OFF/ON sur OFF, puis ON.	0.00
123.4567 ENTER↑	123.46
DSP • 4	123.4567
DSP • 6	123.456700
DSP • 2	123.46
DSP • 0	123.

Affichage en notation scientifique. Cet affichage est particulièrement intéressant lors de calculs de nombres très grands ou très petits. Pour travailler en notation scientifique, appuyer sur la touche DSP, puis sur la touche numérique correspondant au nombre de décimales souhaitées (de 0 à 9). Comme précédemment, l'affichage se fait de gauche à droite et comprend les zéros non significatifs, selon l'arrondi choisi.

Touches	Affichage		Commentaires
Mettre le commutateur OFF/ON sur OFF, puis ON	0.00		
123.4567 ENTER↑	123.46		
DSP 2	1.23	02	$= 1.23 \times 10^{2}$
DSP 4	1.2346	02	$= 1.2346 \times 10^{2}$
DSP 8	1.23456700	02	$= 1.234567 \times 10^{2}$

Revenir à huit décimales en notation virgule flottante.

Touches	Affichage	Commentaires
DSP • 8	1.234567000 02	* = 1.234567×10^2

Revenir à deux décimales en notation virgule flottante:

Touches	Affichage
DSP • 2	123.46
DSP 0005 CHS ENTER↑	−0.00 **

CLIGNOTEMENT DE L'AFFICHAGE

L'écran clignote en cas d'exécution d'opérations illicites. Pour arrêter le clignotement, appuyer sur la touche CLX ou sur tout autre touche qui n'entraînerait pas une nouvelle opération illicite. Les opérations illicites sont données au Tableau 1-1.

AFFICHAGE NON LISIBLE

Pendant l'exécution d'un programme enregistré, l'affichage évolue en permanence; ceci est intentionnel et a pour objet d'indiquer que le programme est en cours d'exécution. Dès l'arrêt du programme, l'affichage redevient lisible.

AFFICHAGE DE POINTS DÉCIMAUX MULTIPLES

La batterie assure une autonomie de fonctionnement d'environ 3 heures. En coupant le contact lorsque le calculateur n'est pas en service, vous économiserez la batterie. Pour économiser la batterie sans perdre un programme ou des résultats, laissez le calculateur sous tension, introduisez un point • et laissez-le dans cet état jusqu'à la reprise des calculs.

Lorsque la batterie ne permet plus que de deux à cinq minutes de travail, tous les points décimaux s'allument sur l'écran.

* Si un nombre est trop grand compte tenu de l'affichage défini, ce nombre est automatiquement affiché en notation scientifique.

** Si un résultat est trop petit, compte tenu de l'affichage défini, c'est la valeur zéro qui est affichée (accompagnée du signe – dans le cas d'un résultat négatif).

Touches	Fonction	Erreur
f LN	Logarithme népérien (base e)	$x \leq 0$
f LOG	Logarithme décimal (base 10)	$x \leq 0$
f $\sqrt{x}$	Racine carrée	$x < 0$
f-1 SIN	Arc sinus	$\|x\| > 1$
f-1 COS	Arc cosinus	$\|x\| > 1$
f / f-1 D.MS+	Addition / Soustraction degrés, minutes, secondes	$\|x\|$, $\|y\|$ ou $\|y \pm x\| > 99999.59599$ D.MS
f / f-1 →D.MS	Conversion d'angles décimaux en sexagésimaux (degrés, minutes, secondes) et vice versa	$\|x\| > 99999.99999$ degrés ou équivalent en radians ou grades
f →OCT	Décimal en octal	x n'est pas un entier ou $\|x\| > 1073741823_{10} = 7777777777_8$
f-1 →OCT	Octal en décimal	x n'est pas un entier ou $\|x\| > (12222222221)_8 = 9999999999_8 = 1380525201_{10}$
g 1/x	Inverse	$x = 0$
g y^x	Exponentielle	$y \leq 0$
g n!	Factorielle	x n'est pas un entier ou $x < 0$
■	Division	$x = 0$
	Lecture de carte magnétique	Carte vierge; lecture incorrecte

Tableau 1-1. Opérations illicites (clignotement de l'affichage).

Même dans ce cas, la position réelle de la virgule est connue, car un emplacement entier lui est réservé.

Exemple:

De même, si, lors du fonctionnement du moteur du lecteur de cartes magnétiques, les points décimaux s'allument puis disparaissent, cela signifie que la batterie est presque déchargée.

La poursuite des opérations plus de deux à cinq minutes après le signal de fin de charge de la batterie peut entraîner des erreurs de calcul. Il faut alors recharger la batterie en branchant le HP-65 sur son chargeur. Ne pas oublier de couper l'alimentation du calculateur avant de brancher le chargeur au calculateur.

INTRODUCTION DES GRANDS ET DES PETITS NOMBRES

Vous pouvez introduire des nombres correspondant à une puissance de 10 au moyen de la touche EEX (introduction des exposants). Exemple: Introduire 15.6 trillions ($15{,}6 \times 10^{12}$), puis multiplier par 25.

Touches	Affichage		Commentaires
15.6 EEX	15.6	00	
12	15.6	12	15.6×10^{12} *
ENTER↑	1.560000000	13	1.56×10^{13}
25 ×	3.900000000	14	Réponse

* Pour introduire un nombre négatif (par exemple -15.6×10^{12}), appuyer sur la touche CHS avant la touche EEX.

PUISSANCE EXACTE DE DIX

Pour introduire un nombre positif qui est une puissance exacte de 10, appuyer d'abord sur la touche EEX, puis introduire la puissance 10. Exemple: Introduire 1 million (10^6), puis diviser par 52.

Touches	Affichage
EEX 6	1. 06
ENTER↑	1000000.00
52 ÷	19230.77

Pour obtenir ce résultat en notation scientifique avec 6 décimales:

Touches	Affichage
DSP 6	1.923077 04

PETITS NOMBRES (EXPOSANTS NÉGATIFS)

Pour introduire des exposants négatifs, introduire d'abord le nombre, appuyer successivement sur les touches EEX et CHS afin d'obtenir un exposant négatif, puis introduire la puissance de 10. Exemple: Introduire la constante de Planck (h) environ: $6{,}625 \times 10^{-27}$ erg/sec et la multiplier par 50.

Touches	Affichage
6.625 EEX	6.625 00
27	6.625 27
CHS	6.625 −27
ENTER↑	6.625000 −27
50 ×	3.312500 −25

En appuyant sur les touches DSP • 2, le résultat sera arrondi à zéro.

REGISTRE LAST X

Le registre Last X permet non seulement de corriger facilement les erreurs d'introduction de données, mais offre en outre la possibilité de réutiliser, lors de calculs multiples, un même

argument, ce dernier pouvant être rappelé à tout instant en appuyant successivement sur les touches g et LST x. En effet, à chaque nouvelle opération, le dernier argument d'entrée d'un calcul quelconque est automatiquement mis en mémoire dans le registre Last X. Ce registre est effacé en cas d'arrêt du calculateur ou bien si un nouvel argument remplace le précédent.

Ainsi, si vous exécutez un long calcul dans lequel vous vouliez soustraire la valeur 3 de la valeur 12 et que vous ayez exécuté une division à la place, vous pourrez rectifier de la façon suivante:

Touches	Affichage	Commentaires
12 ENTER↑ 3 ÷	4.00	Vous vouliez soustraire et non pas diviser.
g LST x	3.00	Rappel du dernier nombre affiché avant l'opération de division.
×	12.00	Inversion de l'opération de division: vous vous retrouverez donc au point de départ initial.
g LST x	3.00	Rappel du dernier nombre affiché avant l'opération de multiplication.
−	9.00	L'opération correcte donne le résultat souhaité.

Si vous voulez corriger un nombre au cours d'un calcul, le registre Last X vous évite de tout recommencer. Par exemple: diviser 12 par 2,157 après l'avoir divisé par 3,157 par erreur.

Touches	Affichage	Commentaires
12 ENTER↑ 3.157 ÷	3.80	Vous vouliez diviser par 2,157 et non par 3,157.
g LST x	3.16	Rappel du dernier nombre affiché avant l'opération.
×	12.00	Vous êtes revenu à votre point de départ.
2.157 ÷	5.56	L'opération correcte donne le résultat souhaité.

D'autre part, le registre Last X peut être utilisé lors des calculs dans lesquels x figure plusieurs fois. Nous en donnerons quelques exemples, sans entrer dans les détails, puisque nous n'avons pas encore vu ces fonctions. Exemples:

$$\frac{\sin x}{x},\ y^x - \sqrt{x},\ \sin x + \cos^3 x$$

x est conservé dans le registre Last X après l'exécution de la première opération.

Les opérations suivantes (y compris les opérations inverses) conservent **x** dans le registre Last X: [+], [−], [×], [÷], [→D.MS], [D.MS+], [INT], [LN], [LOG], [→OCT], [R→P], [SIN], [COS], [TAN], [n!], [√x], [1/x], [y^x], [ABS]. La touche [CLX] est sans effet sur le registre Last X.

MOUVEMENTS DE LA PILE OPÉRATIONNELLE

Comme nous l'avons vu précédemment, l'introduction d'un nouveau nombre à la suite d'un calcul «monte» automatiquement le résultat du calcul dans le registre **Y** sans qu'il soit nécessaire de le mémoriser par [ENTER↑]. Le même mouvement se produit si vous rappelez une valeur en **X** à partir d'un registre mémoire, du registre Last X, ou si vous rappelez la valeur de Pi (préprogrammé). Par contre, tout nombre introduit après [ENTER↑] et [CLX] est placé dans le registre **X** (pas de montée de pile).

Vous n'avez pas à vous préoccuper de ces mouvements: ils s'effectuent automatiquement dans la plupart des calculs. Toutefois, à titre indicatif, les touches provoquant la montée de la pile sont indiquées dans l'encadré ci-après. Un grand nombre d'opérations n'ont aucun effet sur les mouvements de la pile. La plupart des opérations ne vous ont pas encore été présentées, mais elles le seront dans les pages suivantes de ce manuel.

Opérations n'entraînant pas de mouvement dans la pile: Touche R/S si elle est immédiatement précédée par la génération d'un nombre en x par le programme. Introduction d'un nombre après pression de ENTER↑ ou CLx.

Opérations entraînant des mouvements dans la pile: Toutes les touches de chiffres: 0 ... 9, •, EEX, π, mais pas CHS. • Toutes les touches de calcul: −, +, ×, ÷, ABS, COS, →D.MS, D.MS+, INT, LN, LOG, LST x, n!, →OCT, R→P, SIN, TAN, 1/x, √x, y^x. • Touches de manipulation de pile: R↑, R↓, x⇄y, sauf ENTER↑. • Touches des registres mémoire: STO RCL.

Opérations ne modifiant pas l'état de la pile: Les autres touches, en particulier toutes les touches de programmation, les touches de calcul d'angle, les touches d'affichage, les touches d'effacement (sauf CLx), et la touche CHS n'ont pas d'effet sur l'état de la pile.

CHAPITRE 2

REGISTRES

Dans ce chapitre est décrite l'utilisation des neuf registres mémoire ainsi que des touches de manipulation des données contenues dans la pile (R↓, R↑, x⇄y). De plus, pour illustrer une utilisation des registres de la pile opérationnelle, nous donnerons ci-après la solution manuelle d'un problème d'intérêts composés.

REGISTRES MÉMOIRE

Le calculateur HP-65 dispose de 9 registres mémoire R_1 à R_9. Pour éviter toute confusion entre le nom d'un registre et son contenu, un registre est désigné par une majuscule et son contenu par une minuscule; ainsi r_1, $r_2 \ldots r_9$ sont les contenus respectifs des registres R_1, $R_2 \ldots R_9$.

Ces registres servent essentiellement à cumuler des sommes ou à stocker des constantes ou des résultats intermédiaires. Il est possible de stocker la valeur du registre **X** de la pile dans l'un quelconque des registres mémoire ou de rappeler dans le registre **X** la valeur figurant dans l'un quelconque des registres mémoire. De plus, on peut stocker dans tout registre la somme arithmétique ainsi que la différence, le produit ou le quotient du registre désigné et du registre **X**. Ainsi, si le registre R_5 contient la valeur 100 et le registre **X** la valeur 70, on peut stocker la différence $(100-70=30)$ dans le registre R_5 en appuyant simplement sur les touches STO − 5.

STOCKAGE ET RESTITUTION DES DONNÉES

Pour stocker une valeur affichée (suite à un calcul ou à une introduction au clavier), procéder comme suit:

1. Appuyer sur la touche STO.

2. Appuyer sur une touche numérique [1] à [9] afin d'identifier le registre mémoire. Si le registre mémoire sélectionné contient déjà une valeur, cette valeur est effacée. La valeur se trouvant en **X** reste inchangée.

Pour rappeler un nombre stocké dans l'un des neuf registres mémoire:

1. Appuyer sur la touche RCL.
2. Appuyer sur la touche numérique [1] à [9] d'identification du registre.

Lorsqu'on rappelle un nombre, celui-ci n'est pas effacé de son registre: c'est une copie du nombre stocké qui est transférée sur l'écran, l'original restant dans le registre mémoire jusqu'à ce qu'intervienne une des opérations suivantes: ① stockage d'un nouveau nombre dans le même registre; ② mise hors service du calculateur ou ③ effacement de l'ensemble des neuf registres mémoire au moyen des touches f REG. Le rappel d'un nombre contenu dans un registre fait monter la pile, à moins que cette opération ne soit invalidée.

Invalidation de la montée de la pile opérationnelle: la pile ne monte pas automatiquement lors de l'introduction ou du rappel d'un nombre si la dernière touche pressée est ENTER↑ ou CLX. Dans ce cas, le nombre introduit remplace le contenu du registre **X**.

Exemple 1: Une personne achète trois articles coûtant respectivement 1000, 2000 et 3000 francs. Une remise de 5% étant accordée pour tout achat supérieur à 5000 francs, quels seront le prix réel des différents articles et le coût total?

Solution:

Touches	Affichage	Commentaires
1 ENTER↑ .05 − STO [1]	0.95	Stockage de la constante 0.95 (soit 95%) dans le registre R_1.

1000 RCL 1 ×	950.00	Coût d'achat réel du premier article.
2000 RCL 1 ×	1900.00	Coût d'achat réel du second article.
3000 RCL 1 ×	2850.00	Coût d'achat réel du troisième article.
+ +	5700.00	Coût total.

Exemple 2: La capacité et la hauteur de trois réservoirs sont données ci-dessous en unités américaines. Quelles seront la capacité et la hauteur de chaque réservoir dans le système métrique?

	Capacité (gallons)	**Hauteur** (pouces)
Réservoir 1	3.6	13.5
Réservoir 2	5.5	20.9
Réservoir 3	11.3	32.8

1 gallon américain = 3,7854 litres
1 pouce = 2,54 cm

Stockons ces constantes dans les registres R_1 et R_2.

Touches	**Affichage**	**Commentaires**
DSP • 4		
3.7854 STO 1	3.7854	Stockage de la constante de conversion litres/gallons dans R_1.
2.54 STO 2	2.5400	Stockage de la constante de conversion centimètres/pouces dans R_2.
3.6 RCL 1 ×	13.6274	Capacité du réservoir 1 en litres.
13.5 RCL 2 ×	34.2900	Hauteur du réservoir 1 en centimètres.
5.5 RCL 1 ×	20.8197	Capacité du réservoir 2 en litres.
20.9 RCL 2 ×	53.0860	Hauteur du réservoir 2 en centimètres.
11.3 RCL 1 ×	42.7750	Capacité du réservoir 3 en litres.

32.8 RCL 2 ×	83.3120	Hauteur du réservoir 3 en centimètres.
DSP • 2	83.31	Affichage à 2 décimales.

CHOIX DES REGISTRES MÉMOIRE

En dehors des cas particuliers des registres R_8 et R_9, les registres mémoire s'emploient sans aucune restriction.

Le registre R_8 sert lors de l'utilisation du compteur programmé (touche DSZ) (voir chapitre 4). Dans ce cas, il faudra éviter d'avoir recours à ce registre pour d'autres utilisations. Dans le cas contraire, ce registre est normalement utilisé.

Le registre R_9 est inutilisable lors des calculs trigonométriques, des conversions de coordonnées polaires/rectangulaires et lors des tests de comparaison (utilisés en programmation). A ces exceptions près, le registre R_9 est normalement utilisé.

Les opérations suivantes effacent le contenu du registre R_9 :
SIN, COS, TAN, R→P (fonctions trigonométriques et leur inverse) ;
x≠y, x≤y, x=y, x>y (tests de comparaison).

ARITHMÉTIQUE DIRECTE ENTRE REGISTRES ET PILE

Jusqu'ici tous les calculs ont fait intervenir le registre **X** ou les registres **X** et **Y** pour obtenir un résultat en **X**. On peut faire des opérations directement sur des registres mémoire ; le résultat s'inscrit dans le registre mémoire et **x** demeure inchangé.

Soustraction :
Pour soustraire **x** de r_n, appuyer sur STO − n
Addition :
Pour ajouter **x** à r_n, appuyer sur STO + n
Multiplication :
Pour multiplier **x** et r_n, appuyer sur STO × n
Division :
Pour diviser r_n par **x**, appuyer sur STO ÷ n

Exemple: stocker la valeur 6 dans le registre R_1, puis lui ajouter 2.

Touches	Affichage	Commentaires
6 STO 1	6.00	Stockage de 6 dans le registre R_1.
2 STO + 1	2.00	Addition de 2 à r_1.
RCL 1	8.00	Confirmation: $r_1 = 8$.

Soustraire ensuite la valeur 5 du contenu de ce registre:

5 STO − 1	5.00	
RCL 1	3.00	Confirmation: $r_1 = 3$

Enfin, multiplier par 2 la nouvelle valeur du registre R_1:

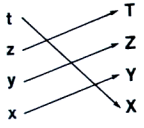
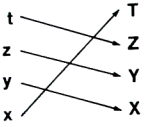
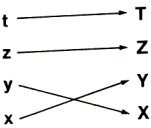
2 STO × 1	2.00	
RCL 1	6.00	Confirmation: $r_1 = 6$

TOUCHES R↑, R↓ ET x⇄y

Tous les calculs sont exécutés dans la pile à l'exception des calculs portant sur les registres mémoire. Il suffit d'introduire le ou les nombres et d'appuyer sur la touche de fonction appropriée. Vous savez également qu'il est possible de conserver des résultats intermédiaires dans la pile, au-dessus des calculs en cours. Pour faire monter la pile, vous avez utilisé la touche ENTER↑ ainsi que la montée automatique. Pour faire descendre la pile, vous avez utilisé les fonctions arithmétiques.

Ci-après sont décrites les touches R↑, R↓ et x⇄y qui permettent de vérifier, par affichage sur l'écran, le contenu des registres de la pile (en dehors du registre **X**) et de «jongler» avec les données:

Touche R↑: affichage du contenu **t** du registre **T**
Touche R↓: introduction de **x** dans le registre **T**
Touche x⇄y: permutation de **x** et **y**

R↓ Montée	R↓ Descente	x⇄y Permutation
		

Les fonctions R↑, R↓ et x⇄y figurent sur les touches 9, 8 et 7.

Vous remarquerez qu'à la mise sous tension initiale, les symboles R↓ et x⇄y existent également sur les touches D et E. Les cinq fonctions les plus utilisées (1/x, √x, y^x, R↓ et x⇄y) ont été regroupées dans la fenêtre: elles servent essentiellement à l'utilisation manuelle du clavier comme dans le cas présent. Chacune permet l'exécution, en appuyant sur une seule touche, de fonctions qui nécessiteraient normalement la pression de deux touches. Lorsque les touches A, ..., E sont redéfinies dans un programme, les fonctions figurant dans la fenêtre sont néanmoins accessibles par la pression de deux touches.

Exemple. Mettre le commutateur OFF/ON sur la position OFF, puis sur la position ON. Ranger les valeurs 4, 3, 2 et 1 respectivement dans les registres **T**, **Z**, **Y** et **X** et afficher le contenu de la pile au moyen des touches suivantes: x⇄y, E, R↓, D et R↓.

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

Touches: 4 ↑ 3 ↑ 2 ↑ 1

La touche ENTER↑ est représentée par ↑.

T	4	4	4	4
Z	3	3	3	3
Y	1	2	1	2
X	2	1	2	1
Touches:	g x⇄y	g x⇄y	E	E
	Utilisation g x⇄y		Utilisation E	

Remarquer que l'utilisation des touches g x⇄y ou de la touche E deux fois repositionne la pile à son état initial.

T	1	2	3	4	1	2	3	4
Z	4	1	2	3	4	1	2	3
Y	3	4	1	2	3	4	1	2
X	2	3	4	1	2	3	4	1
Touches:	g R↓	g R↓	g R↓	g R↓	D	D	D	D
	Utilisation g R↓				Utilisation D			

Remarquer que l'utilisation des touches g R↓ ou de la touche D quatre fois repositionne la pile à son état initial.

T	3	2	1	4
Z	2	1	4	3
Y	1	4	3	2
X	4	3	2	1
Touches:	g R↑	g R↑	g R↑	g R↑

Remarquer que l'utilisation de la touche R↑ quatre fois repositionne la pile à son état initial.

PI (π)

Pi est une des constantes programmées du HP-65. Pour l'utiliser, appuyer sur les touches g π.

Exemple : Quelle est la superficie S d'un cercle de 3 mètres de rayon ? $S = \pi 3^2$.

Touches	Affichage	Commentaires
g π	3.14	Rappel de π en **X**
3 ENTER↑ ×	9.00	Calcul de 3×3
×	28.27	Réponse: $9 \times \pi$

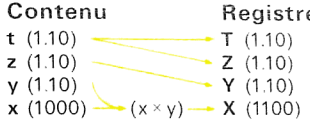
UTILISATION ORIGINALE DU HP-65 – ÉVOLUTION D'UN PLACEMENT

Les calculateurs de poche Hewlett-Packard laissent libre cours à l'innovation et à la créativité de l'utilisateur. Certains de nos clients sont parvenus à résoudre avec leur HP-65 des problèmes au moyen de méthodes de calcul originales. Un de ces problèmes – calcul d'une progression géométrique dans le cas de l'évolution d'un placement – est décrit ci-après.

Une progression géométrique est une suite de nombres dans laquelle chaque terme est calculé en multipliant le terme précédent par un facteur donné. Exemple: 4, 8, 16, 32, etc. Dans ce cas, le facteur est 2. Sur un plan pratique, l'évolution d'un capital de 1000 francs placé à 10% par période constituera une progression géométrique dans laquelle le premier terme est 1000 et le facteur de progression 1,10. La solution adoptée par notre client est la suivante:

Touches	Affichage	Commentaires
1.10 ENTER↑ ENTER↑ ENTER↑ 1000	1000.00	Montant initial
×	1100.00	Montant après une période
×	1210.00	Montant après deux périodes
×	1331.00	Montant après trois périodes
×	1464.00	Montant après quatre périodes
×	1610.00	Montant après cinq périodes
×	1771.56	Montant après six périodes
etc...		

Mettre le facteur de progression (1.10) dans les registres **Y**, **Z** et **T** de la pile opérationnelle et le premier terme (1000) dans le registre **X**. Ensuite, pour obtenir le terme suivant, appuyer sur la touche [×]. Lorsqu'on appuie sur [×] pour la première fois, on calcule 1000 × 1.10 (soit x × y). Le résultat (1100.00) est affiché dans le registre **X** et une nouvelle copie du facteur de progression (1.10) descend dans le registre **Y**. Il est inutile de réintroduire le facteur de progression, ce dernier étant automatiquement introduit chaque fois que la pile descend.



Contenu de la pile avant et après la première multiplication

CHAPITRE 3

FONCTIONS

Vous savez déjà utiliser les fonctions arithmétiques ([+], [−], [×], [÷]) avec les registres de la pile et les registres mémoire. Vous savez également transférer des nombres entre les registres du calculateur et introduire ainsi qu'afficher des données en notation fixe et en notation scientifique. Pour en terminer avec les calculs manuels, sont décrites ci-après les fonctions non arithmétiques: sinus, logarithme, racine carrée...

Touches décrites dans ce chapitre:

ABS	→D.MS	INT	n!	R→P	$1/x$
COS	D.MS+	LN	→OCT	SIN	$\sqrt{x}$
DEG	GRD	LOG	RAD	TAN	y^x

Il se peut que vous n'utilisiez pas directement toutes ces fonctions; cependant, il est presque certain que vous les utiliserez indirectement lorsque vous emploierez les cartes préprogrammées de la bibliothèque «Programmes types» ou d'autres bibliothèques. Ainsi, vous avez déjà utilisé la fonction exponentielle ([y^x]) et la fonction [INT] dans le programme de placement présenté en introduction. Sans la fonction exponentielle [y^x], le programme aurait dû mettre en œuvre des méthodes arithmétiques plus complexes.

Ces fonctions sont à la fois faciles à assimiler et à utiliser. Dans l'introduction, nous avons dit que pour utiliser une fonction, il fallait d'abord appuyer sur une touche de préfixation ([f], [f^{-1}] ou [g]), puis sur la touche de la fonction désirée: la touche [g] permet d'exécuter une fonction inscrite en bleu, la touche [f] une fonction inscrite en jaune et la touche [f^{-1}] l'inverse ou le complément de cette fonction.

Jusqu'ici nous n'avons pas encore décrit toutes les fonctions disponibles ainsi que les règles qui en régissent leur utilisation. Ces fonctions sont regroupées sur les tableaux 3-1, 3-2 et 3-3. Pour exécuter n'importe quelle fonction, ces tableaux donnent les conditions relatives à la (ou aux) valeur d'entrée, les touches à utiliser et les conditions applicables aux résultats. Si vous voulez effectuer un calcul immédiatement, reportez-vous aux tableaux 3-1 à 3-3.

FONCTIONS ANGULAIRES

Les opérations, énumérées dans le tableau 3-1, comprennent les fonctions trigonométriques (sinus, cosinus, tangente et leurs inverses), les conversions de coordonnées rectangulaires/polaires, l'addition et la soustraction d'angles exprimés en degrés, minutes, secondes et la conversion d'angles décimaux en angles sexagésimaux (degrés, minutes, secondes) ou inversement.

MODE ANGULAIRE

Le HP-65 calcule les angles en degrés décimaux, radians ou grades. A sa mise sous tension, le HP-65 calcule automatiquement en degrés décimaux.

Fonctions mode angulaire

Touches	Fonction
g GRD	Sélection mode grades
g RAD	Sélection mode radians
g DEG	Sélection mode degrés

400 grades = 360 degrés = 2π radians

Touches auxquelles s'applique le mode angulaire
SIN, COS, TAN, R→P, →DMS

Dans les exemples, le HP-65 travaille en mode degrés.

Touches	Fonction	Valeur(s) d'entrée	Résultat(s)
f COS	Cosinus	Angle[1]	Cosinus (**x**) en **X**
f^{-1} COS	Arc cosinus	**x** non supérieur à 1 ni inférieur à −1 ($\|x\| \leq 1$)	Valeur principale d'arc cosinus (**x**) en **X** ($0° \leq$ résultat $\leq 180°$)[2]
f SIN	Sinus	Angle[1]	Sinus (**x**) en **X**
f^{-1} SIN	Arc sinus	**x** non supérieur à 1 ni inférieur à −1 ($\|x\| \leq 1$)	Valeur principale d'arc sinus (**x**) en **X** ($-90° \leq$ résultat $\leq 90°$)[2]
f TAN	Tangente	Angle[1]	Tangente (**x**) en **X**
f^{-1} TAN	Arc tangente	**x** (sans limites)	Valeur principale d'arc tangente (**x**) en **X** ($-90° \leq$ résultat $\leq 90°$)[2]
f R→P	Conversion de coordonnées rectangulaires (**x**, **y**) en coordonnées polaires (r, θ)	**x**, **y** dans **X**, **Y**	r, θ^1 dans **X**, **Y**
f^{-1} R→P	Conversion de coordonnées polaires (r, θ) en coordonnées rectangulaires (**x**, **y**)	r, θ^1 dans **X**, **Y** dépassement inférieur de capacité : $\theta = \pm 90°$ (mod. 180°) et $r < 1$)	**x**, **y** dans **X**, **Y**
f →D.MS	Conversion d'angle décimal en format DDDDD.MMSS[3]	Angle décimal[4]	DDDDD.MMSS en **X**
→D.MS	Conversion d'angle sous forme DDDDD.MMSS[3] en format décimal	DDDDD.MMSS	Angle décimal en **X**[4]
f D.MS+	Addition (**x** + **y**) en format DDDDD.MMSS[3]	**y** } DDDDD.MMSS[4] **x** }	DDDDD.MMSS en **X** (somme)[4]
f^{-1} D.MS+	Soustraction (**x** − **y**) en format DDDDD.MMSS[3]	**y** } DDDDD.MMSS[4] **x** }	DDDDD.MMSS en **X** (différence)[4]

[1] Angle décimal dans le mode angulaire existant
[2] Ou équivalent en grades ou radians
[3] DDDDD.MMSS : D = degrés, MM = minutes, SS = secondes
[4] L'angle ne doit pas dépasser 99999.99999 degrés décimaux (ou l'équivalent en radians ou grades) ou 99999.59599 sous la forme DDDDD.MMSS pour le mode angulaire radians ou grades

Tableau 3-1. Fonctions angulaires.

CONVERSION DE DEGRÉS DÉCIMAUX EN DEGRÉS SEXAGÉSIMAUX

Un angle exprimé sous la forme décimale peut être converti sous la forme sexagésimale en degrés, minutes, secondes ou inversement. Pour les degrés, minutes, secondes, le format est DDDDD.MMSS. Pour afficher ce format, utiliser les touches DSP • 4. Cette fonction dépend du choix du mode, ainsi qu'il est indiqué ci-après.

Exemple 1: Convertir $\pi/7$ radians en degrés, minutes, secondes.

Touches	Affichage	Commentaires
DSP • 4		Affichage à 4 décimales
g π 7 ÷	0.4488	$\pi/7$
g RAD	0.4488	Sélection mode radians
f →D.MS	25.4251	Réponse : 25° 42' 51"

Exemple 2: Opération inverse, mais avec reconversion en grades (au lieu de radians). Cette méthode permet la conversion d'un mode angulaire à l'autre.

Degrés décimaux ⇄ radians
Degrés décimaux ⇄ grades
Radians ⇄ grades

Touches	Affichage	Commentaires
g GRD	25.4251	Sélection mode grades
f-1 →D.MS	28.5713	Réponse en grades

Exemple 3: Conversion en degrés décimaux.

Touches	Affichage	Commentaires
g LST x	25.4251	25° 42' 51"
g DEG	25.4251	Sélection mode degrés
f-1 →D.MS	25.7142	Réponse en degrés décimaux
DSP • 2	25.71	Retour à 2 décimales

ADDITION OU SOUSTRACTION DE DEGRÉS, MINUTES, SECONDES (mode DDDDD.MMSS).

Exemple 1: Calcul de la somme de 45° 10' 50" et 44° 49' 10".

Touches	Affichage	Commentaires
DSP • 4	0.0000	
45.1050	45.1050	
ENTER↑	45.1050	
44.4910	44.4910	
f DMS+	90.0000	Réponse: 90° 00′ 00″

Exemple 2: Une émission musicale commence à 9 h 25′ 7″ et se termine à 9 h 39′ 47″. Quelle en est la durée?

Touches	Affichage	Commentaires
DSP • 4	0.0000	
9.3947	9.3947	Fin de l'émission
ENTER↑	9.3947	
9.2507	9.2507	Début de l'émission
f-1 DMS+	0.1440	Réponse: durée 14′ 40″
DSP • 2	0.14	Retour à 2 décimales

FONCTIONS TRIGONOMÉTRIQUES

Calcul de cosinus 60°.

Touches	Affichage	Commentaires
g DEG 60	60.	
f COS	0.50	Réponse

Calcul d'arc cosinus (–1.) exprimé en radians.

Touches	Affichage	Commentaires
g RAD 1 CHS	–1	
f-1 COS	3.14	Réponse en radians

Calcul de sinus 30°.

Touches	Affichage	Commentaires
g DEG 30	30.	
f SIN	0.50	Réponse

Calcul d'arc sinus (1.00) exprimé en radians.

Touches	Affichage	Commentaires
g RAD 1	1.	
f-1 SIN	1.57	Réponse en radians

Calcul de tangente 40°.

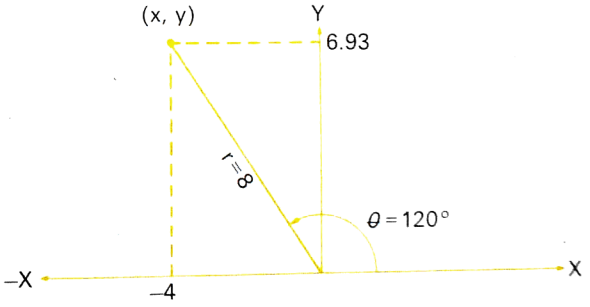
Touches	Affichage	Commentaires
g DEG 45	45.	
f TAN	1.00	Réponse

Calcul d'arc tangente (39.4) exprimé en radians.

Touches	Affichage	Commentaire
g RAD 39.4	39.4	
f^{-1} TAN	1.55	Réponse en radians

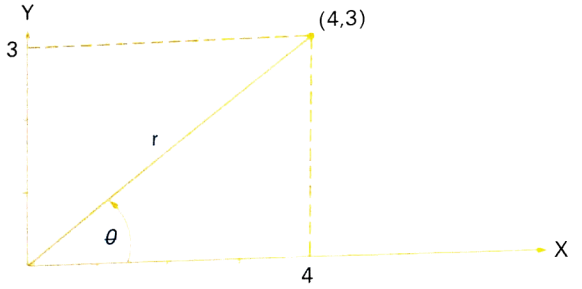
CONVERSION DE COORDONNÉES POLAIRES/ RECTANGULAIRES

Exemple: Convertir les coordonnées polaires ($r = 8$, $\theta = 120°$) en coordonnées rectangulaires.



Touches	Affichage	Commentaires
g DEG	0.00	
120 ENTER↑	120.00	θ
8	8	r
f^{-1} R→P	–4.00	Coordonnée **x**
g x⇄y	6.93	Coordonnée **y**

Exemple: Convertir les coordonnées rectangulaires ($x = 4$, $y = 3$) en coordonnées polaires, l'angle étant exprimé en degrés:



Touches	Affichage	Commentaires
g DEG 3 ENTER↑	3.00	Coordonnée **y**
4	4.	Coordonnée **x**
f R→P	5.00	r (module)
g x⇄y	36.87	θ (argument)

CONVERSIONS

Les conversions sont énumérées au tableau 3-2. Pour les conversions, la valeur d'entrée est supposée se trouver dans le registre **X**, le résultat restant dans ce registre. Les conversions d'angles sont données dans le tableau 3-1.

Exemple: Conversion d'entier à base 10 en entier à base 8. De nombreux ordinateurs utilisent la numération octale (base 8) au lieu de la numération décimale (base 10). La fonction →OCT vous permet d'exécuter facilement des conversions d'octal en décimal. Quel est l'équivalent octal du nombre décimal 512?

Touches	Affichage	Commentaires
512 f →OCT	1000.00	Représentation de 512_{10} en octal

Conversion de la valeur octale 2000 en son équivalent décimal.

2000 f-1 →OCT	1024.00	Représentation de 2000_8 en décimal

Touches	Fonction	Valeur(s) d'entrée	Résultat(s)
f →OCT	Conversion d'entier décimal en octal (base 8)	x_{10} entier décimal inférieur à 1073741824_{10}	x_8 en **X**
f-1 →OCT	Conversion d'entier octal en décimal (base 10)	x_8, entier octal*	x_{10} en **X**
f INT	Troncature à entier avec signe	± entier, fraction en **X**	± entier, 0 en **X**
f-1 INT	Troncature à fraction avec signe	± entier, fraction en **X**	± 0, fraction en **X**
g ABS	Valeur absolue	± **x**	Si **x** est négatif, – **x** en **X**, sinon aucun changement

* Lorsqu'on convertit un nombre octal en nombre décimal, il est indispensable que la valeur initiale soit effectivement une valeur octale. Un nombre octal incorrect (par exemple 998) sera ainsi interprété :

$$(9 \times 8^2) + (9 \times 8) + 8 = 656$$

998 f-1 →OCT → 656_{10}

656_{10} f →OCT → 1220_8

Tableau 3-2. Conversions

Exemple: troncatures. On peut obtenir deux troncatures réelles d'un nombre donné: partie entière et partie décimale; et une troncature apparente par modification de l'affichage.

Touches	Affichage	Commentaires
DSP • 4		
12.1980	12.1980	
f INT	12.0000	Réponse: partie entière
g LST x	12.1980	Rappel de la valeur initiale
f-1 INT	0.1980	Réponse: partie fractionnaire
DSP • 2	0.20	Retour à l'affichage à 2 décimales

Exemple: Valeur absolue. Lors de certains calculs, il est parfois intéressant de connaître l'ordre de grandeur d'un nombre. Dans le cas de calculs effectués à partir du clavier, vous pouvez afficher le nombre et en changer le signe s'il est négatif (touche CHS). Par contre, dans le cas d'utilisation d'un programme, la touche ABS vous permet de changer le signe s'il est négatif. Calcul de la valeur absolue de 3 et −3.

Touches	Affichage	Commentaires
3 g ABS	3.00	|+3|
CHS	−3.00	
g ABS	3.00	|−3|

FONCTIONS DE x ET FONCTION EXPONENTIELLE (y^x)

Ces fonctions sont énumérées sur le tableau 3-3. Pour les fonctions de **x**, la valeur d'entrée est supposée se trouver dans le registre **X**, le résultat restant dans ce registre. Par contre, pour la fonction exponentielle (y^x), la valeur de **y** est supposée se trouver dans le registre **Y**. Les valeurs contenues dans les registres **X** et **Y** ne doivent pas entraîner une opération illicite (voir tableau 1-1, page 21). Par exemple, si nous essayons de

Touches	Fonction	Valeur(s) d'entrée	Résultat(s)
f LN	Logarithme népérien (base e)	**x** non égal ou inférieur à zéro (**x** > 0)	ln (**x**) en **X**
f⁻¹ LN	Antilogarithme népérien (e^x)	**x** (sans limites)	e^x en **X**
f LOG	Logarithme décimal (base 10)	**x** non égal ou inférieur à zéro (**x** > 0)	log (**x**) en **X**
f⁻¹ LOG	Antilogarithme décimal (10^x)	**x** (sans limites)	10^x en **X**
f √x	Racine carrée (x)	**x** non négatif (**x** ≥ 0)	$\sqrt{x}$ en **X**
f⁻¹ √x	Carré (x^2)	**x** (sans limites)	x^2 en **X**
g 1/x	Inverse (1/**x**)	**x** différent de zéro (**x** ≠ 0)	1/x en **X**
g n!	Factorielle d'entier (n!) $n! = 1 \cdot 2 \cdot 3 \cdot \ldots \cdot (n-1) \cdot n$ $0! = 1$	Entier non négatif n en **x** (**x** ≥ 0; **x** entier)	n! en **X**
g y^x	Exponentielle (y^x)	**y** positif, **x** sans limites (**y** > 0)	y^x dans **X**; la pile descend

Tableau 3-3. Fonctions de **x** et fonction exponentielle (y^x)

calculer l'inverse de zéro, le clignotement de l'affichage nous signale l'erreur. Mais pourquoi ne pas essayer? Appuyez simplement sur les touches CLX g 1/x. Pour arrêter le clignotement de l'affichage, appuyer sur n'importe quelle touche de votre calculateur.

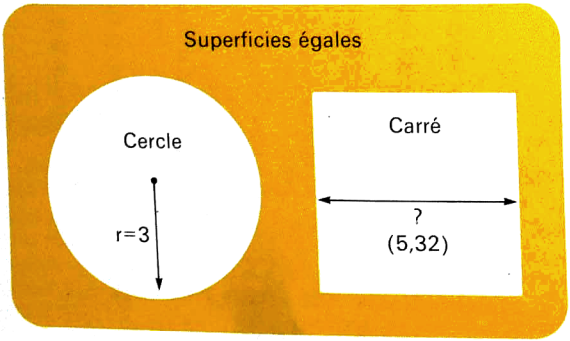
Exemple: Logarithme décimal. Calculer le gain en puissance en décibels d'un amplificateur fournissant deux fois la valeur de la puissance d'entrée.

Décibels = 10 log (2)

Touches	Affichage	Commentaires
10 ENTER↑	10.00	Mémorisation de 10
2 f LOG	.30	log 2
×	3.01	Réponse

Exemple: Antilogarithme népérien. Affichage de la constante e avec 9 décimales ($e = e^1$ = antilogarithme népérien 1).

Touches	Affichage	Commentaires
1 f-1 LN	2.72	
DSP • 9	2.718281828	Réponse
DSP • 2	2.72	Retour à l'affichage à 2 décimales



Exemple: Carré et racine carrée. Quelle est la longueur du côté d'un carré dont la superficie est égale à celle d'un cercle ayant un rayon de 3?

Superficie du cercle: $\pi \times 3^2$. La racine carrée de cette équation donne le côté d'un carré de même superficie.

Touches	Affichage	Commentaires
g [π]	3.14	π
3 f-1 [√x]	9.00	3^2
×	28.27	Superficie du cercle
f [√x]	5.32	Côté du carré

Exemple: Inverses. Calculer: 1/4 = .25.

Touches	Affichage	Commentaires
4 g [1/x]	0.25	Inverse de 4

Naturellement, vous pouvez utiliser cette valeur dans un autre calcul. Pour calculer, à la suite:

$$\frac{1}{\frac{1}{4}+\frac{1}{3}}$$

¼ étant déjà calculé, il nous suffit de procéder comme suit:

Touches	Affichage	Commentaires
	0.25	Inverse de 4
3 g [1/x]	0.33	Inverse de 3
+	0.58	Somme des inverses
g [1/x]	1.71	Réponse: inverse de la somme

Exemple: Factorielle. Six personnes doivent être photographiées ensemble. Calculez le nombre de combinaisons possibles, leurs positions changeant à chaque fois.

Touches	Affichage	Commentaires
6 g [n!]	720.00	Réponse

Exemple: Exponentielle. Au chapitre précédent, nous avons calculé les termes successifs d'une progression géométrique pour déterminer qu'après six périodes, une somme de 1000 francs

placée à 10% atteignait 1771,56 francs. En utilisant la fonction y^x, on obtiendra le même résultat en calculant

$1000\ (1.10)^6$

Touches	Affichage	Commentaires
1000 ENTER↑	1000.00	Montant initial
1.10 ENTER↑ 6		
g y^x	1.77	$(1.10)^6$
×	1771.56	Réponse

CHAPITRE 4

PROGRAMMATION

Le HP-65 est avant tout un calculateur «entièrement programmable». Nous allons nous consacrer maintenant à l'étude de sa programmation.

Vous pouvez sans doute être étonné du grand nombre de pages que nous allons consacrer à cette étude... Cela ne signifie nullement que le HP-65 est difficile à programmer.

Après avoir lu le premier paragraphe, vous pourrez déjà programmer 80% de vos problèmes! Les 20% restants font appel aux notions des paragraphes suivants. Vous n'êtes donc pas obligé de tout lire en même temps, quoique cette lecture n'ait rien de très accablant!

Nous allons tout d'abord rédiger un petit programme pour nous mettre en confiance, ce qui va nous amener à certaines questions concernant la machine elle-même, questions auxquelles nous nous efforcerons de répondre simplement.

Puis, nous affinerons notre manière de programmer et découvrirons peu à peu tout ce qui fait du HP-65 un calculateur évolué, «entièrement programmable».

Nous appliquerons ensuite nos connaissances à la résolution de deux problèmes complets à la manière d'un analyste.

Enfin et comme la machine est très astucieuse, quelques «trucs» de programmation seront indiqués à la fin de ce chapitre.

L'informatique emploie souvent des mots un peu barbares. Vous trouverez à la fin de ce chapitre un petit répertoire de ces expressions, directement orienté vers le HP-65.

Bonne lecture, et ne perdez pas le fil d'Ariane.

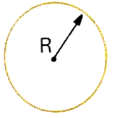
UN PROGRAMME ULTRA-SIMPLE

Dans ce paragraphe, nous allons nous attacher à la mise au point d'un programme très simple. La méthode employée pour résoudre le problème est générale; on peut donc s'y référer pour toute programmation sur le HP-65.

Problème: Calculer la surface d'un cercle dans le cas général (rayon R). Ce résultat étant acquis, calculer le volume du cône droit de même cercle de base et de hauteur donnée H.
Imaginons un instant que le HP-65 ne soit pas programmable!

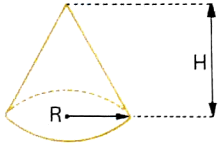
1. RÉSOLUTION DU PROBLÈME SANS PROGRAMMATION

Même si vos souvenirs de mathématiques sont un peu lointains, vous connaissez la formule donnant la surface d'un cercle de rayon R:



$$S = \pi R^2$$

Pour le volume du cône de rayon de base R et de hauteur H, le dictionnaire le plus proche nous indique:



$$V = \frac{1}{3}\pi R^2 H$$

La première étape (souvent la plus difficile) est franchie: nous avons trouvé les formules ou équations donnant les deux résultats.

Dès lors, si nous désirons résoudre le problème pour les valeurs suivantes:

R = 5
H = 15

en utilisant les touches du calculateur, nous presserons la séquence suivante:

T											
Z											
Y					25.00		78.54	78.54		1178.10	
X	5.	5.00	25.00	25.00	3.14	78.54	1.	15.	1178.10	3.	392.70
Touche	5	f^{-1}	$\sqrt{x}$	g	π	×	1	5	×	3	÷
		D_1			S_1			D_2		S_2	

Dans cette séquence nous pouvons séparer deux types de touches:

les touches permettant d'introduire les données

- D_1 pour le rayon R
- D_2 pour la hauteur H

les touches permettant les calculs, une fois les données introduites

- S_1 permet le calcul de la surface S
- S_2 permet le calcul du volume V.

Les touches de données sont essentiellement variables car dépendant de la valeur des données introduites.
Par contre, les touches de calcul forment deux séquences absolument identiques dans tous les cas de données.

L'idée qui vient immédiatement à l'esprit est la suivante:

«Et si le calculateur était capable de mémoriser les deux séquences identiques de mon problème?»

Il l'est, c'est le HP-65: programmable.

Le passage du calcul au clavier au calcul automatique par programme est extrêmement simple.

2. PASSAGE À UN PROGRAMME

Le HP-65 possède une mémoire très spéciale, la mémoire-programme, permettant, comme son nom l'indique, de stocker un programme en vue de son exécution.

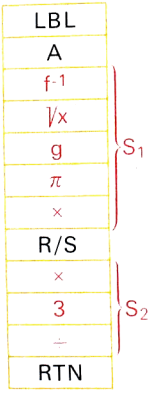
Dès lors, deux questions se posent:

Le programme (séquences identiques uniquement) étant rangé dans la mémoire, comment y accéder?

Le programme étant lancé, l'exécution est séquentielle. Pourquoi s'arrêterait-elle?

La réponse est immédiate quand on examine le programme équivalent à l'exécution manuelle du paragraphe précédent:

Quatre touches nouvelles distinguent ce programme des séquences S_1 et S_2 réunies.



LBL **A** constitue une «adresse» qui permet à la machine de repérer le début des calculs effectifs.

NB: On aurait pu utiliser les suffixes **B**, **C**, **D** ou **E** pour identifier notre «label».
La première question est résolue.

R/S constitue une commande «d'arrêt non définitif». Lorsque l'exécution atteint ce «pas» de programme, elle est suspendue et le contenu du registre **X** est affiché. L'exécution peut reprendre à partir de ce pas une fois la nouvelle donnée introduite.

RTN constitue un arrêt définitif pour la fonction placée après l'adresse **LBL** **A**. Le contenu de **X** sera affiché lorsque l'exécution aura repris et atteint le pas **RTN**.

On peut ainsi programmer 5 fonctions à définir (pour le moment!) en utilisant les labels suivants:

LBL **A** **LBL** **B** **LBL** **C** **LBL** **D** **LBL** **E**

Tout arrêt non définitif sera programmé par un pas R/S placé à l'intérieur d'une fonction.
L'arrêt le plus logique d'une fonction commençant par un LBL alphabétique est le pas RTN.
Les LBL peuvent ne pas être programmés dans l'ordre A..E.

Les fonctions se programment les unes après les autres:
(LBL C ... RTN LBL A ... RTN ...)
fonction C fonction A

Revenons à notre problème. Nous avons écrit le programme mais nous ne voyons encore pas bien comment il peut fonctionner. Le plus simple est de l'essayer!

3. MISE EN MÉMOIRE DU PROGRAMME (mode W/PRGM)

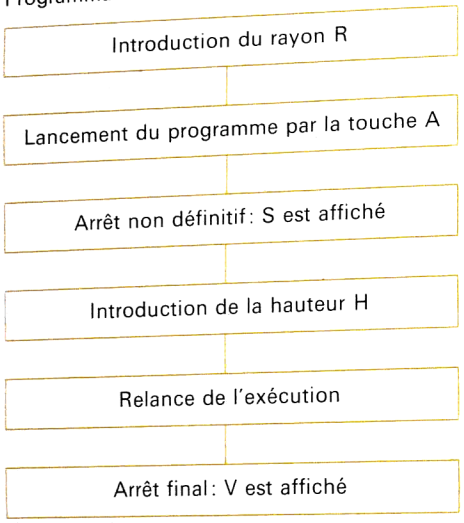
Lorsque vous introduisez un nouveau programme dans la mémoire du calculateur, à partir du clavier, vous devez toujours procéder comme suit:

1. Passage au mode programme: W/PRGM ▮ RUN.
2. Mise à zéro de la mémoire-programme:
   presser f PRGM. 00 00 doit s'afficher.
3. Introduire le programme au clavier, du premier pas jusqu'au dernier. En cas d'erreur, pour le moment, recommencer en 2.
4. Repasser en mode «exécution»: W/PRGM ▮ RUN.

Notre programme est maintenant rangé dans la mémoire de la machine, prêt à être essayé ou «testé».

4. ESSAI DU PROGRAMME (mode RUN)

Si nous considérons la feuille de papier sur laquelle est porté notre programme et si nous la comparons avec la séquence exécutée précédemment sans recours à la programmation, nous voyons très simplement comment il va falloir commander l'exécution du programme.



Exemple: R = 5, H = 15

Presser 5 **A**	**78.54** s'affiche. C'est S.
Presser 15 **R/S**	**392.70** notre volume!

L'intérêt d'un programme serait absolument nul si on en restait là. Un programme est uniquement fait pour pouvoir exécuter plusieurs calculs avec des données différentes.

Exemple: R = 10, H = 10

Presser 10 **A**	donne S = **314.16**
10 **R/S**	donne V = **1047.20**

Voilà qui devient intéressant! Nous n'avons plus à répéter les calculs identiques qui sont pris en charge par notre programme.

La seule pression de **A** remplace la séquence S_1 de cinq touches, alors que **R/S** commande l'exécution de la séquence S_2 de trois touches.

NB : R/S en anglais «Run and Stop» = «Exécuter et s'arrêter». S'arrêter quand? Sur le prochain R/S du programme (arrêt non définitif) ou sur un arrêt définitif RTN.
NB : RTN représente un arrêt définitif si l'exécution de la fonction a bien été appelée par la touche A.

Le programme étant au point, nous avons l'intention de le garder en réserve. Le HP-65 nous le permet grâce à ses cartes magnétiques.

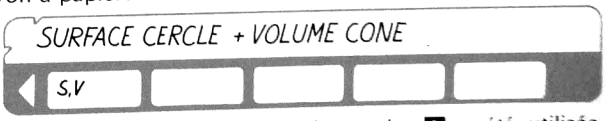
5. STOCKAGE DU PROGRAMME SUR CARTE MAGNÉTIQUE

Dans tous les cas, pour stocker un programme sur carte magnétique, on se conformera aux instructions suivantes:

1. Choisir une carte non protégée (on dit encore non «sécurisée»). Ce sont celles dont le coin entaillé n'a pas été découpé.
2. Passer en mode W/PRGM ▮▮▮ RUN (dit aussi mode «écriture»).
3. Introduire doucement la carte dans la fente inférieure droite, face imprimée visible, jusqu'à ce que le moteur l'entraîne effectivement.
4. La carte ressort de l'autre côté et la machine affiche 00 00
5. Repasser en mode W/PRGM ▮▮▮ RUN (dit aussi mode «lecture»).

Le programme est maintenant stocké sur la carte magnétique (il est toujours présent en mémoire-programme!).

Il nous reste à identifier la carte ainsi enregistrée grâce à un crayon à papier:



Nous indiquons ainsi que seule la touche A a été utilisée lors de la programmation.

Pour le moment, nous avons ce programme présent à l'esprit mais le but principal de la carte magnétique est de pouvoir l'exécuter à nouveau dans plusieurs semaines, voire plusieurs mois! D'ici là, nous aurons écrit bien d'autres programmes et le mode d'emploi de celui-ci aura été oublié. Nous allons donc rédiger une «Feuille d'instruction». Pour cela nous pouvons utiliser le cahier fourni avec chaque bibliothèque de programmes.

STEP	INSTRUCTIONS	INPUT DATA/UNITS	KEYS	OUTPUT DATA/UNITS
1.	Intr. la carte	R	A	S
2.	Intr. rayon	H	R/S	V
3.	Intr. hauteur			
	Pour un nouveau cas aller en 2.			

Revenons un instant sur ce petit programme.

C'est le premier et il est assez imparfait si l'on considère plusieurs détails importants:

1. Les données R et H ne sont pas «indépendantes».
2. On ne peut pas directement calculer le volume V quand on donne une surface S.

AMÉLIORATIONS POSSIBLES:

1. DONNÉES INDÉPENDANTES

L'idée est la suivante: Faire stocker par le programme les données R et H dans deux fonctions indépendantes, une troisième permettant le calcul de S et V.

Ainsi on pourra modifier une donnée sans modifier l'autre et exécuter quand même le programme

LBL
A
STO 1
RTN

LBL
B
STO 2
RTN

LBL
C
RCL 1
f^{-1}
$\sqrt[y]{x}$
g
π
x
R/S
RCL 2
x
3
÷
RTN

La fonction **A** stocke R dans le registre r_1, tandis que **B** stocke H dans le registre r_2, **C** se contentant de rappeler les contenus de ces registres pour exécuter le programme.

Exécution: 1. Introduire R, presser **A**
2. Introduire H, presser **B**
3. Exécuter la fonction: **C** affiche S puis **R/S** affiche V.

Avantage évident: pour changer R sans changer H, il suffit d'aller en 1, puis de sauter directement en 3. Le contenu du registre r_1 seul aura été modifié. De même pour modifier H sans modifier R.

Ce qui vient d'être montré est général: il faut toujours essayer de rendre les données indépendantes pour faciliter l'exécution des programmes (souplesse d'utilisation).

2. RÉSULTATS INDÉPENDANTS

De la même manière, essayons de rendre les deux résultats indépendants. Il suffit pour cela de programmer le calcul de V dans une nouvelle fonction **D** en ayant pris le soin de stocker le résultat S dans un registre.

Le programme devient très séduisant:

LBL
A
STO 1
RTN

LBL
B
STO 2
RTN

LBL
C
RCL 1
f^{-1}
$\sqrt{x}$
g
π
x
STO 3
RTN

LBL
D
RCL 3
RCL 2
x
3
$\div$
RTN

Il est plus long que le programme initial, mais données et résultats sont indépendants.

Exécution: 1. Introduire R presser **A**
2. Introduire H presser **B**
3. Pour calcul S presser **C**
4. Pour calculer V presser **D**

Il n'est plus nécessaire que S soit affiché pour calculer V. Toutefois, dans cet exemple, il est nécessaire de calculer au moins une fois S avant de calculer V. On peut ainsi effectuer des calculs intermédiaires au clavier puis rappeler la valeur de V en appuyant sur la touche **D** directement.

Bien! Nous commençons à connaître un peu mieux le HP-65 et nous désirons maintenant comprendre un peu mieux son fonctionnement.

LA MÉMOIRE-PROGRAMME DU HP-65

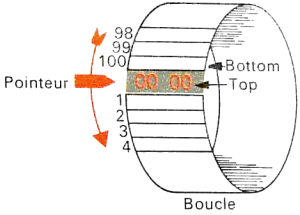
Qu'est-ce donc que cette partie du calculateur qui le distingue fondamentalement de ses cousins HP-35, 45 ou 80?

L'explication qui suit n'a aucun rapport avec la technique adoptée pour réaliser la mémoire. Elle sert uniquement à illustrer son fonctionnement.

MODÈLE DE FONCTIONNEMENT DE LA MÉMOIRE-PROGRAMME

Ce modèle utilise deux éléments constitutifs:

Le premier sera considéré comme statique: la BOUCLE.
Le second sera considéré comme mobile: le POINTEUR.



- La boucle est représentée comme un support pouvant contenir jusqu'à 100 pas de programme.

Un pas est constitué de une ou deux touches (dans ce cas-là, on parle de pas «combiné»).

Deux pas portent un nom spécial permettant de les distinguer des autres:

Le début (TOP en anglais) affiché par le code 00 00 qui est le seul pas non utilisable par l'utilisateur.
Le dernier pas (BOTTOM en anglais), centième pas, dernier à pouvoir être inscrit dans la mémoire.

TOP et BOTTOM se rejoignent pour boucler la boucle.

- Le pointeur est l'élément mobile du modèle.

Il permet d'exécuter le programme en se déplaçant de pas en pas dans le sens:

début → fin
pas 1 → pas 100

et ceci quel que soit le programme. (Nous verrons plus loin que cela n'empêche pas les branchements conditionnels et que l'exécution n'est pas obligatoirement séquentielle.)

Le pointeur est toujours matérialisé:

- en mode W/PRGM: sa position dans la mémoire est affichée par le code du pas devant lequel il est placé;
- en mode RUN: sa position ne cesse de changer quand le programme se déroule. Lors des arrêts, il affiche le contenu du registre **X**.

Tout saut de pas demandés dans un programme ne modifie pas le sens de rotation du pointeur.

Même si l'adresse du saut à effectuer se trouve très proche de sa position actuelle mais du côté du début, le pointeur rejoindra cette adresse en passant par la fin effectuant parfois un tour presque complet de la boucle (ce qui n'est pas plus long d'ailleurs!).

CODAGE DES TOUCHES

L'affichage est seulement numérique et il est pourtant nécessaire de connaître le contenu de la mémoire-programme pour éventuellement le corriger ou l'effacer en partie, ou simplement pour vérifier les pas d'un programme.

Le codage de chaque touche est composé de deux chiffres; chaque pas pouvant comporter de une à deux touches, le codage d'un pas sera deux ou quatre chiffres;

Le code d'une touche quelconque n'est pas à apprendre par cœur car il peut facilement se retrouver, à la simple vue de la machine. Pour cela, il faut suivre deux règles:

Règles:

1) Une touche de chiffre est codée comme le chiffre.
2) Toute autre touche est codée ligne-colonne.

Colonnes 3

Lignes 2

NB : C'est la touche qui est codée et non la fonction de la touche :
Ex. : [COS], [y^x] et 5 ont le même code 05.

La machine reconnaît le cos du [y^x] grâce au code du préfixe :

cosinus = [f] (codé 31) + [COS] (codé 05)

Codes combinés :
Les fonctions les plus couramment utilisées sur le HP-65 qui nécessitent la pression de deux touches ne forment qu'un pas de programme : les codes des deux touches sont combinés et quatre chiffres sont affichés :

- Manipulation de nombres dans la pile :

g [x≷y]	35 07
g [R↓]	35 08
g [R↑]	35 09

- Tests de comparaison :

g [x≠y]	35 21
g [x≤y]	35 22

g x=y — 35 23
g x>y — 35 24

- Stockage et rappel d'un nombre (sauf pour R_9):

STO 2 — 33 02
RCL 7 — 34 07

- Touches spéciales:

g LST x — 35 00
g NOP — 35 01

MODE DE REMPLISSAGE DE LA MÉMOIRE-PROGRAMME

Nous connaissons un peu mieux ce qu'est la mémoire mais ne savons encore rien de la manière dont elle se remplit, se vide, etc.

Revenons à notre exemple: calcul de la surface d'un cercle, et voyons comment est insérée dans la mémoire la séquence constituant le programme:

LBL
C
f^{-1}
$\sqrt{x}$
g
π
×
RTN

Lorsque nous avons introduit ce programme, notre première précaution a été de «vider» le contenu de la mémoire. Aussi bizarre que cela puisse paraître, lorsque l'on met la machine ON, la mémoire n'est pas vide! Nous reviendrons là-dessus par la suite.

Donc, nous avons pressé:

f PRGM

et 00 00 est apparu sur l'écran.

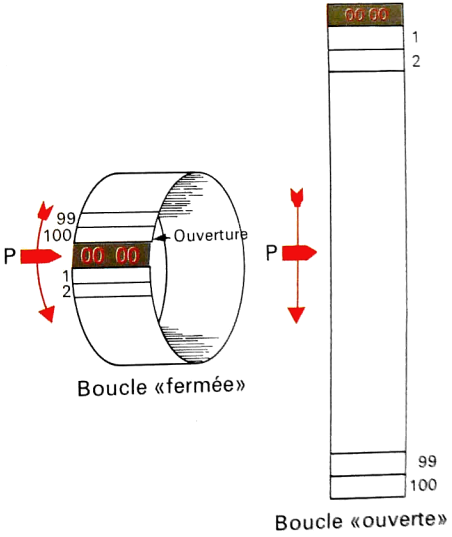
f PRGM

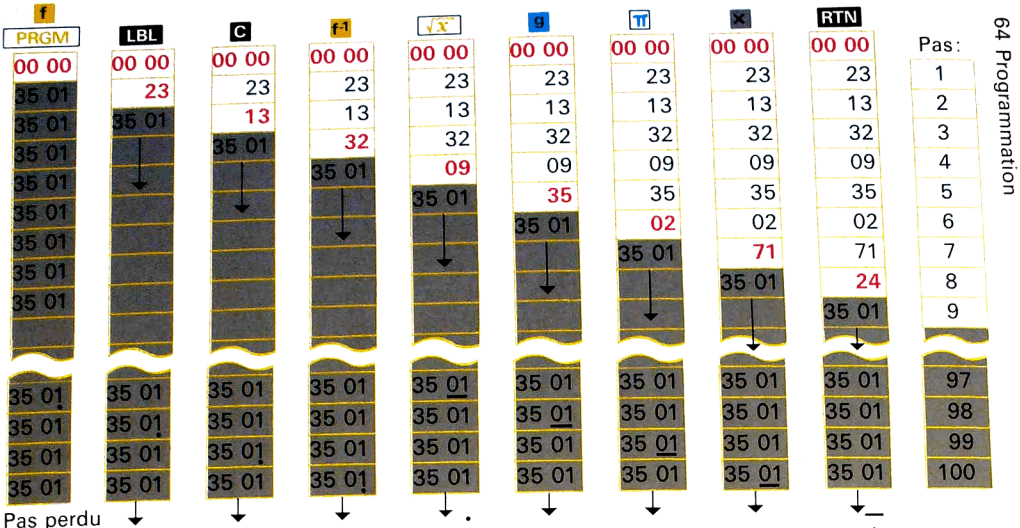
La pression de ces deux touches remplace les 100 pas de la mémoire par 100 pas non opératoires (code : **35 01**) correspondant aux touches :

g NOP

Dans un programme, un tel pas fait avancer le pointeur d'une position sans provoquer aucune opération.

Pour mieux comprendre la suite des opérations, nous allons remplacer la boucle fermée par une boucle ouverte au niveau du TOP.





En rouge, la position du pointeur lors de l'insertion des pas correspondant aux touches.
Lorsqu'on presse une touche formant un pas de programme, la mémoire s'ouvre au niveau du pas suivant le pas affiché avant l'introduction, le pointeur avance d'une position, affichant le nouveau pas introduit. Le reste de la mémoire glisse d'une position vers la fin, le 100e pas étant définitivement perdu.

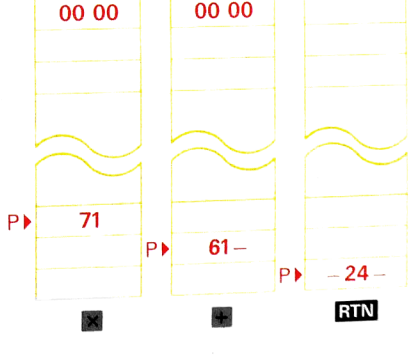
Cas spéciaux:

- **99e pas**: Lorsque l'on presse une ou deux touches formant le 99e pas dans la mémoire, un signe moins s'affiche avec le code de ce pas.
- **100e pas**: Lorsque le 100e pas est défini, le calculateur l'affiche sous forme codée et entre deux signes «–».
- **Pas supérieurs à 100**: Ils sont affichés entre deux signes «–» mais non mémorisés.

Destruction du 100e pas:
Lorsqu'un programme comporte 100 pas déjà inscrits dans la mémoire-programme et que le pointeur n'est pas au centième, toute insertion de pas supprime le 100e, le remplaçant par le 99e, etc.

Heureusement, on peut savoir quand le programme comporte 100 pas, même si l'on n'est pas au 99e ou au 100e. En effet, le calculateur affiche un signe «–» avec chacun des 100 pas dans ce cas.

Exemple: Nous avons rempli 97 pas et il nous reste seulement trois pas pour inroduire × + RTN (programme de 100 pas).



AFFICHAGE DE LA MÉMOIRE-PROGRAMME PAS PAR PAS

En mode W/PRGM, on peut afficher facilement chaque pas de la mémoire-programme grâce à la touche **SST** ce qui permet de vérifier l'exactitude des pas, d'en ajouter ou supprimer, etc. Cette touche n'est pas programmable, sinon le code 25 devrait s'inscrire à chacune de ses pressions !

Faisons une petite expérience :
- Mettons la machine OFF puis ON
- Passons en mode W/PRGM : 00 00, le code du TOP est affiché.
- Pressons maintenant **SST** un certain nombre de fois.

Pression n°	Code affiché	Touches correspondantes	
1	23	LBL	1/x
2	11	A	
3	35	g	
4	04	$\frac{1}{x}$	
5	24	RTN	
6	23	LBL	$\sqrt{x}$
7	12	B	
8	31	f	
9	09	$\sqrt{x}$	
10	24	RTN	

Euréka ! A la mise sous tension de la machine, les cinq touches de la rangée supérieure **A**, **B**, ..., **E** sont définies comme de petits programmes calculant 1/x, $\sqrt{x}$ ... $x \rightleftarrows y$. C'est la raison pour laquelle il faut «vider» la mémoire avant de créer son propre programme.

EFFACEMENT D'UN PAS DE PROGRAMME

La pression de **g** DEL supprime de la mémoire un pas affiché, remontant la partie inférieure d'une position vers le début de

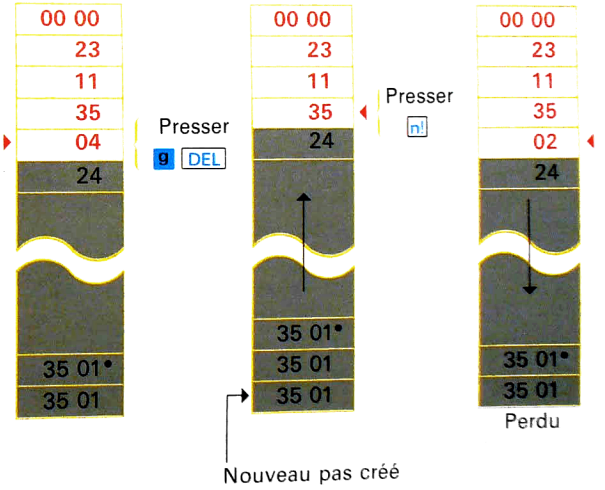
la mémoire. Ainsi un pas g NOP (35 01) est créé comme 100e pas.

Exemple:

– Mettons la machine OFF puis ON

– Passons en mode W/PRGM (00 00 affiché)

– Pressons SST jusqu'à ce que le pas 04 représentant le 1/x du programme par défaut A, soit affiché. (Il faut presser SST quatre fois.)



La pression de g DEL lorsque 04 était affiché a effacé ce pas de la mémoire, remontant tous les suivants d'une position et créant ainsi un pas 35 01 (g NOP) comme 100e pas. Le pointeur est remonté au pas précédent; ainsi il suffit de presser

le 02 ([n!]) pour changer la fonction, qui calculera ainsi **x!** à la place de [1/x].

Pour vérifier ce changement, exécuter la fonction RUN.

Presser	Affichage	Commentaires
5 A	120.00	5! = 120
3 A	6.00	3! = 6

NB:

1. Pour effacer une séquence complète de pas, il est plus facile de déplacer le pointeur jusqu'au dernier pas à effacer, puis de presser g [DEL] autant de fois qu'il y a de pas à effacer.
   A chaque pression, le pas affiché s'efface et le pointeur affiche le pas précédent.

2. Cas particulier: **Mémoire pleine** (100 pas)
   Lorsque la mémoire a été complètement remplie et que l'on affiche son contenu pas à pas à l'aide de SST, un signe – apparaît avec chaque code.

   - Si l'on essaie d'effacer un pas quelconque entre le 1er et le 99e inclus, le pas n° 100 est également effacé.
   - Si l'on essaie d'effacer le 100e pas d'une mémoire pleine, les pas 99 et 100 sont effacés.

NB: SST en mode RUN exécute un programme pas à pas, ce qui permet une mise au point rapide d'un programme un peu long.

BRANCHEMENT INCONDITIONNEL

On a vu précédemment que l'ordre normal d'exécution d'un programme est séquentiel, le pas situé en 63e position sur la boucle ne pouvant être exécuté que si le pas en 62e position vient de l'être.

Dans ces conditions et si cette seule forme d'exécution n'était possible, le HP-65 ne pourrait à aucun prix être considéré comme un calculateur «entièrement» programmable. Ce vocable est réservé à des catégories de calculateurs se rapprochant un peu des ordinateurs par les fonctions logiques suivantes:

- tests, branchements conditionnels
- sous-programmes
- branchements inconditionnels.

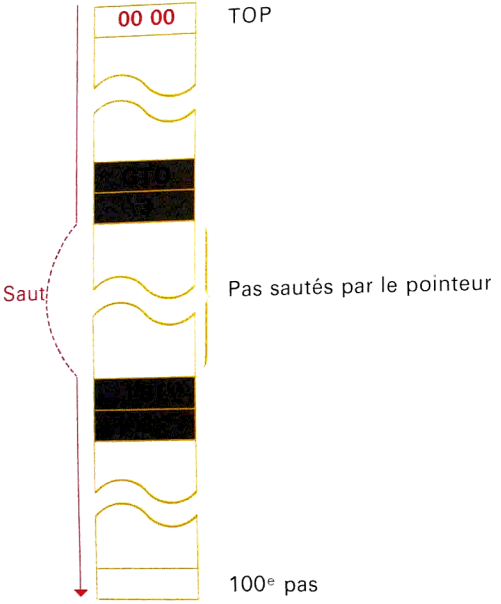
Etudions pour le moment ce dernier, qui est le plus simple.

Il est possible de placer comme pas de programme un ordre de rupture de séquence, dit encore branchement direct ou inconditionnel qui, atteint par le pointeur de programme, stoppe l'exécution de celui-ci. La commande de branchement donne toujours «l'adresse» à laquelle doit se faire le branchement direct. Le pointeur «cherche» la première adresse ainsi indiquée qu'il rencontre (sachant que son mouvement autour de la boucle a toujours lieu dans le sens unique début → fin) et reprend l'exécution dès qu'il l'a «trouvée». S'il n'y a pas d'adresse, il risque de chercher indéfiniment! Vous aurez toujours la possibilité de stopper cette recherche inutile en pressant **R/S**.

4 pas de programme sont nécessaires pour effectuer un branchement correct:

Commande	1er pas:	**GTO** (go to)
	2e pas:	au choix **0**, **1**, ... **9**, **A**, **B**, ... **E**
Adresse «cherchée»	3e pas:	**LBL** (label)
	4e pas:	correspondant au choix **0**, **1**, ... **9**, **A**, **B**, ... **E**.

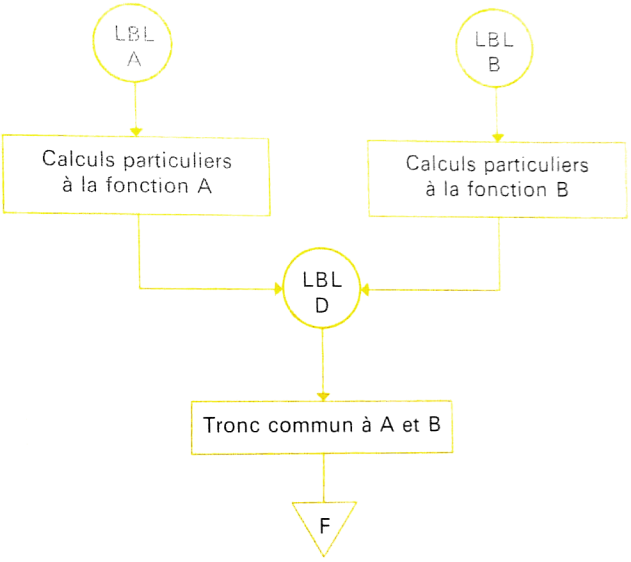
Par exemple, **GTO** **3** renvoie la suite de l'exécution après le premier **LBL** **3** rencontré dans la boucle de programme.



A quoi peut bien servir un tel branchement? Nous le verrons vraiment quand nous étudierons le branchement conditionnel. Nous pouvons déjà imaginer une utilisation très intéressante quand il s'agit de «gagner» des pas de programme.

Le branchement inconditionnel est très souvent employé lorsque deux ou plusieurs fonctions possèdent un groupe de pas commun, en particulier en fin de calcul.

Schéma fonctionnel (organigramme)



Exemple : En utilisant les fonctions **A** et **B**, écrire :

– un programme calculant

$$A = \frac{\sin x}{3 \sin^2 x + 2}$$ dans la fonction **A**

– et un programme calculant

$$B = \frac{\cos x}{3 \cos^2 x + 2}$$ dans la fonction **B**

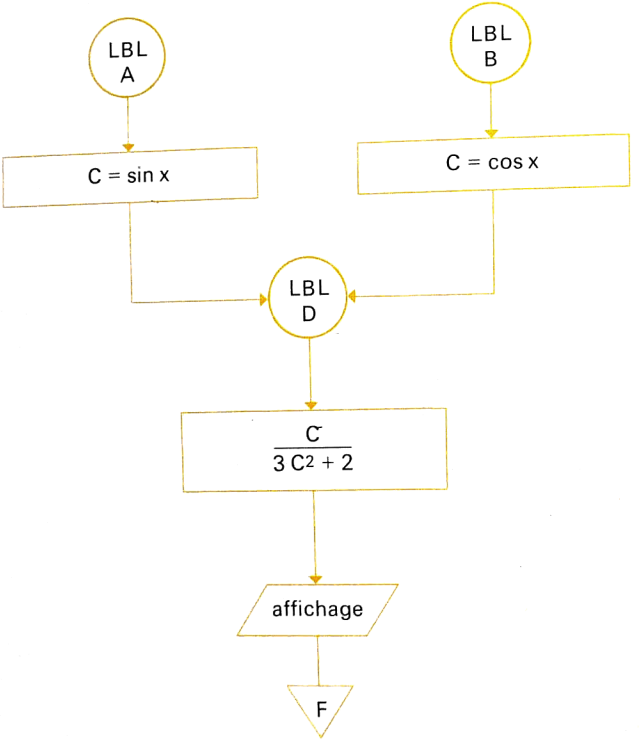
Nous avons utilisé le **LBL** **D** pour effectuer le branchement. C'est astucieux et possible dans la mesure où :

1) l'on a ainsi une fonction **D** calculant la quantité

$$b(x) = \frac{x}{3 x^2 + 2}$$ si on donne **x** par simple pression de **D**

2) l'on n'a pas besoin de **D** pour une autre fonction dans le programme.

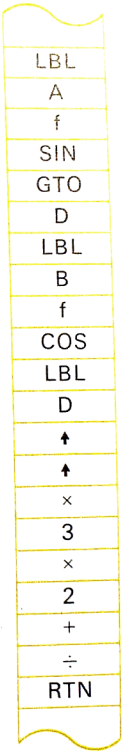
L'organigramme sera :



On peut remarquer qu'il suffit d'un seul ordre **GTO** **D** pour réaliser cet organigramme, c'est-à-dire le double branchement, en utilisant le fait que la mémoire est circulaire.

Le programme correspondant le montre bien.

PROGRAMME CORRESPONDANT



◀ Branchement à l'adresse **LBL** **D**

◀ Pas besoin de **GTO** **D** pour la fonction **B** puisque l'exécution est séquentielle

◀ La fonction **D** peut être exécutée indépendamment de **A** et **B**

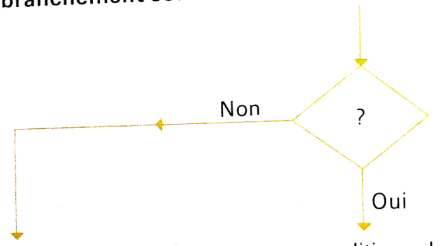
Essai du programme :

Presser	Affichage	Explication
30 **A**	0.18	$\frac{\sin 30°}{3 \sin^2 30° + 2}$

60 **B** 0.18 $\frac{\cos 60^\circ}{3\cos^2 60^\circ + 2}$ $(\sin 30^\circ = \cos 60^\circ)$

4 **D** 0.08 $\frac{4}{3.4^2 + 2}$

L'utilisation la plus importante des branchements inconditionnels consiste à incorporer un de ces branchements dans les deux pas suivant un test (voir pages suivantes) pour réaliser un **branchement conditionnel.**



Branchement conditionnel

Si la condition du test est remplie, le branchement sera effectué, différenciant deux branches dans le programme.

SOUS-PROGRAMMES

Rappel :
Jusqu'à présent, nous avons créé des fonctions **A**, **B**, ... **E** qui étaient exécutées dans un ordre séquentiel, avec ou sans branchements inconditionnels (**GTO**→ **LBL**).

Imaginons qu'un programme comporte par exemple une série de pas identiques à répéter plusieurs fois. Si nous répétons cette séquence dans le programme, la place risque de nous manquer très rapidement.

De même, si plusieurs fonctions font appel au même calcul partiel, il semble inutile de répéter ce calcul.

C'est en effet bien inutile !

Nous avons un premier moyen de nous en tirer puisque nous venons juste de voir le branchement inconditionnel.

Exemple : Nous avons vu que lorsque deux fonctions ont la même fin, on pouvait éviter de répéter cette fin.

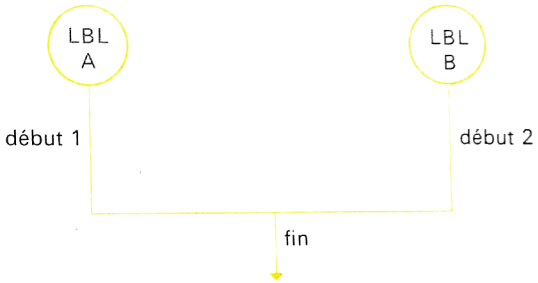


Schéma fonctionnel

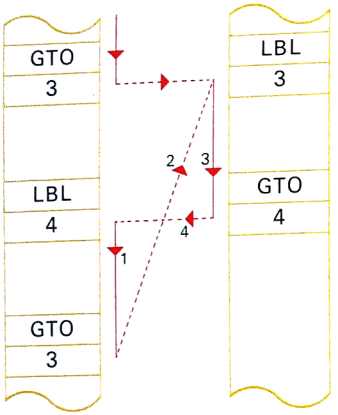
Nous voyons immédiatement la lacune de ce genre de branchement :

> Un branchement **GTO** 4 ne peut jamais rejoindre qu'une adresse **LBL** 4.

Or, si nous ne voulons pas avoir d'ennui d'exécution, nous avons intérêt à n'utiliser qu'une adresse **LBL** 4 dans la boucle de programme. Nous ne pouvons donc pas utiliser la même branche

LBL 3 ... **GTO** 4

pour rejoindre, après exécution, une autre partie du programme.



Le HP-65 nous fournit un outil beaucoup plus puissant qui est:
un niveau de sous-programme.

FONCTION

Un sous-programme est une séquence de pas, appelée directement par un programme (dit «principal» ou «appelant»). Le branchement se fait au sous-programme qui est exécuté puis un ordre de retour ramène l'exécution au pas suivant l'appel du S.P. dans le programme principal.

Schéma fonctionnel
NB: Les deux parties
{pr. principal
{sous-programme
sont situés l'un après l'autre sur la boucle (l'ordre importe peu!).

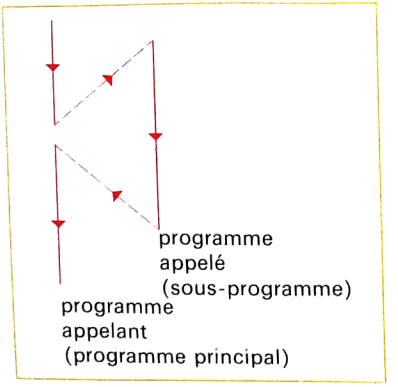
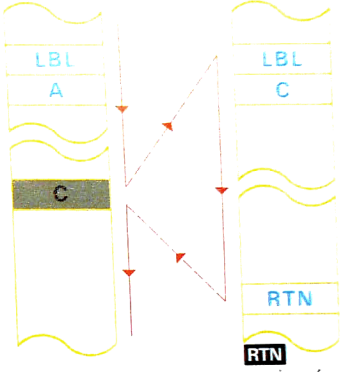


Schéma réel

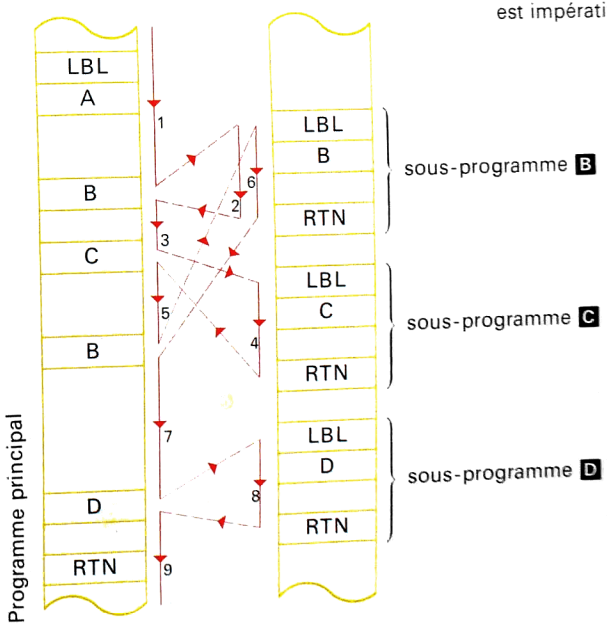
La pression de **A** (RUN) appelle le programme principal **LBL A**...

Celui-ci appelle le programme **C** (sous-programme de **A**).

RTN redonne l'exécution au programme principal **A**.



Ce qui est possible :



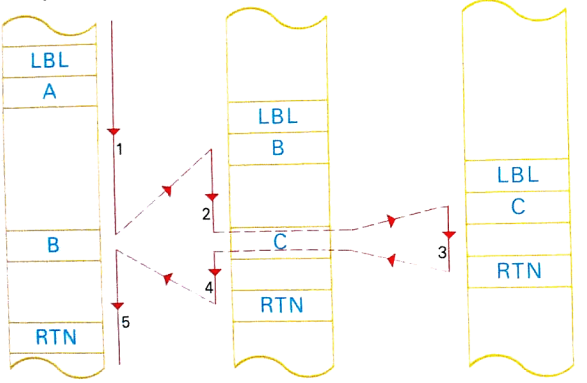
Ordre d'exécution: 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9

La configuration «série (analogie avec l'électricité) est toujours possible.

Ce qui n'est pas possible: imbrication de deux sous-programmes.

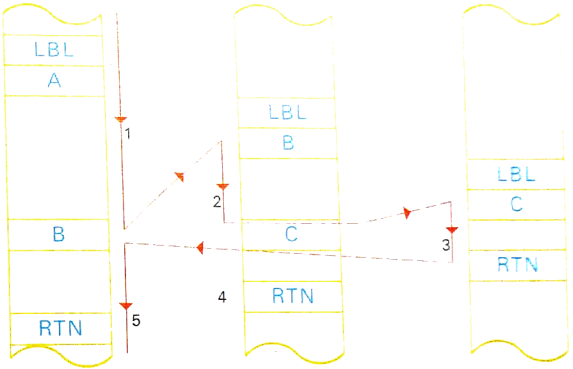
C'est la configuration «parallèle». Elle n'est pas possible car le HP-65 possède un seul niveau de sous-programme. (Il est capable de retenir seulement une adresse de retour au programme principal.)

Ce que l'on voulait faire



Ordre souhaité: 1 – 2 – 3 – 4 – 5
(**B** est sous-programme de **A**
C est sous-programme de **B**)

Ce que la machine fait :



Ordre réel : 1 – 2 – 3 – 5
4 n'est jamais exécuté
(**B** et **C** sont considérés comme SP de **A** –
C est seulement un branchement pour **B**)

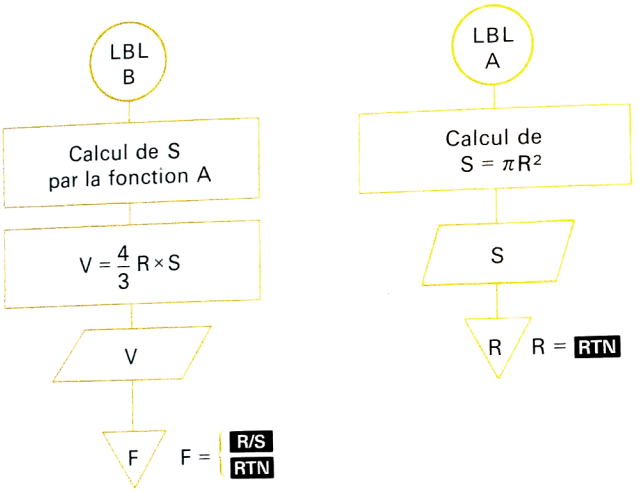
NB : Imbriquer deux sous-programmes n'est pas quelque chose d'interdit, à condition de savoir exactement ce qu'il va se passer !

Exemple : Créons deux fonctions **A** et **B** qui calculent :

A : la surface d'un cercle de rayon donné
$S = \pi R^2$
B : le volume d'une sphère de rayon donné
$V = \frac{4}{3}\pi R^3$

Nous ferons de **A** un sous-programme de **B**.

Organigramme



Si la touche **A** est pressée, la fonction est normale et le calcul s'arrête automatiquement sur le **RTN**.
Par pression de **B**, **A** est considérée comme sous-programme de **B** et le **RTN** ramène le pointeur au programme principal **B**.

NB: Il ne faut pas oublier de conserver le rayon avant d'aller exécuter le sous-programme.

Programme

LBL	
A	
f^{-1}	
$\sqrt{x}$	
g	
π	
×	
RTN	$S = \pi R^2$
LBL	
B	
↑	Ce pas conserve le rayon dans **Y**
A	appel du sous-programme
×	$S \times R$
4	
×	
3	
÷	
RTN	$V = \frac{4}{3} SR = \frac{4}{3} \pi R^3$

Essai du programme

Surface d'un cercle de rayon 10:

Presser	**Affichage**
10 **A**	314.16

Volume d'une sphère de rayon 5:

Presser	**Affichage**
5 **B**	523.60

Ça marche!

On désire créer un sous-programme car il apparaît que certains pas sont répétés une, deux ou plusieurs fois dans un programme: c'est une bonne idée à condition de savoir si le «seuil» de gain de pas est atteint.

Il ne faut pas oublier en effet qu'un sous-programme demande au minimum trois pas de plus que la séquence à répéter: **LBL**, **A** ... **E**, **RTN**.

Le «seuil» dépend bien entendu:
- du nombre de pas de la séquence à répéter,
- du nombre de fois que le sous-programme sera appelé.

DÉCISIONS LOGIQUES (ou TESTS)

Votre HP-65 permet les branchements inconditionnels, possède un niveau de sous-programme. Ceci ne suffit pas pour parler d'une machine «entièrement programmable».

Tous les ordinateurs ou les calculateurs vraiment programmables doivent être capables de décisions pendant une exécution de programme. C'est le cas du HP-65!

Mieux, il vous fournit trois types de tests:
- *Tests de comparaison*
- *Compteur programmé*
- *Tests de flags.*

Mieux encore, tous ces tests fonctionnent selon le même schéma, ce qui facilite leur utilisation.

Un test est une question figurant dans le programme. Lorsque le pointeur de programme atteint les pas posant cette question, le calculateur «réfléchit».

S'il trouve que la réponse à la question est:
- OUI, il continue l'exécution séquentielle
- NON, il saute deux pas suivant le test, puis continue l'exécution séquentielle.

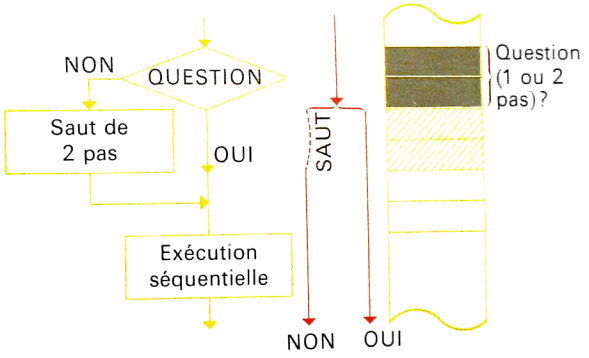
En fait, aucune confusion n'est possible si l'on réfléchit à voix haute, comme ceci:

Si «il fait beau», j'irai à pied sinon, je prendrai un bus.

TEST OUI NON

La réponse OUI est avant la réponse NON comme dans le programme!

Schéma fonctionnel des tests de la HP-65

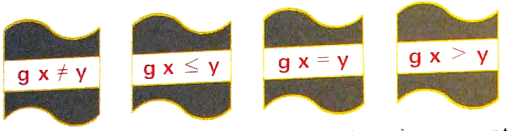


TESTS DE COMPARAISON

Dans un programme, ces tests portent sur des comparaisons entre les contenus des registres X et Y de la pile.
Quatre questions peuvent être posées:

Les touches correspondantes sont:

Les codes sont combinés pour donner un pas unique.



Le fonctionnement est identique au fonctionnement général.

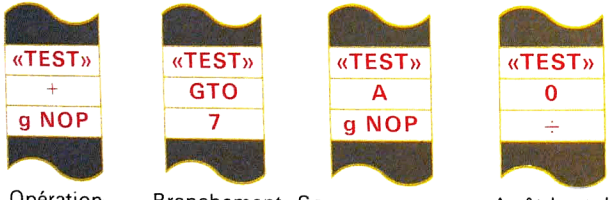
- Si la comparaison formulée dans la question correspond à la réalité (réponse OUI), l'exécution séquentielle n'est pas interrompue.
- Si, au contraire, la comparaison ne correspond pas à la réalité (réponse NON), le pointeur saute deux pas avant de continuer l'exécution.

Exemple :

branchement au prochain **LBL** [1] si **x≠y**

sinon exécution séquentielle

Principaux cas d'utilisation

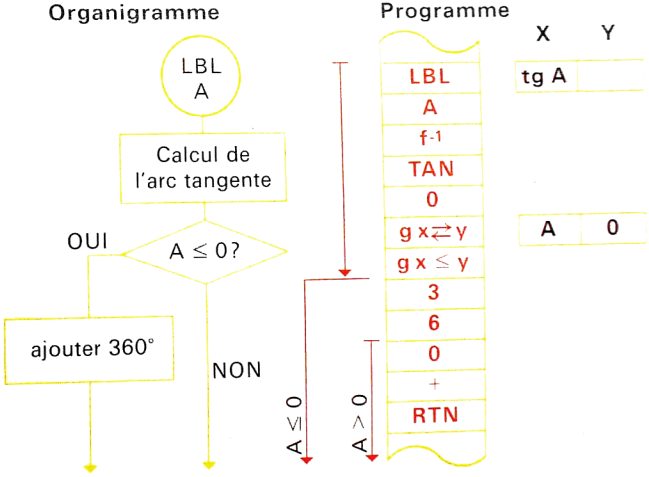


Opération — Branchement — Sous-programme — Arrêt brutal

Conditionnels

Les tests de comparaison permettent parfois des «astuces» de programmation ayant pour effet de gagner des pas, souci constant du programmeur.
L'exemple ci-dessous illustre bien une utilisation classique d'un test.

Exemple : La tangente d'un angle inconnu est donnée (registre **X**). Créer un programme qui calcule l'angle et affiche :
- l'angle calculé s'il est positif
- l'angle augmenté de 360° s'il est négatif ou nul.



Si l'angle calculé par l'arc tangente est négatif ou nul, 360 est ajouté ; s'il est positif, les deux pas composant 36 sont sautés et seul 0 est ajouté à l'angle qui demeure ainsi inchangé.

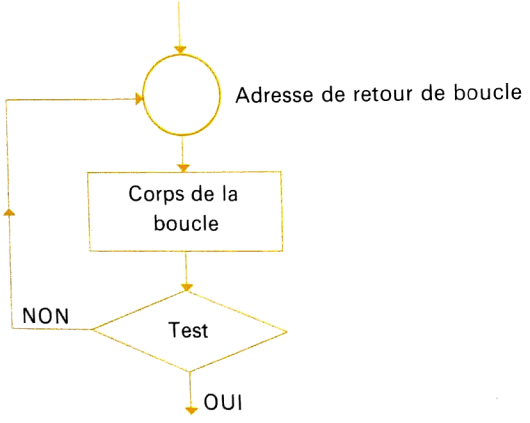
Attention !

1. Les tests de comparaison utilisent le registre R_9. Si un nombre se trouve placé dans ce registre avant un des tests, il est irrémédiablement détruit par le test. Il ne faut donc pas se servir de ce registre pour stocker des nombres si un test doit être rencontré dans la suite du calcul.

2. Ne pas oublier que le contenu de la pile ne bouge pas pendant un test. L'exécution conditionnelle qui résulte de la décision reprend la pile dans l'état où elle était lors de la question.
3. Attention à la précision dans un test. N'oubliez pas qu'un nombre est considéré comme égal à zéro par le HP-65 s'il est inférieur ou égal à 10^{-99}.
   Il est souvent préférable de comparer un résultat à 10^{-3}, 10^{-4}, etc... que de le comparer à 0.

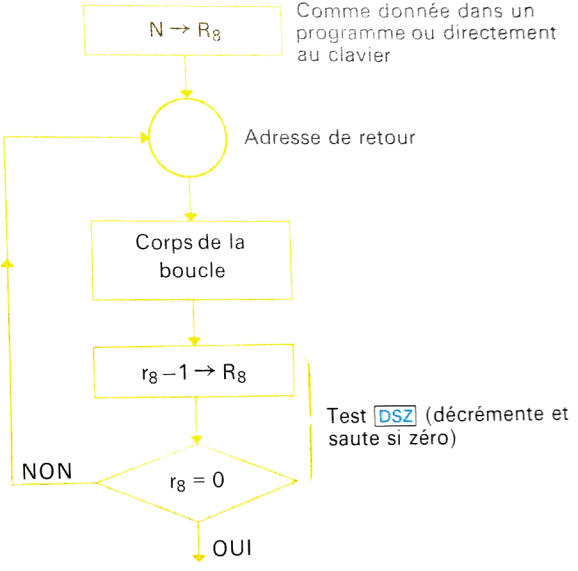
COMPTEUR PROGRAMMÉ

Il est souvent bien difficile de mettre en œuvre des boucles de programmes. Une boucle, c'est une partie de programme qui est répétée pendant l'exécution un certain nombre de fois.

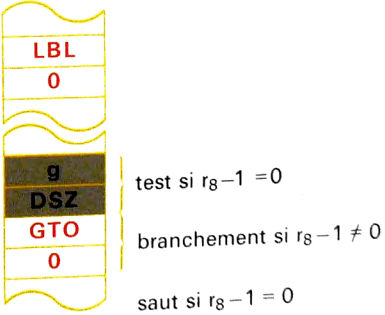


Le HP-65 est doté d'un compteur automatique de boucles. Son fonctionnement utilise la touche [DSZ].

Mise en œuvre



TYPE RÉEL DE PROGRAMME



NB: Pour que le bouclage s'arrête, il faut que zéro soit trouvé, donc qu'au départ:
– la valeur que l'on met dans R_8 soit entière
– qu'elle soit positive et inférieure à 10^{10}.

Si $r_8 < 0$ au départ, $(r_8 - 1)$ n'est jamais égal à zéro et le programme ne peut s'arrêter.

Exemple: Calcul de série convergente
La base des logarithmes naturels, le nombre bien connu e, peut être obtenu par la série:

$$e = \sum_{k=0}^{\infty} \frac{1}{k!} = \frac{1}{0!} + \frac{1}{1!} + \frac{1}{2!} + \ldots + \frac{1}{k!} + \ldots$$

Ecrivons un programme qui calcule e par cette méthode itérative. Nous arrêterons le programme au bout de N itérations, calculant:

$$e_{approché} = \sum_{k=0}^{N} \frac{1}{k!} = \frac{1}{0!} + \frac{1}{1!} + \ldots + \frac{1}{N!}$$

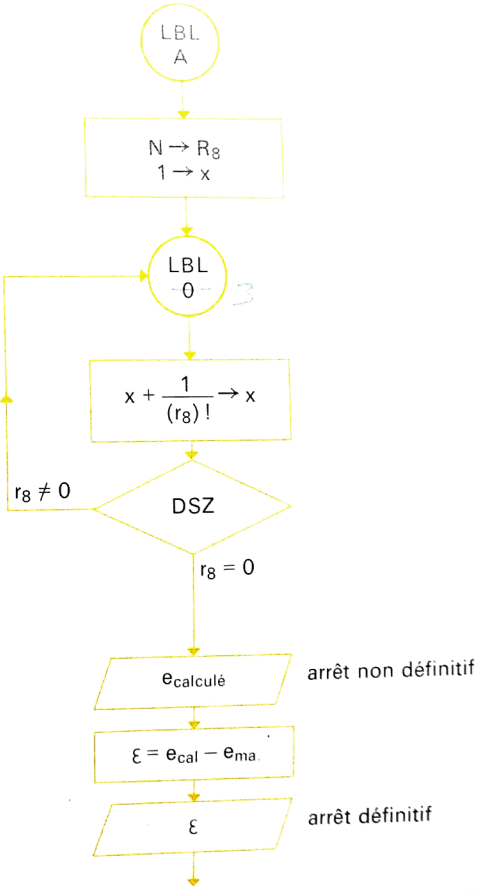
Nous verrons ainsi la convergence de cette série.

La valeur étant calculée, nous la comparerons avec celle fournie par le HP-65 grâce à la touche [LN]:

($e = e^1$ Presser: 1 [f-1] [LN])

Le résultat sera: $\varepsilon = e_{calculé} - e_{machine}$

Organigramme



NB : 1. On met 1 dans le registre **X** de la pile au départ (et non 0) car la boucle va être faite N fois seulement.

calculant: $\frac{1}{N!} + \frac{1}{(N-1)!} + \ldots + \frac{1}{1!}$

Il reste à ajouter encore: $\frac{1}{0!} = 1$. C'est ce que nous faisons au début.

2. N est supposé entier positif. Nous aurions pu le tester en début de programme.

Programme

LBL
A
STO 8
1
LBL
3
RCL 8
g
n!
g
$1/x$
+
g
DSZ
GTO
3
R/S
1
f^{-1}
LN
–
RTN

Essai du programme

N = 5 5 A → $e_{calculé}$ = 2.716666667 (DSP • 9 pour l'afficher)

R/S → ε = −0.001615161 (DSP 9 pour l'afficher)

N	5	10	15	20	25
$e_{cal.}$	2.716666667	2.718281803	2.718281830	. . .	. . .
ε	−0.001615161	−0.000000025	0.000000002	. . .	. . .

Cas principaux d'utilisation

– séries entières du type:

$$\sum_{k=1}^{N} F(k, \ldots)$$

$$\prod_{k=1}^{N} F(k, \ldots)$$

– séries alternées (en liaison avec Flag)
– chaque fois qu'un compteur est nécessaire.

NB: Si N est le résultat d'un calcul, il risque de ne pas être entier positif; il est alors prudent d'en prendre la valeur absolue et la partie entière avant de le ranger dans R_8.

FLAGS

Un Flag (en français: drapeau) est un élément interne du calculateur qui remplit une fonction logique à la manière d'une variable booléenne en mathématiques: une telle variable ne peut prendre que deux valeurs, 0 ou 1.
La machine possède deux Flags (1 et 2) indépendants et fonctionnant de la même manière.
Un Flag est caractérisé uniquement par son état (0 ou 1). Souvent, on dit pour l'état 0 que:

– le Flag est «OFF» / enlevé / absent

alors que dans l'état 1

– le Flag est { «ON» mis présent.

La suite des explications concerne le Flag 1 (F 1) et est bien entendu applicable pour le Flag 2.

Mécanisme du Flag

1. Généralités

On distingue deux étapes dans l'utilisation d'un Flag:

- Choix de l'état du Flag
  Cette étape est programmée ou non, explicitée ou non.
- Test sur l'état du Flag
  Cette étape figure seulement de manière explicite dans un programme.

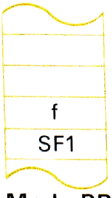
2. Fonctionnement réel

a) Pour fixer l'état du Flag, il suffit de le mettre **f** SF1 ou de l'enlever **f⁻¹** SF1.

Ceci peut se faire:
– comme commande (en mode RUN) avant de démarrer une partie de programme
– à l'intérieur d'un programme.

Touches pressées:
5 **f** SF1 **A**

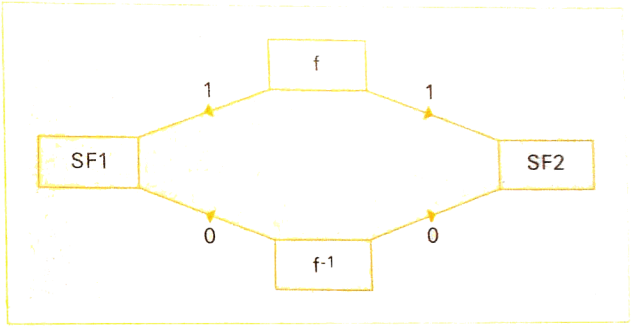
Mode RUN
Le programme A est exécuté avec 5 comme donnée et le Flag 1 étant présent.



mémoire-programme

Mode PRGM
Le Flag 1 est mis à l'intérieur d'un programme.

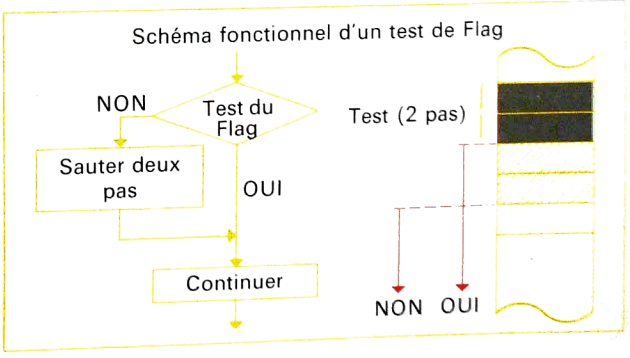
La machine possédant deux Flags, l'état d'un Flag quelconque peut être résumé par le schéma :



b) Pour tester l'état d'un Flag, le mécanisme est celui employé pour n'importe quel test sur le HP-65 :

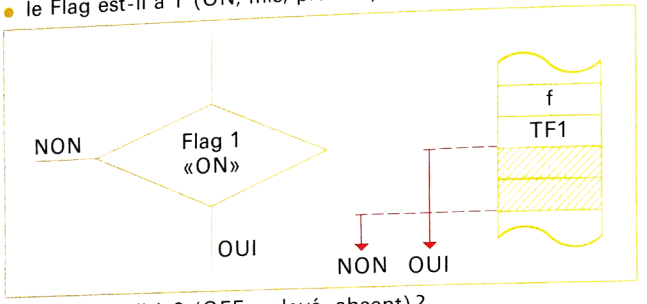
– une question est posée à la machine à l'intérieur du programme ;
– suivant la réponse, l'exécution est reprise
  - au pas suivant le test (si OUI)
  - au troisième pas suivant le test (si NON).

La machine possédant deux Flags, l'état d'un Flag quelconque peut être résumé par le schéma :

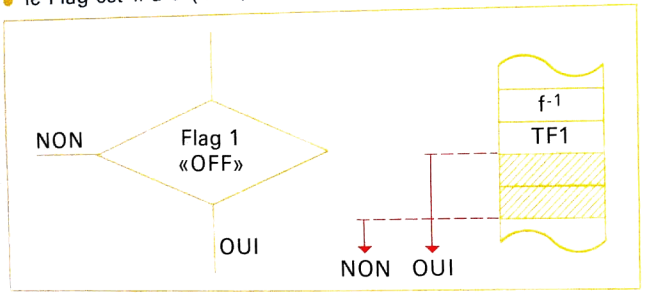


La question du test peut être :

- le Flag est-il à 1 (ON, mis, présent) ?



- le Flag est-il à 0 (OFF, enlevé, absent) ?



NB : Le fait de tester un Flag ne change pas son état. A la mise sous tension, les deux Flags sont «OFF».

Utilisation des Flags

Le Flag est un artifice extrêmement puissant qui permet en général les astuces de programmation entraînant des «gains» de pas substantiels. Quand on veut programmer une décision qui ne résulte pas d'une comparaison directe des registres X et Y de la pile, on doit penser aux Flags !

Exemple : Créer un programme dans la touche **A** qui calcule

$$\sqrt{x^2+1} \text{ ou } \sqrt{\frac{x^2+1}{x^2}}$$

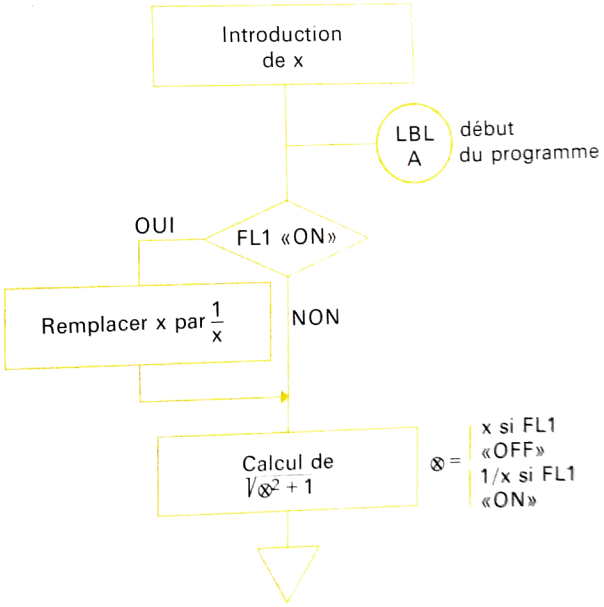
Au départ, il semble difficile d'obtenir deux résultats différents en pressant la même touche!
Pensons donc aux Flags. Le choix est

- calcul de $\sqrt{x^2+1}$
- calcul de $\sqrt{\frac{x^2+1}{x^2}}$

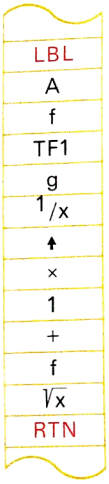
C'est un choix arbitraire ne résultant d'aucune comparaison.

Supposons donc que nous calculerons $\sqrt{x^2+1}$ si le Flag 1 est absent et $\sqrt{\frac{x^2+1}{x^2}}$ si le Flag 1 est présent.

En remarquant que $\sqrt{\frac{x^2+1}{x^2}} = \sqrt{1+\frac{1}{x^2}}$, on suivra l'organigramme suivant :



Programme :



Essai du programme :
Au départ, il n'y a pas de Flag, donc $\sqrt{x^2+1}$ doit être calculé par pression de **A**.

Presser	**Affichage**
1 **A**	1.41
0 **A**	1.00

Pour calculer $\sqrt{\frac{x^2+1}{x^2}}$, mettre le Flag 1 :

f SF1		
2 **A**	1.12	$(=\sqrt{\left(\frac{1}{2}\right)^2+1}$
5 **A**	1.02	
0 **A**	Flash !	(division par 0)

Le Flag reste dans l'état où on l'a mis.
Enlevons le pour calculer à nouveau $\sqrt{x^2+1}$

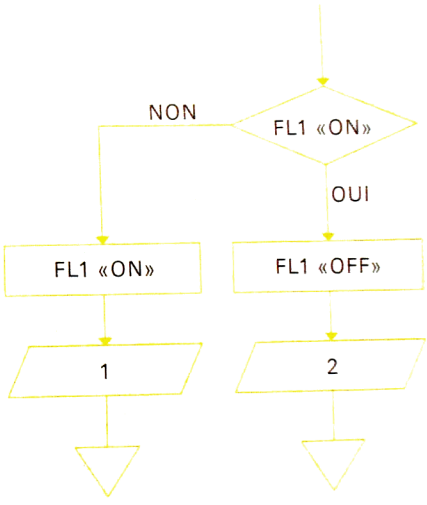
f1 SF1	
1 **A**	1.41
2 **A**	2.24

Structures-type de programmes utilisant des flags

1. Compteur de deux

Exemple de principe: Une partie de programme est exécutée toutes les deux pressions de la touche A.

LBL
A
f
TF1
GTO
1
1
f
SF1
RTN
LBL
1
2
f^{-1}
SF1
RTN



Presser	Affichage
A	1.00
A	2.00
A	1.00
A	2.00

Ce type de programme permet:

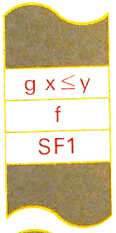
- d'introduire des données dans le même label, mais en sélectionnant le stockage de ces données
- de calculer des séries alternées, etc...

On pourrait imaginer de la même manière des compteurs de 4, 8, 16, etc...

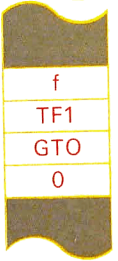
2. Combinaison d'un Flag avec un test de comparaison

L'état d'un Flag peut être fixé à la suite d'un test de comparaison, ce qui permet souvent d'économiser des pas de programme si la comparaison devait se faire plusieurs fois dans le même programme.

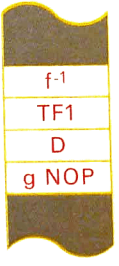
1) Le Flag 1 est mis si le résultat de la comparaison est OUI.



2) Le branchement au LBL 0 a lieu si le test de comparaison était vrai (le Flag 1 est alors présent).



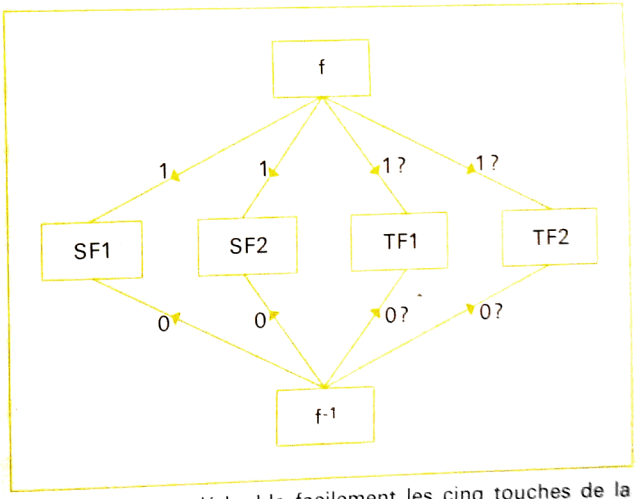
3) Le sous-programme D est exécuté si le test de comparaison était faux (pas de Flag 1 dans ce cas).



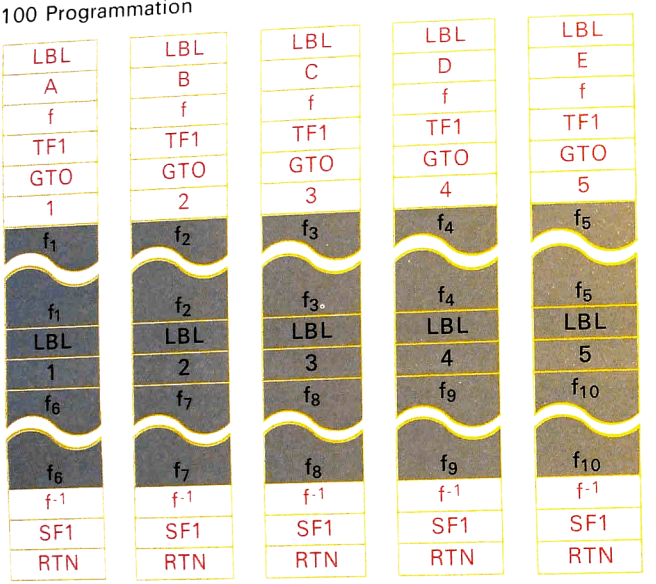
4) Enfin, le programme enlève le Flag 1 avant de s'arrêter (réinitialisation automatique).

L'utilisateur un peu averti trouvera bien d'autres cas où un Flag peut être utilisé et c'est là tout l'intérêt de la machine.

Les combinaisons offertes sont très nombreuses si l'on se souvient du schéma de base:



En particulier, on dédouble facilement les cinq touches de la ligne du haut **A**, . . . , **E** pour obtenir en fait 10 fonctions à définir de la manière suivante:



EXEMPLES COMPLETS DE PROGRAMMATION

Nous vous présentons deux exemples dont l'un est utilisable par chacun, puisqu'il s'agit de la statistique la plus élémentaire, l'autre exemple étant plus mathématique et traitant un calcul itératif. Ce second exemple vous permettra d'extrapoler ce genre de calcul à vos propres problèmes. Il prouve, s'il en était encore besoin, la puissance de calcul de votre calculatrice de poche!

Dans le premier exemple, nous nous attacherons surtout à montrer la progression réelle de la pensée de celui qui écrit un programme déterminé: ses hésitations en particulier. Il n'y a pas de programme idéal et l'on est toujours ramené à un choix en fonction des données que l'on peut introduire et des résultats que l'on peut espérer.

Pour ne pas prendre un exemple trop spécialisé, nous allons mettre en œuvre la statistique la plus simple et la plus connue de tous :

Calcul de la moyenne et de l'écart-type d'une série de données

Cette phrase à elle seule ne suffit évidemment pas à définir complètement notre problème.

Spécifications, limitations du problème
Nous allons nous borner à n'introduire que des données discrètes $x_1, x_2, \ldots, x_n$ (non liées à une fréquence d'apparition). Le programme devra permettre l'erreur d'introduction. (On sera capable d'enlever la donnée incorrecte pour réintroduire la donnée exacte.)

Ceci nous permettra de calculer des «moyennes mobiles», c'est-à-dire de faire des calculs successifs de moyenne en supprimant le premier terme et en y ajoutant un autre.

Recherche des formules (explicites ou implicites)
Le calculateur ne digérant pas les phrases mais les formules mémorisées sous forme de pas de programme, notre préoccupation va être d'isoler, si possible, l'expression de la moyenne $\bar{x}$ et de l'écart-type S à gauche d'un signal égal, le calcul étant le membre de droite.

NB : Dans certains problèmes, il est impossible d'expliciter ainsi les résultats. On peut facilement les résoudre sur le HP-65 par des méthodes dites itératives. Il suffit alors de trouver la solution sous la forme

$F(x) = 0$ (x étant l'inconnue)

et de faire des essais successifs de x jusqu'à ce que le résultat F (x) soit le plus près possible de 0.

Dans le cas présent, si nous avons introduit n données :

$(x_1, x_2, \ldots, x_n)$,

la moyenne est évidemment:

$$x = \frac{x_1 + x_2 + \ldots x_n}{n} = \left(\sum_{i=1}^{n} x_1 \right) \div n$$

L'écart-type (moyenne des écarts par rapport à la moyenne) se trouve dans n'importe quel traité de statistiques. Pour un échantillon:

$$S = \sqrt{\frac{\sum_{i=1}^{n} x_i^2 - n\bar{x}^2}{n-1}}$$

(Il existe un autre écart-type mais nous calculerons celui-ci qui est le plus utilisé.)

Nous tenons nos deux formules!

Analyse des formules

C'est l'étape la plus importante et la plus difficile. Au vu de la formule, il faut décider de la manière dont on va résoudre le problème!

Pour ce faire on «décortique» par la pensée sur le papier, les formules de $\bar{x}$ et S.

Analyse de $\bar{x}$

$$\bar{x} = \frac{\sum x_i}{n}$$

Première décision: Le nombre n étant en principe non limité, nous ne pouvons pas imaginer de mémoriser les données pour seulement ensuite en calculer la somme.

La somme partielle sera donc calculée après chaque introduction d'un nombre x_i.

Deuxième décision: Souvent, lorsqu'on fait une moyenne portant sur de nombreuses valeurs, on ne connaît pas et l'on n'a pas envie du tout de connaître le nombre n de données.

Le nombre n sera «initialisé» à zéro, et «incrémenté» de 1 à chaque introduction d'un nombre.

Ayant examiné le problème élémentaire posé par n et Σx_i, nous ne pouvons guère faire mieux.

Analyse de S

$$S = \sqrt{\frac{\Sigma x_i^2 - n\bar{x}^2}{n-1}}$$

Première constatation : S ne peut être déterminé que lorsque $\bar{x}$ l'aura été.

Mais : Σx_i^2 doit être calculé par sommes successives (on ajoute à la somme mémorisée le carré du nombre venant d'être introduit et on mémorise ainsi la nouvelle somme).

Nous avons fait le tour des problèmes posés par les formules, mais ce n'est pas tout.

Le problème est-il soluble sur le HP-65?
Quand vous vous posez ce genre de question, il faut avoir deux réactions principales:

1) Tout problème ne comportant pas plus de 13 ou 14 «paramètres» à mémoriser au même instant de l'exécution, ou une boucle de plus de 100 pas, est soluble par le HP-65!

En effet, ce n'est pas la complexité de l'équation ou la longueur du programme qui limite notre calculateur: on peut enchaîner les cartes, donc avoir des programmes de plus de 100 pas.

La pile opérationnelle, les registres, les conditions de calcul (mode d'angles, Flags...) ne sont absolument pas affectés par la lecture d'une carte.

Seule la mémoire-programme est «écrasée» par les 100 nouveaux pas lus sur la carte.

2) Si le nombre de manipulations délicates à l'exécution, le mode opératoire, les précautions risquent d'être diaboliques, il faut penser que ce genre de problème n'est pas adapté à votre calculateur. (Il est néanmoins «faisable» dans les conditions exposées en 1.)

Dans notre cas, nous aurons au maximum et en n'étant pas bien malins quatre résultats intermédiaires ou définitifs à stocker dans des registres mémoire adressable ou dans la pile.

Pas de problème donc!

Conception de la carte

Il paraît bizarre que l'on pense à cela maintenant, avant même d'avoir écrit le premier pas de notre programme. Et pourtant c'est ce qu'il faut faire. Lorsque vous aurez écrit de nombreux programmes, c'est même ce que vous ferez en premier. Une carte doit être très indicatrice, par conséquent l'idéal serait qu'elle puisse nous indiquer tout, le mode opératoire devenant pratiquement inutile.

Réfléchissons un peu:

Nous avons à faire une initialisation de:

n à 0

Σx_i à 0

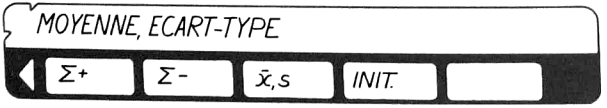
Σx_i^2 à 0

Une touche «INIT» serait donc souhaitable.

Nous introduisons les nombres un par un; il paraît facile d'utiliser la même touche pour chaque introduction en l'intitulant par exemple: « $\Sigma+$ », ce qui est assez explicite.

Nous avons besoin d'une touche «erreur». Vu la manière dont nous allons calculer, cette touche ne peut faire que l'inverse de la touche « $\Sigma+$ », c'est-à-dire « $\Sigma-$ ». Il faudra, bien entendu, réintroduire le nombre faux (ou déjà utilisé dans le calcul d'une première moyenne pour la moyenne mobile), puis presser la touche en question pour «effacer» complètement l'effet produit par cette erreur.

La carte, à laquelle nous arrivons, est donc:



Bien entendu, il faut une touche pour afficher la moyenne $\bar{x}$ et l'écart-type S.

Comme il nous reste une touche, nous aurions pu penser afficher S par cette touche, ce qui n'aurait pas été bien malin:

> On ne peut pas calculer S si l'on n'a pas d'abord obtenu $\bar{x}$: séparer les calculs de $\bar{x}$ et de S dans deux touches implique que l'on peut tenter d'avoir S sans avoir eu $\bar{x}$. Dans ce cas, le calcul donne évidemment n'importe quoi.

Pour des raisons de facilité d'emploi, nous choisissons donc d'avoir $\bar{x}$ et S l'un après l'autre et dans l'ordre $\bar{x}$ puis S.

> La pression de **C** donnera $\bar{x}$
> la pression de **R/S** donnera S (arrêt non définitif sur un R/S).

Nous avons ainsi défini tous nos «labels».

Organigramme

L'organigramme est la représentation logique simplifiée de notre problème. Il donne l'ordre dans lequel sont exécutées les différentes phases au sein d'un même programme. Il n'est pas nécessaire de décortiquer complètement le calcul dans cette représentation.
Comme nous avons pu discerner 4 labels différents et presque indépendants (en tous cas au point de vue entités de programmes), nous avons quatre petits organigrammes.

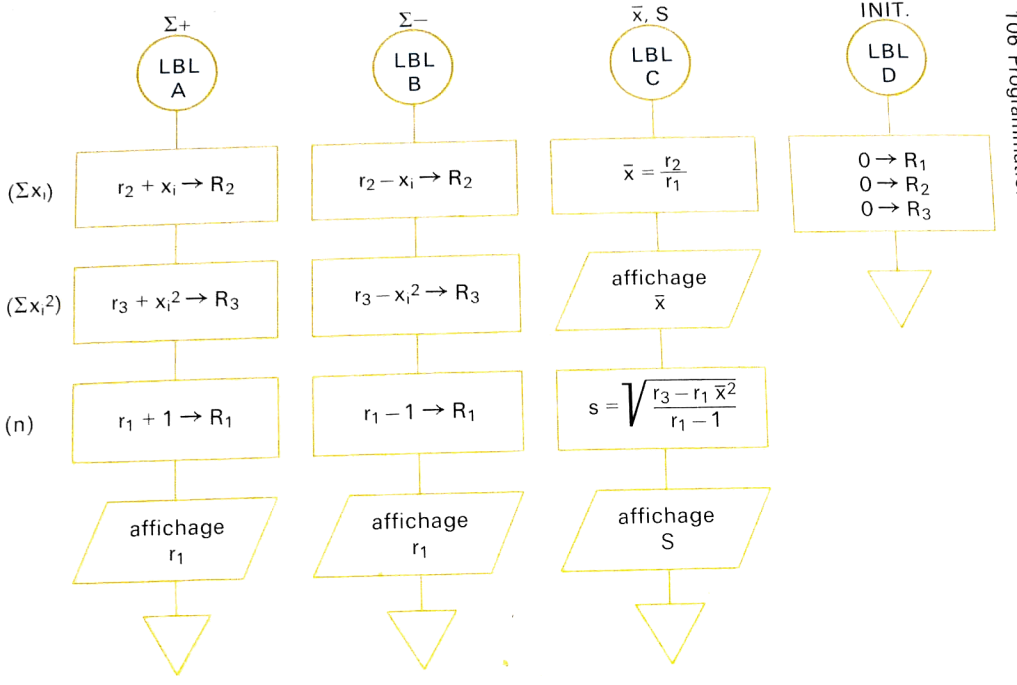
Nous utilisons les registres de la manière suivante:

Registre R_1: conservation de n (initialisé)
Registre R_2: conservation de Σx_i (————)
Registre R_3: conservation de Σx_i^2 (————)

Pourquoi n'avons nous pas commencé par le calcul de n dans les fonctions **A** et **B**? Pour la simple raison qu'il vaut mieux afficher en dernier lieu la valeur du n courant (du i), ce qui permet à l'utilisateur de connaître l'indice, donc le nombre, des points introduits.

Passage au programme:

Quand on a l'organigramme, le passage est très rapide, les seuls problèmes restant encore à résoudre concernent plutôt une bonne connaissance des possibilités du calculateur en utilisation sans programme.


Σ+
LBL A
(Σxi)
r2 + xi → R2
(Σxi2)
r3 + xi2 → R3
(n)
r1 + 1 → R1
affichage r1
Σ−
LBL B
r2 − xi → R2
r3 − xi2 → R3
r1 − 1 → R1
affichage r1
x̄, S
LBL C
x̄ = r2 / r1
affichage x̄
s = √((r3 − r1 x̄2) / (r1 − 1))
affichage S
INIT.
LBL D
0 → R1
0 → R2
0 → R3

Passer en mode W/PRGM.

Presser f PRGM pour supprimer le programme placé en mémoire, ou le programme définissant «par défaut» les touches A, B, . . . , E

Title: *Moyenne, écart-type* Page 1 of 1

SWITCH TO W/PRGM PRESS f PRGM TO CLEAR MEMORY

KEY ENTRY	CODE SHOWN	COMMENTS
LBL	23	
A	11	
STO	33	$r_2 + x_i \rightarrow R_2$ (Σx_i)
+	61	
2	02	
f-1	32	
$\sqrt{x}$	09	
STO	33	$r_3 + x_i^2 \rightarrow R_3$ (Σx_i^2)
+	61	
3	03	
RCL 1	34 01	
1	01	$r_1 + 1 \rightarrow R_1$ (n)
+	61	
STO 1	33 01	
RTN	24	n affiché
LBL	23	
B	12	
STO	33	$r_2 - x_i \rightarrow R_2$
−	51	
2	02	
f-1	32	
$\sqrt{x}$	09	
STO	33	$r_3 - x_i^2 \rightarrow R_3$
−	51	
3	03	
RCL 1	34 01	
1	01	$n - 1 \rightarrow R_1$
−	51	
STO 1	33 01	
RTN	24	n affiché
LBL	23	
C	13	
RCL 2	34 02	
RCL 1	34 01	
÷	81	
RTS	84	$\bar{x}$ affiché
f-1	32	
$\sqrt{x}$	09	
RCL 1	34 01	
x	71	
RCL 3	34 03	
$R \leftrightarrow y$	35 01	
−	51	
RCL 1	34 01	
1	01	
−	51	
÷	81	
f	31	
$\sqrt{x}$	09	
RTN	24	S affiché

KEY ENTRY	CODE SHOWN	COMMENTS
LBL	23	
D	14	
f	31	$0 \rightarrow R_1, R_2, R_3$
REG	43	
RTN	24	

REGISTERS

R_1 n

R_2 Σx_i

R_3 Σx_i^2

R_4

R_5

R_6

R_7

R_8

R_9

LABELS

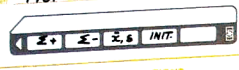
A $\Sigma+$
B $\Sigma-$
C $\bar{x}$, s
D INIT
E
0
1
2
3
4
5
6
7
8
9

FLAGS

1
2
3

Title: Moyenne, écart-type — Page 1 of 1

Programmer: Moi — Date: 30/4/73



STEP	INSTRUCTIONS	INPUT DATA/UNITS	KEYS	OUTPUT DATA/UNITS
1	Introduire la carte		D	
2	Initialiser		A	
3	Pour chaque donnée x_i	x_i		i
	$i = 1, \ldots, n$ faire 3			
4	Pour une donnée à enlever x_k	x_k	B	$k-1$
	Faire 4			
5	A tout moment, calcul de $\bar{x}$ et de s		C	$\bar{x}$
			R/S	s
6	Pour continuer avec cette même série de données, aller en 3 ou 4. Faire 5 pour recalculer une moyenne ou un écart-type. Pour un nouveau cas, aller en 2.			

Essai du programme:

Nous connaissons les résultats de nos ventes jusqu'au 9e mois de l'année et nous désirons calculer les moyennes suivantes:

moyenne des ventes du 1er au 6e mois
moyenne des ventes du 2e au 7e mois
moyenne des ventes du 3e au 8e mois
moyenne des ventes du 4e au 9e mois

Ce problème de moyennes mobiles est résolu très simplement par le calculateur et le programme que nous venons d'écrire:

Mois	1	2	3	4	5
Ventes	1230	1502	1789	1438	1242

Mois	6	7	8	9
Ventes	1681	1702	1725	1689

Il nous suffit d'introduire les 6 premiers mois, d'en faire la moyenne (et l'écart-type), puis de retrancher le premier mois ($\Sigma -$) et d'ajouter le 7e mois pour recalculer la deuxième moyenne, etc...

On obtient Périodes	1→6	2→7	3→8	4→9
Moyenne	1480.33	1559.00	1596.17	1579.50
Ecart-type	226.84	203.28	211.01	196.16

Exemple II : programme itératif

Approximation de la racine carrée d'un nombre par la méthode de Newton.

Le HP-65 permet le calcul de la racine carrée d'une manière automatique par simple pression d'une ou de deux touches. Il est d'ailleurs possible que cette méthode itérative soit utilisée dans les circuits électroniques de la machine.

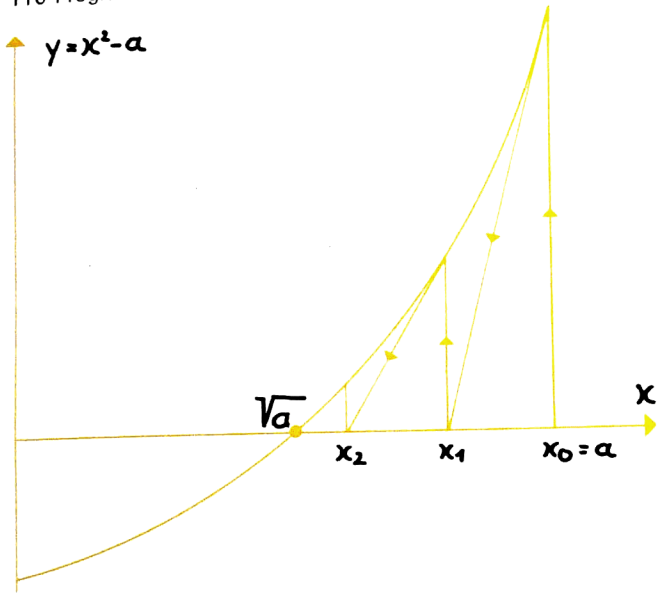
On cherche la racine carrée d'un nombre a (si possible positif ou nul !).

Un résultat mathématique bien connu (théorème du point fixe) nous affirme que la suite ci-dessous converge vers $\sqrt{a}$.

$$x_o = a$$

$$x_{n+1} = \frac{1}{2}\left(x_n + \frac{a}{x_n}\right) \quad \text{avec } x > 0$$

On peut en avoir une représentation sur un graphique en portant $y = x^2 - a$ en fonction de x.



On trace la tangente à la courbe au point d'abscisse $x_o = a$, cette tangente coupe l'axe des x au point d'abscisse x_1. De même à partir de x_1 pour obtenir x_2, etc... Graphiquement on se rend compte de la convergence vers $\sqrt{a}$.

La méthode choisie ici est une méthode de résolution approchée. Il se pose donc le problème de l'arrêt des calculs. On choisit le nombre très petit ε (par exemple 10^{-4}) et on s'arrêtera lorsqu'on aura atteint le premier terme x_m tel que:

$$|x_m^2 - a| < \varepsilon$$

Le programme que nous allons écrire doit:

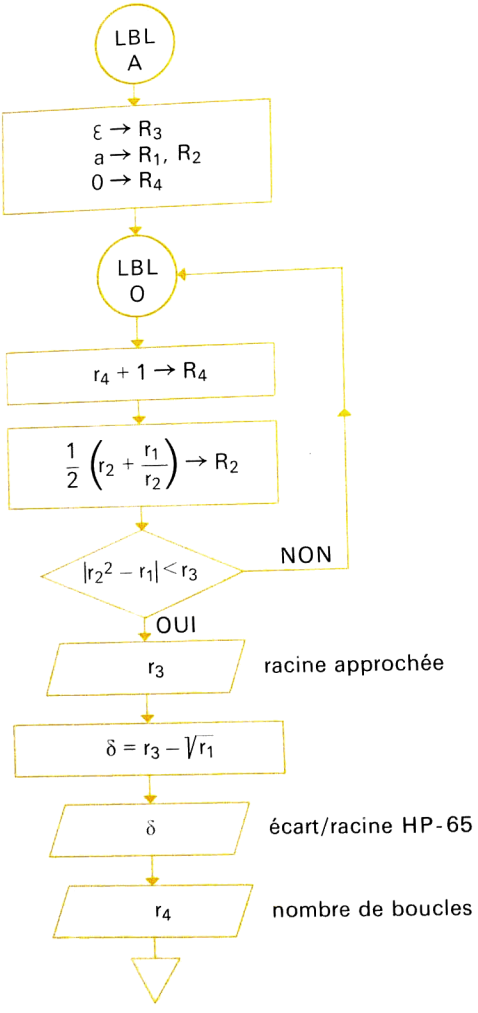
1) refuser de calculer la racine d'un nombre négatif (test d'entrée)
2) comparer la valeur obtenue par itération à la valeur $\sqrt{a}$ calculée par les touches du HP-65

3) compter le nombre d'itérations afin de se faire une idée de la fonction:
nb. d'itérations = f (précision demandée).

L'analyse étant faite, passons à l'organigramme (dans le cas de programme itératif, on parle d'algorithme).

La valeur de a est stockée dans le registre R_1.
Les valeurs $x_0, x_1, \ldots, x_m$ seront stockées dans R_2.
ε dans R_3.
Le compteur de boucles est initialisé à 0 et placé dans R_4.
Les données a et ε sont placées respectivement:
ε dans le registre X de la pile
a dans le registre Y de la pile
et l'on utilisera seulement une touche de fonction à définir pour ce programme.

Organigramme :



Racine carrée par la méthode de Newton

KEY ENTRY	CODE SHOWN	COMMENTS
LBL	23	ε dans X, a dans Y
A	11	
STO 3	33 03	
g ŷ	35 08	
STO 2	33 02	
STO 1	33 01	
0	00	
STO 4	33 04	
LBL	23	Adresse de retour
0	00	
1	01	
STO	33	
+	61	
4	04	
RCL 1	34 01	
RCL 2	34 02	
÷	81	
RCL 2	34 02	Boucle totale
+	61	
2	02	
÷	81	
STO 2	33 02	
f-1	32	
$\sqrt{x}$	09	
RCL 1	34 01	
−	31	
g	35	
ABS	06	
RCL 3	34 03	
g x≤y	35 22	Test
GTO	22	
0	00	
RCL 2	34 02	
R/S	84	Affichage de $\sqrt{a}$ calculé
RCL 1	34 01	
f	31	
$\sqrt{x}$	09	
−	51	
DSP	21	Affichage scientifique
g	09	de l'écart δ
R/S	84	
RCL 4	34 04	
DSP	21	
.	83	Affichage à deux
2	02	décimales
RTN	24	

KEY ENTRY	CODE SHOWN	COMMENTS

REGISTERS

R_1 a

R_2 X_n

R_3 ε

R_4 m

R_5

R_6

R_7

R_8

R_9

LABELS

A B C D E 0 1 2 3 4 5 6 7 8 9

FLAGS

1

2

Title: Racine carrée par la méthode de Newton — Page ___ of ___

Programmer: — Date

STEP	INSTRUCTIONS	INPUT DATA/UNITS	KEYS	OUTPUT DATA/UNITS
1	Introduire la carte			
2	Introduire le nombre a	a	↑	
	et la précision demandée	ε	A	$\sqrt{a}_c$
3	Comparaison avec le $\sqrt{a}$ de la machine		R/S	écart
4	Nombre d'itérations		R/S	nombre
	Pour un nouveau cas, aller en 2			

Essai du programme

Calcul de $\sqrt{625}$ (a = 625)
avec différentes valeurs de ε.

Pour exécuter ce calcul, se reporter au mode opératoire.
Pour afficher la valeur de $\sqrt{a_c}$ calculée par itérations, faire DSP • 8.

ε = 1	$\sqrt{a_c} = 25.00177597$	$\delta = 1.775970000 \times 10^{-3}$	m = 7
ε = 0.1	mêmes résultats		
ε = 0.01	$\sqrt{a_c} = 25.00000007$	$\delta = 7 \times 10^{-8}$	m = 8
ε = 10^{-6}	$\sqrt{a_c} = 25.00000000$	$\delta = 0 \times 10^{0}$	m = 9

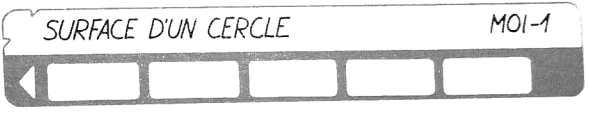
La suite est très convergente.

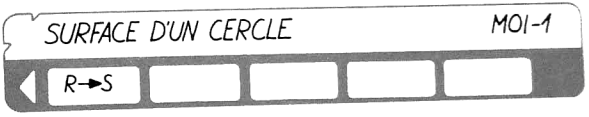
QUELQUES «TRUCS» POUR AMÉLIORER VOS PROGRAMMES

En tant que créateur et utilisateur de vos programmes, vous devez, successivement dans le temps, vous placer dans les deux positions et vos préoccupations seront alors très différentes:

- Vous êtes le programmeur-analyste!
  Votre programme doit être le plus court possible, «tourner» le plus vite possible et marcher correctement.

- Vous êtes aussi l'utilisateur!
  Une fois le programme écrit, il faut absolument qu'il soit facile à utiliser, c'est-à-dire en particulier que les indications figurant dans les petits rectangles de votre carte-programme soient en mesure de vous donner le maximum d'informations quant à l'utilisation du programme.





De ces deux cartes enregistrées, il est bien évident que la deuxième ne nécessite aucune explication quant au mode opératoire, à la mise en œuvre de l'exécution:

Introduire R, presser **A**, S s'affiche!

C'est lumineux.

La première carte n'est pas explicite car:

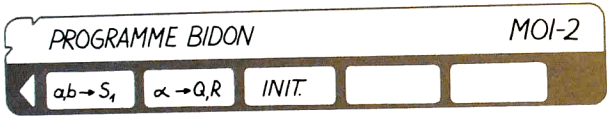
1. Elle n'indique pas quelle touche donne accès au programme. Ce n'est pas très grave si une seule fonction a été utilisée. Dans ce cas, on pourrait presser indifféremment **A** **B** **C** **D** **E** ou **R/S** pour calculer S. Ceci n'est pas valable si deux fonctions au moins ont été programmées.
2. Il y a seulement une donnée à introduire, donc pas de problème. S'il y en avait deux ou plus, comment les introduire?

Un autre point important du point de vue utilisateur est que, le programme créé, on doit avoir une confiance aveugle

en son exécution. Par conséquent, tous les cas particuliers doivent être analysés lors de la conception du programme (ce qui l'allonge!... le programmeur n'est pas content).
Autre possibilité (moins intéressante): on ne teste pas les cas particuliers mais on indique clairement que telle ou telle donnée ne convient pas pour ce programme. Bien sûr, cela oblige à relire l'explication lors de chaque utilisation du programme!

EXEMPLE DE CARTE BIEN REMPLIE!

- La fonction **A** calcule S_1 avec a et b comme données.
- La fonction **B** calcule Q et R avec α comme donnée.
- La fonction **C** sert à initialiser le programme.
- **D** et **E** ne sont pas utilisés.



Là encore, le programmeur doit avoir une qualité essentielle: *l'unité* dans la présentation d'un programme.

Pour la fonction **A** calculant S_1, la seule vue de la carte ne permet pas de savoir comment a et b seront introduits. On peut essayer de deviner:

	RTN	a	R/S	b	A	?
ou	a	↑	b	A		?
ou	a	A	b	R/S		?, etc.

Si le programmeur décide de mettre sur la carte:

a, b → S_1

et que chaque fois qu'il écrit ceci, il programme la fonction pour que les données soient introduites par:

a ↑ b **A**,

le programme sera facile à utiliser et cette méthode sera rapidement assimilée par l'utilisateur.

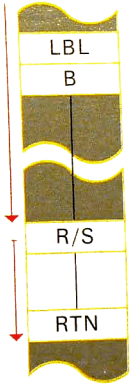
Autre possibilité (encore plus explicite) : la carte sera identifiée par

$$a + b \rightarrow S_1$$

Lorsque plusieurs résultats sont inscrits sous la même lettre (exemple : B, le problème est facile à résoudre :

la pression de B donne Q
R/S donne R.

Dans ce cas, le programme a obligatoirement l'allure suivante :



Introduire α
presser B

Premier résultat Q
presser R/S

(ou R/S)

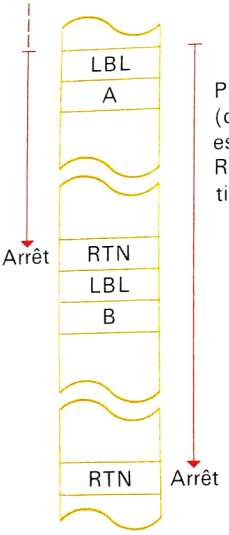
Deuxième résultat R

Attention aux touches RTN et R/S.

Dans un programme :
– le pas RTN est la fin logique d'une fonction à définir.

Oui mais : Si RTN est effectivement placé comme dernier pas de cette fonction, on doit obligatoirement appeler la fonction par la lettre correspondante. On ne peut pas démarrer le programme par R/S, même si le pointeur est positionné au début de la fonction. Dans ce cas, l'exécution ne s'arrête pas au RTN de la fonction mais au R/S ou RTN suivant sur la boucle (qui n'ont rien à voir avec la fonction à calculer). Le résultat est, bien sûr, faux dans ce cas.

Exemple



RTN pressé en mode RUN, comme une commande, ramène le pointeur en début de mémoire. Dans ce cas, le programme s'arrête :

- sur le premier **R/S** trouvé
- sur le deuxième **RTN** trouvé

dans la boucle si on presse **R/S** (voir explications du cas précédent).

CONCLUSION
On a intérêt à arrêter une fonction par un **R/S** sauf si cette fonction doit servir de sous-programme (**RTN** est alors impératif).

R/S dans un programme arrête son exécution quelles que soient les touches que l'on a pu presser pour le démarrer.

R/S comme commande (en mode RUN) démarre l'exécution du programme à partir de la position actuelle du pointeur:

- Si l'on avait préalablement pressé une touche **A** . . . **E**, le programme s'arrête sur le **RTN** correspondant à ce niveau. (Il peut y avoir un sous-programme!)
- Si l'on n'avait pas pressé de touches alphabétiques, le programme s'arrête:
  - à coup sûr sur un **R/S**
  - jamais sur le **RTN** suivant qu'il rencontre.

R/S et index affiché

Exemple: Supposons qu'un programme A calcule le résultat intermédiaire R_1 et que pour une certaine raison, nous ayons besoin de connaître l'index de ce résultat pour introduire alors la donnée supplémentaire permettant d'obtenir le résultat définitif.

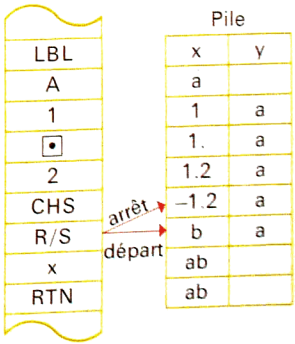
On peut faire afficher un nombre de un ou plusieurs chiffres juste avant de programmer un **R/S**, à condition que l'index affiché soit programmé par des touches de chiffres [0] [1] . . . [9] (pas de **RCL** [LST x], etc...). Si nous introduisons une nouvelle donnée pendant cet arrêt, la pile ne monte pas! L'index est simplement remplacé par la nouvelle entrée dans le registre X.

Exemple

Introduction de a
Pression de **A**

–1.2 est affiché comme index
Introduction de b
Pression de **R/S**

a×b est affiché!
(et non pas bx(–1.2))



Essai

4 [A] → –1.2
5 [R/S] → 20.00 (4×5)

1. MISE AU POINT RAPIDE DES PROGRAMMES

On a vu que l'on pouvait exécuter un programme pas par pas à l'aide de la touche [SST] en mode RUN, un sous-programme est exécuté comme un seul pas. Une méthode tout aussi rapide pour vérifier un programme long est de placer quelques [R/S] pour afficher des résultats intermédiaires significatifs. Ainsi, par pression de [R/S], on exécutera des parties de programme complètes et on pourra juger d'un coup si elles sont justes.
Ne pas oublier d'enlever les [R/S] de mise au point avant d'enregistrer le programme sur carte.

2. CAS DES LONGS PROGRAMMES

On peut très bien n'utiliser aucune des touches de fonctions à définir pour écrire un programme très long. Dans ce cas, les commandes et les arrêts dans le programme ne peuvent être que des [R/S].

Il est évident que la feuille d'instructions de l'utilisateur sera, dans ce cas, nécessaire pour exécuter le programme à chacune de ses utilisations.

On programme directement les pas sans aucun [LBL] [A] . . . [E] à partir du début de la mémoire. Pour ramener le pointeur au début de cette mémoire en exécution, presser [RTN], introduire les premières données puis presser [R/S] pour démarrer l'exécution.

Si un programme, même très travaillé et affiné, dépasse les 100 pas de la mémoire, il ne tiendra pas sur une carte.

On peut enchaîner autant de cartes que l'on veut en se souvenant que seule la mémoire-programme est entièrement remplacée par lecture d'une carte.

En particulier, la pile et les registres ne sont pas affectés.

3. DÉPASSEMENTS DE CAPACITÉ

Un programme s'arrête lors d'un dépassement de capacité, c'est-à-dire lorsqu'il est incapable de calculer en raison des arguments qui sont trop grands (dépassement supérieur) ou trop petits (dépassement inférieur).

Dépassement supérieur:
Si le résultat du calcul devait être plus grand que 10^{99}, le programme affiche: **9.99999999 10^{99}** et s'arrête.

Dépassement inférieur:
Si le résultat d'un calcul est inférieur à 10^{-99}, le programme affiche: **0.00** et s'arrête.

Ce dépassement est plus difficile à déceler. On peut toutefois passer en mode W/PRGM pour identifier le pas où le programme est arrêté.
Si ce pas est différent d'un **R/S** ou **RTN**, il y a dépassement de capacité.

NB: A chaque exécution d'un programme pas à pas, on peut passer en W/PRGM mode temporairement pour vérifier les pas (utile en cas de clignotement de l'affichage).

DÉFINITIONS (pour le HP-65)

PROGRAMME

Séquence de pas (1 ou 2 touches) implantée en mémoire-programme.

SOUS-PROGRAMME

Séquence de pas appelée par un pas alphabétique (**A** . . . **E**) d'une autre séquence. Un sous-programme commence obligatoirement par **LBL** **A** . . . **E** et finit obligatoirement par **RTN**. Il ramène la suite de l'exécution au programme appelant dit principal.

BOUCLE

Séquence de pas répétée plusieurs fois au cours de l'exécution.

BRANCHE

Séquence de pas à l'intérieur d'un programme. Un programme est composé d'une ou plusieurs branches.

POINTEUR

Organe directeur de l'exécution d'un programme. Mouvement dans un sens unique sur le HP-65.

TEST

Question posée à la machine dans le programme. Suivant le résultat, deux pas de programme sont sautés (si réponse NON) ou non (si réponse OUI).

BRANCHEMENT INCONDITIONNEL

Pas indiquant au pointeur qu'il faut suspendre l'exécution pour la reprendre à une certaine adresse.

ADRESSES

Points de repère placés dans le programme. Peuvent être alphabétiques **A** . . . **E** (fonctions à définir) ou numériques [0] . . . [9] (points de branchements).

BRANCHEMENT CONDITIONNEL

Branchement programmé dans les deux pas suivant un test.

DONNÉE

Nourriture indispensable d'un programme.

VARIABLE

Représentation d'une donnée (symbole)

PAS DE PROGRAMME

Elément de base d'un programme. Le HP-65 a une capacité de 100 pas, chaque pas correspondant à une ou deux pressions de touches (code combiné).

CODE

Représentation d'un pas en mémoire. Le codage des touches est matriciel sauf pour les touches de chiffres (code naturel).

CODE COMBINÉ

Pour certaines touches préfixées, deux pressions ne donnent qu'un pas codé sur quatre chiffres.

REGISTRE

Mémoire de nombres. On distingue les registres de la pile qui permettent les opérations (registres X, Y, Z, T, last X actifs) des registres de stockage $R_1 \ldots R_9$ qui permettent le rangement et le rappel de nombres.

COMPTEUR

Organe essentiel d'un bouclage. La machine permet d'utiliser facilement un compteur en liaison avec le registre 8.
Permet d'effectuer N fois la même séquence de pas.

DÉPASSEMENT DE CAPACITÉ

Calcul refusé par la machine, en raison des arguments:
– dépassement supérieur de capacité: le nombre calculé est plus grand que 10^{99};

– dépassement inférieur de capacité: le nombre calculé est inférieur à 10^{-99}.

Dans les deux cas, le programme s'arrête!

LABEL

Préfixe d'une adresse pour le HP-65.

LISTING

Par analogie avec un ordinateur, le listing est l'affichage pas à pas d'un programme (touche **SST**).

ORGANIGRAMME

Représentation symbolique d'un problème.

Adresse de fonction ou de branchement (label)

Calcul et stockage

Affichage et arrêt (définitif ou non)

Test

Arrêt final

ANNEXE A

LIMITES OPÉRATIONNELLES

PRÉCISION

La précision du HP-65 varie selon l'opération exécutée. Dans le cas de fonctions transcendantes, les erreurs relatives données ci-dessous sont à considérer comme des ordres de grandeur. Les limites de précision ne sont donc données ici qu'à titre indicatif; elles définissent l'erreur maximale pour les différentes fonctions.

Dans le cas d'opérations simples telles que g [x], f [$\sqrt{x}$], f-1 [→D.MS], l'erreur maximale est de ±1 sur le dixième chiffre (chiffre le moins significatif). Pour ces opérations, les erreurs se produisent lorsque la machine arrondit les réponses au dixième chiffre.

Exemple: calcul de $(\sqrt{5})^2$. L'arrondi de $\sqrt{5}$ à dix chiffres significatifs donne 2,236067977, dont le calcul du carré donne un nombre à 19 chiffres: 4,999999997764872529 qui, arrondi à 10 chiffres, donne 4,999999998. Si vous calculez le carré du nombre approximé par excès suivant: 2,236067978, vous obtenez 5,000000002. Il n'existe pas de nombre de dix chiffres dont le carré est 5,000000000.

Lors de la soustraction de nombres ayant 10 chiffres significatifs, la réponse est correcte à ±1 sur le dixième chiffre le moins significatif de l'opérande ayant la valeur algébrique la plus élevée. La fonction factorielle (g [$n!$]) est précise à ±1 unité sur le neuvième chiffre. Les valeurs converties en degrés, minutes, secondes f [→D.MS] sont correctes à ±1 seconde, de même que les résultats des conversions telles que: f [D.MS+] et f-1 [D.MS+].

La précision des autres opérations (trigonométriques, logarithmiques et élévation à une puissance) dépend de l'argument. La réponse fournie sera la valeur exacte d'un argument d'entrée

à ± N unités près sur le dixième chiffre le moins significatif de l'argument d'entrée initial, N étant donné dans le tableau ci-après. Par exemple, votre HP-65 indique que le ln 5 = 1.609437912. En fait, il ne s'agit que d'une approximation puisque 1,609437912 n'est en réalité que le logarithme d'un nombre compris entre 4.999999998 et 5.000000002, ce qui traduit bien l'erreur de ± 2 unités sur le chiffre le moins significatif (10e chiffre) du véritable argument d'entrée (N = 2 pour les logarithmes).

Opération	Valeurs de N
f LOG f LN f^{-1} LN	2*
trigonométriques	3**
g y^x	4 pour **y** et 10 pour **x**
f^{-1} LOG	7
f R→P f^{-1} R→P	4

* Ces opérations sont en outre limitées à ±1 unité pour f LN et ±3 unités pour f LOG sur le 10e chiffre le moins significatif du nombre affiché.

** Les opérations trigonométriques présentent une limitation supplémentaire de $\pm 1 \times 10^{-9}$ sur le résultat affiché.

AFFICHAGE LORS D'UN DÉPASSEMENT DE CAPACITÉ

Afin d'obtenir la plus grande précision, le HP-65 effectue tous ses calculs avec un nombre de dix chiffres suivi d'une puissance de dix. Cette manière abrégée d'exprimer les nombres est dite notation scientifique. Par exemple: 23712,45 s'affiche 2.371245×10^4.

DÉPASSEMENT INFÉRIEUR DE CAPACITÉ

Si un résultat est trop petit pour tenir dans un registre, le registre est mis à zéro et le programme s'arrête s'il est en cours d'exécution.

DÉPASSEMENT SUPÉRIEUR DE CAPACITÉ

Si le résultat d'un calcul dépasse la capacité d'un registre, ce dernier est remplacé par des 9 (avec signe approprié), ce qui représente le nombre le plus grand pouvant tenir dans un registre et le programme s'arrête s'il est en cours d'exécution.

TEMPÉRATURES DE FONCTIONNEMENT

Limites globales:	0° à 40° C
Limites du lecteur de cartes:	5° à 40° C en lecture
	10° à 40° C en écriture

ANNEXE B

ACCESSOIRES

Pour commander des accessoires (standards ou optionnels), remplir la carte de commande et la retourner à Hewlett-Packard.

ACCESSOIRES FOURNIS AVEC LE CALCULATEUR

Votre HP-65 vous est livré avec un exemplaire de chacun des accessoires suivants:

Accessoires	Références N°
Batterie	82001A
Chargeur/adaptateur (110/220 V)	82010A
Ecrin de transport	82018A
Etui souple	82017A
Manuel d'utilisation du HP-65	00065-90219
Etiquettes personnalisées (4)	7120-2946
Aide-mémoire du HP-65	00065-90224
Bibliothèque «Programmes types» comprenant:	00065-67012

- 19 programmes préenregistrés sur cartes magnétiques
- 20 cartes magnétiques vierges
- 20 porte-cartes en carton de format de poche, chacun étant prévu pour 2 cartes magnétiques et modes opératoires
- 1 carte de nettoyage des têtes
- 1 fascicule d'utilisation

Cahier de 50 feuilles de programmation 9320-0616

ACCESSOIRES OPTIONNELS

D'autres accessoires, dont les bibliothèques de programmes spécialisées, sont indiqués sur la carte se trouvant à la fin du présent manuel.

Accessoires optionnels	Références N°
Porte-batterie + batterie	82004A
Berceau de sécurité	82015A
Etui de voyage	82016A
Cartes magnétiques vierges (40)	00065-67010
Cartes magnétiques vierges (100)	00065-67032
Cahier de 50 feuilles de programmation	9320-0616
Porte-cartes en carton de format de poche (20)	9320-0613

Le porte-batterie avec batterie (HP 82004A) comporte un dispositif de charge et une batterie de rechange, ce qui permet de recharger une batterie pendant que l'autre est utilisée.

Hewlett-Packard diffusera ultérieurement d'autres bibliothèques de programmes spécialisées.

ANNEXE C

SERVICE ET MAINTENANCE

PROCÉDURES DE CONTRÔLE DU HP-65

NOTE

Rechargez la batterie avant de mettre votre calculateur dans votre poche.

Votre calculateur est doté d'une batterie rechargeable, et il peut fonctionner pendant que la batterie se recharge; toutefois, si vous envisagez d'emporter votre calculateur, prenez soin de recharger la batterie (14 heures). Peu importe la position du commutateur «OFF/ON» pour recharger la batterie.

ATTENTION

Le calculateur peut être détérioré par des charges statiques élevées.

BAISSE DE PUISSANCE

Tous les points décimaux de l'écran s'allument ensemble, signalant ainsi à l'utilisateur qu'il ne lui reste plus que 2 à 5 minutes de fonctionnement. Trois solutions se présentent:

- travailler sur le secteur
- recharger la batterie
- remplacer la batterie par une batterie chargée correctement.

ABSENCE D'AFFICHAGE

En cas d'absence d'affichage, mettre le HP-65 hors de service, le commutateur W/PRGM-RUN sur la position RUN, puis le

remettre en marche. Si l'indication 0.00 n'apparaît toujours pas, vérifier les points suivants :

1. Si le chargeur de batterie est relié au HP-65, vérifier qu'il est bien branché sur la bonne tension-secteur.
2. Vérifier que la batterie n'est pas déchargée et que les contacts ne sont pas défectueux.
3. S'il n'y a toujours pas d'affichage, faire fonctionner le HP-65 sur le secteur.
4. Enfin, si l'affichage est toujours éteint, le HP-65 est défectueux (voir ci-dessous paragraphe «Garantie»).

GARANTIE

APPAREIL SOUS GARANTIE

Le HP-65 est garanti contre tous vices de matières ou de fabrication. Cette garantie s'applique pendant une durée d'un an à compter de la date de livraison. Hewlett-Packard s'engage à réparer ou à remplacer les composants qui se révéleraient défectueux pendant la période de garantie, à condition que les calculateurs défectueux lui soient retournés, conformément aux instructions données au paragraphe «Instructions d'expédition» (voir page 132).

Cette garantie ne s'applique pas si le calculateur a été détérioré par accident, par suite d'une fausse manœuvre, ou à la suite d'une intervention ou d'une modification exécutée par une quelconque personne, en dehors d'un centre agréé Hewlett-Packard.

Aucune autre garantie explicite ou implicite n'est accordée. Hewlett-Packard ne sera pas responsable des dommages indirects.

APPAREIL HORS GARANTIE

Après l'expiration du délai de garantie, l'appareil sera réparé au plus juste prix. Renvoyer le calculateur avec sa batterie et son chargeur/adaptateur dans son écrin de transport (voir instruc-

tions d'expédition). Si la batterie seule est défectueuse, commander alors une nouvelle batterie.

INSTRUCTIONS D'EXPÉDITION

En cas de défauts de fonctionnement imputables au calculateur ou au chargeur de batterie, il faut nous retourner:

1. le HP-65 avec sa batterie et son chargeur/adaptateur, dans son écrin de transport;
2. **une carte d'entretien/réparation dûment remplie** (voir pochette en fin de manuel).

Si une batterie encore sous garantie est défectueuse, il faut nous retourner:

1. la batterie défectueuse;
2. **une carte d'entretien/réparation dûment remplie** (voir pochette en fin de manuel).

Retournez-nous le matériel convenablement emballé à l'adresse figurant sur cette carte. En principe, le calculateur sera renvoyé après réparation dans les cinq jours ouvrables suivant sa réception. Au cas où d'autres problèmes de maintenance se poseraient, veuillez téléphoner au bureau HP le plus proche (voir adresses au dos).

RECHARGE DE LA BATTERIE ET FONCTIONNEMENT SUR SECTEUR

Avant de brancher le chargeur sur le secteur, mettre le commutateur OFF/ON sur la position OFF. Le HP-65 peut être utilisé de nouveau une fois le chargeur raccordé au secteur, durant toute la charge. D'autre part, il peut fonctionner de manière continue sur le secteur, sans aucun risque de surcharge de la batterie.

Etant donné que vous ne pouvez utiliser le lecteur de cartes sans une batterie chargée, même si le chargeur est branché, la

batterie ne doit pas être retirée pendant le fonctionnement sur secteur. Si une batterie est totalement déchargée, elle doit être chargée pendant au moins 5 minutes avant la lecture ou l'écriture d'une carte. Votre batterie a besoin d'être rechargée si les points décimaux de l'écran s'allument pendant l'introduction de la carte, puis s'effacent.

Une batterie vide est complètement rechargée en 14 heures. Un temps de charge moins long permettra de travailler moins longtemps. Recharger, de préférence, la batterie pendant la nuit.

ATTENTION

Si vous utilisez le HP-65 sur secteur (sans batterie), vous risquez de détériorer votre calculateur.

Pour utiliser le chargeur de batterie, procéder comme suit:

1. S'assurer que le commutateur de sélection de tension du chargeur est positionné sur la tension appropriée.

ATTENTION

Pour éviter d'endommager votre HP-65, positionner le sélecteur de tension du chargeur en fonction de la tension secteur disponible.

2. Mettre le commutateur OFF/ON du HP-65 sur la position OFF.
3. Introduire le connecteur du chargeur dans la prise arrière du HP-65, puis brancher le chargeur sur le secteur. Quand le HP-65 est relié à son chargeur, le fonctionnement n'est possible que si le chargeur lui-même est sous tension.
4. Mettre le commutateur sur la position ON. Si le commutateur W/PRGM-RUN est sur la position RUN, l'écran doit afficher 0.00.
5. Remettre le commutateur sur OFF si on ne compte pas utiliser le HP-65 pendant la charge.

6. Une fois la charge terminée, on peut soit continuer à utiliser le HP-65 sur le secteur, soit l'utiliser sur la batterie (voir l'alinéa 7 suivant).
7. Le commutateur étant sur OFF, débrancher le chargeur du secteur, puis séparer le chargeur du calculateur.

ATTENTION

Vous risquez de détériorer votre calculateur si vous utilisez un chargeur autre que le chargeur HP-82010A fourni avec votre calculateur.

FONCTIONNEMENT SUR BATTERIE

Utiliser exclusivement la batterie rechargeable HP-82001A, contrôlée et garantie un an. La batterie assure environ trois heures de fonctionnement continu. En coupant le contact lorsque le calculateur n'est pas en service, la réserve de la batterie du HP-65 est très suffisante pour une journée de travail normal.

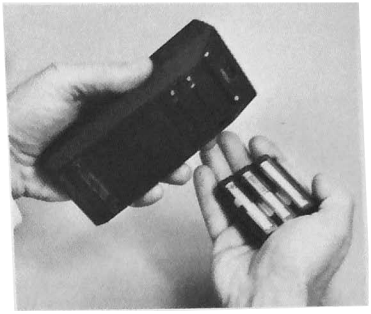
Quand la batterie ne permet plus que 2 à 5 minutes de travail, tous les points s'allument sur l'écran. Même dans ce cas, la position réelle de la virgule reste connue car un emplacement entier lui est réservé.

CHANGEMENT DE BATTERIE

Pour changer la batterie, procéder comme suit:

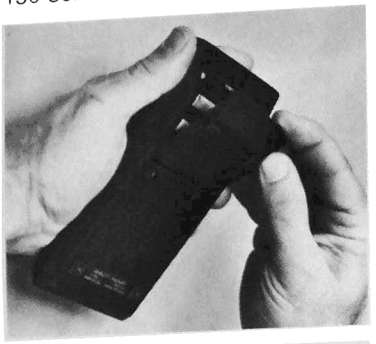
1. Mettre le commutateur OFF/ON sur la position OFF, puis débrancher le chargeur.
2. Faire glisser les deux verrous du couvercle du compartiment de la batterie.

3. Enlever alors le couvercle, puis la batterie.

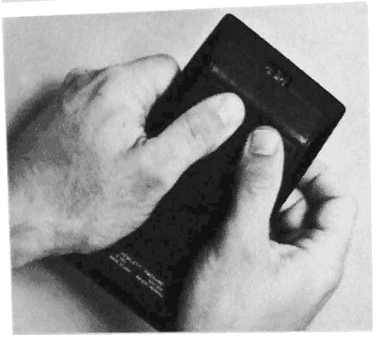


4. Vérifier que les ressorts du connecteur de batterie n'ont pas été aplatis accidentellement. Si nécessaire, les redresser, puis essayer à nouveau la batterie.

5. Introduire la batterie avec ses plots côté calculateur et en contact avec les ressorts du calculateur.



6. Refermer le couvercle.



7. Verrouiller le couvercle du compartiment batterie en appuyant doucement et en faisant glisser les deux verrous.

REMARQUE

En cas d'utilisation intensive de votre HP-65 à l'extérieur ou en voyage, il est préférable de commander l'ensemble HP-82004A (porte-batterie + batterie). Il vous permettra de recharger une batterie tout en travaillant avec l'autre.

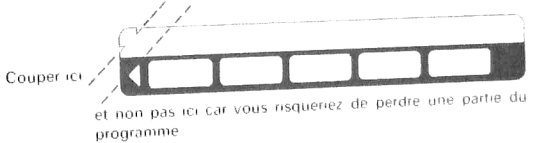
Une dégradation temporaire, particulière aux batteries cadmium-nickel, peut amener une réduction du temps de fonctionnement sur batterie. Dans ce cas, laisser le commutateur sur la position ON pendant au moins cinq heures pour décharger totalement la batterie. Recharger ensuite pendant 14 heures : cette dégradation temporaire doit alors disparaître.

Une batterie qui ne «tient pas la charge» peut être défectueuse. Si la garantie est encore valable, renvoyer cette batterie à Hewlett-Packard conformément aux instructions d'expédition (voir Annexe C). Dans le cas contraire, utiliser la carte de commande d'accessoires jointe au HP-65.

CARTES MAGNÉTIQUES

PROTECTION

Pour protéger une carte contenant un programme enregistré, couper au moyen de ciseaux le coin entaillé, comme indiqué ci-dessous :



ENTRETIEN

Conservez vos cartes magnétiques dans un bon état de propreté et exemptes d'huile, de graisse et de saletés. Une carte sale ne peut que dégrader les performances de votre lecteur de cartes. Les cartes peuvent être nettoyées avec de l'alcool au moyen d'un chiffon doux humide.

Evitez d'exposer votre calculateur aux poussières et laissez-le de préférence dans son étui souple lorsque vous ne l'utilisez pas. Chaque étui à cartes comporte une carte de nettoyage des têtes.

La tête d'enregistrement magnétique est similaire à celle de tout matériel d'enregistrement magnétique. Il en résulte que toute accumulation de poussières ou de corps étrangers sur la tête peut empêcher le contact entre la tête et la carte et produire des incidents de lecture ou d'écriture. La carte de nettoyage des têtes comporte une sous-couche abrasive ayant pour objet d'enlever tout corps étranger. Toutefois, si l'on utilise cette carte alors qu'il n'y a pas de corps étranger, on risque d'user la tête. L'utilisation répétée de la carte de nettoyage risque donc de réduire la durée de vie de votre lecteur de cartes. Mais si vous pensez que la tête est sale ou si vous avez des incidents de lecture ou d'écriture de programme, n'hésitez pas à l'utiliser. Toutefois, si un à cinq passages de la carte de nettoyage ne permettent pas d'éliminer l'incident, retournez-nous votre calculateur.

ANNOTATION

On peut faire des annotations sur le côté non magnétique de la carte avec un instrument ne risquant pas de démagnétiser la carte. En général, on écrit le nom du programme à la partie supérieure et les symboles identifiant les fonctions des touches de la rangée supérieure dans les cases se trouvant en dessous. L'utilisation d'une machine à écrire pour porter des annotations sur une carte magnétique risque d'avoir un effet sur les caractéristiques lecture/écriture des cartes.

UTILISATION DE LA SECONDE PISTE

On peut enregistrer un programme sur la deuxième piste d'une carte (et le lire ultérieurement) en introduisant la carte dans le sens inverse de la flèche, face en haut : enregistrement de deux programmes de 100 pas sur une même carte. Toutefois, nous vous conseillons de n'utiliser qu'une seule piste pour les raisons suivantes :

1. Il n'est pas facile d'indiquer les références du second programme.
2. La protection du second programme est extrêmement délicate (ne pas couper au-delà du trait sur la première piste, sinon vous risquez de perdre des informations).
3. Le galet du moteur passe sur la seconde piste et à la longue la lecture risque de ne pas s'effectuer correctement.

FONCTIONNEMENT INCORRECT DU LECTEUR DE CARTES

Si votre calculateur semble fonctionner correctement, mais que la lecture ou l'écriture de cartes programmes donne lieu à des incidents, procéder comme suit :

1. S'assurer que le commutateur W/PRGM-RUN est dans la bonne position compte tenu de l'opération envisagée : position RUN pour la lecture, position W/PRGM pour l'enregistrement.

2. Si le moteur d'entraînement ne démarre pas lors de l'insertion de la carte, s'assurer que le contact est normal au niveau de la batterie et que celle-ci est bien chargée. Ne pas oublier que le chargeur peut à lui seul fournir assez de courant pour l'alimentation du moteur d'entraînement. Pour permettre le fonctionnement du moteur du lecteur de cartes, le chargeur doit être utilisé conjointement avec une batterie partiellement chargée. Si la batterie est complètement déchargée, brancher le chargeur et attendre 5 minutes avant d'utiliser le lecteur de cartes.

3. Si le mécanisme d'entraînement de la carte fonctionne correctement, mais qu'il n'est toujours pas possible de lire ou d'écrire des cartes programmes, il se peut que les têtes d'enregistrement/lecture soient sales. Utiliser alors la carte de nettoyage des têtes. Tester ensuite le calculateur en utilisant les deux cartes de diagnostic fournies dans la bibliothèque «Programmes types». Si l'incident persiste, retourner votre calculateur à Hewlett-Packard.

ATTENTION

Les cartes magnétiques peuvent être effacées accidentellement si elles sont soumises à des champs magnétiques intenses (les magnétomètres des aéroports ne présentent pas de danger).

ANNEXE D

ERREURS COURANTES

1. Labels (lettres A à E) répétés dans la mémoire programme, l'utilisateur ayant enregistré de nouvelles fonctions (en mode W/PRGM) sans avoir appuyé au préalable sur les touches f PRGM.
2. Effacement par erreur d'un programme en mémoire suite à l'introduction d'une carte magnétique, le commutateur W/PRGM-RUN étant sur la position RUN.
3. Effacement par erreur d'un programme sur carte magnétique par suite de l'insertion d'une carte magnétique non protégée, le commutateur W/PRGM-RUN étant sur la position W/PRGM.
4. Introduction d'instructions indésirables dans la mémoire programme (l'utilisateur appuie sur des touches alors que le commutateur est encore en position W/PRGM).
5. Non-exécution d'une fonction jaune (f ou f^{-1}) ou d'une fonction bleue (g) par suite de l'omission d'une touche de préfixation.
6. Perte du contenu du registre **T**, l'introduction d'un nouveau nombre ou le rappel d'un registre entraînant la montée de la pile opérationnelle.
7. Perte du contenu du registre R_9 par suite de l'exécution d'une fonction trigonométrique, d'un test de comparaison ou d'une conversion de coordonnées rectangulaire/polaire.
8. Remplissage incomplet des deux pas sautés lors d'un test (codes combinés).
9. Exécution d'une fonction trigonométrique dans une unité angulaire incorrecte.

10. Tentative d'appel par un label numérique (0 à 9) d'une fonction à définir alors qu'elle doit être appelée par une lettre (**A** à **E**).

11. Non-annulation des flags avant leur utilisation.

12. Registre [LST x], pile opérationnelle ou registres mémoire considérés comme inchangés après appel d'une fonction définie par l'utilisateur au moyen du clavier (lettres **A**, ..., **E**) ou dans un programme.

13. Utilisation de [DSZ] dans un programme et non-initialisation du registre R_8 à la valeur appropriée.

14. Calcul de f(**x**, **y**) avec inversion des opérandes.

15. Perte de programmes ou de données due à l'une des fausses manœuvres suivantes: coupure de l'alimentation du calculateur, débranchement du chargeur ou non-branchement du chargeur.

16. Absence de retour d'un sous-programme après une tentative d'emboîtement. Le programme principal peut appeler un sous-programme (fonction à définir), mais ce sous-programme doit revenir au programme principal avant qu'un autre sous-programme puisse être appelé.

17. Utilisation de [CLx] pour introduire 0 dans le registre **X**, [0] remplacé lors de l'opération suivante parce que [CLx] annule la montée de la pile. Utiliser la touche [0] pour mettre 0 dans le registre **X**.

INDEX

− + × ÷ opérations arithmétiques **6, 8, 30**
0 . . . 9 digits
  adresse (programme) **69**
  introduction des données **6, 22**
  modes d'affichage **18, 21**
  sélection des registres mémoire **27, 29**
• point décimal **6**
π constante π **33**
10^x antilogarithme en base 10 **45**

A

A B C D E touches de la rangée supérieure **52**
ABS valeur absolue **43**
Accessoires **128**
Addition
  degrés, minutes, secondes **38**
  pile opérationnelle **8**
  registre mémoire **30**
  touche + **6**
Adresse **52**
Affichage
  absence **130**
  batterie en fin de charge (...) **20**
  clignotant **20**
  code des touches **60**
  scientifique et fixe **19**
Antilogarithme
  en base 10 (10^x) **45**
  naturel (e^x) **45**
Arc cosinus **38, 40**
Arc sinus **38, 40**
Arc tangente **38, 41**
Arithmétique au niveau des registres **30**

B

Batterie
changement **134**
fonctionnement **134**
recharge **132**
Boucle **59**
Branchement
inconditionnel **68**
conditionnel **74**

C

Calcul sans programme **5**
Calculs en chaîne **9**
Carré **45, 47**
Carte magnétique
écriture **15**
entretien **137**
lecture **11**
protection **137**
CHS changement de signe **6**
Clavier **5**
CLX effacement du registre **X** **17**
Code des touches
combiné **61**
simple **60**
Commutateur
OFF/ON **5**
W/PRGM – RUN **11, 15, 53**
Compteur programmé **86**
COS cosinus, arc cosinus **38**
Conversions
angle décimal ⟷ degré, minute, seconde **39**
coordonnées cartésiennes ⟷ polaires **41**
décimal ⟷ octal **42**
Cosinus **38**

D

Décisions logiques **82**
[DEG] (voir mode d'angle) **37**
Degrés, minutes, secondes
  addition/soustraction **38, 39**
  conversion **38, 39**
[DEL] effacement d'un pas de programme **66**
Dépassement de capacité **126**
Digits (voir [0] ... [9] en début d'index)
Division
  touche ÷ **6**
  pile opérationnelle **8**
  registre mémoire **30**
[D.MS+] addition/soustraction de degrés, minutes, secondes **38**
[→D.MS] angle décimal degrés, minutes, secondes **38**
Données
  conversion (voir conversion)
  affichage **19**
  introduction **6**
[DSP] affichage **19**
[DSZ] compteur programmé **86**

E

e^x antilogarithme naturel **45**
[EEX] introduction de l'exposant **22**
Effacement
  calculateur **18**
  flags **91**
  mémoire programme **18**
  pas de programme **59**
  pile opérationnelle **17**
  préfixe **17**
  registres mémoire **18**
  registre last X **24**
[ENTER↑] copie x en y **6**
Entier en octal **42**

Erreurs
courantes **141**
opérations illicites **20**
Exponentielle (y^x) **45**

F

f f-1 préfixes jaunes **7**
Factorielle (n!) **45**
Flag
positionnement **91**
test **93**
Fonction à définir par l'utilisateur **52**
Fonctionnement sur secteur **132**
Fonctions inverses
10^x (antilogarithme en base 10) **45**
arc cosinus **38**
arc sinus **38**
arc tangente **38**
carré **45**
coordonnées polaires en cartésiennes **38**
degrés, minutes, secondes en degrés décimaux **39**
e^x (logarithme népérien) **45**
octal en entier **43**
soustraction (D.MS) **38**
Fonctions logarithmiques
logarithme décimal (base 10) **45**
logarithme népérien (e^x) **45**
Fonctions trigonométriques **40**
cartésiennes/polaires ou inversement **41**
cosinus/arc cosinus **40**
sinus/arc sinus **40**
tangente/arc tangente **41**

G

g préfixes bleus **7**
Garantie **131**
GRD voir mode d'angle **37**
GTO GO TO **74**

I

[INT] partie entière ou décimale d'un nombre **43**
Instructions d'expédition **132**
Introduction
  données **6**
  nombres négatifs **6**
  exposant **22**

L

LAST X **23**
[LBL] adresse **52**
Lecteur de cartes **139**
Lettres (A à E) **52**
Limites opérationnelles **125**
[LN] logarithme/antilogarithme népérien **45**
[LOG] logarithme/antilogarithme décimal **45**
Logarithme
  décimal (base 10) **45**
  népérien (e^x) **45**
[LST x] registre Last X **23**

M

Maintenance **130**
Mémoire programme **58**
Mise en mémoire (registre R_n) **27**
Mise en mémoire d'un programme **11**
Mode d'angle
  degré **37**
  grade **37**
  radian **37**
Mode RUN **53**
Mode W/PRGM (programme) **53**
Mouvements de la pile opérationnelle **25**
Multiplication
  pile opérationnelle **8**
  registres mémoire **30**
  touche [×] **6**

N

[n!] factorielle **45**
Nettoyage des têtes **137**
Nombre négatif **6**
[NOP] pas d'opération **63**
Notation scientifique (introduction des données) **19**

O

[→OCT] décimal ⟷ octal **43**
OFF-ON commutateur de mise en service **5**
Opérations arithmétiques
  pile **9**
  registre mémoire **30**
Opérations de préfixation **7**
Opérations illicites **20**

P

Partie décimale d'un nombre **43**
Partie entière d'un nombre **43**
Pas de programme **59**
Pas d'opération (NOP) **63**
Permutation de la pile vers le bas **31**
Permutation de la pile vers le haut **31**
Permutation de x et y **31**
Pi (π) **33**
Pile opérationnelle
  arithmétique **8**
  effacement **17**
  mouvement **25**
Point décimal
  affichage **19**
  introduction **6**
Pointeur **59**
Précision **125**
[PREFIX] effacement des préfixes **17**
Préfixe **7**

PRGM effacement de la mémoire programme 18
Procédure de contrôle du HP-65 130
Programmation
adresse 52
affichage de la mémoire-programme pas à pas 66
boucle 59, 63
branchement conditionnel 74
branchement inconditionnel 68
codage des touches 60
compteur programmé 86
décisions logiques 82
définitions 121
dépassement de capacité 121
effacement d'un pas de programme 66
enregistrement d'un programme sur carte magnétique 55
exemples simples 50
exemples complets 100
flag 91
mémoire programme 58
mise au point d'un programme 120
mise en mémoire d'un programme 53
mode W/PRGM 53
mode RUN 53
organigramme 105
pas de programme 59
pointeur 59
quelques «trucs» pour améliorer vos programmes 114
rédaction d'un programme 12
saut de 2 pas de programme 93
sous-programme 74
tests de comparaison 53

R

R↑ permutation de la pile vers le haut 31
R↓ permutation de la pile vers le bas 31
RAD voir mode d'angle 37
Rappel du registre mémoire Rn 27
RCL rappel 28

Rédaction d'un programme **12**
REG effacement des registres mémoire
Registre Last X **23**
Registre mémoire **10, 27**
  arithmétique directe entre registres et pile **30**
  registre R8 (décrémentation) **30**
  registre R9 **30**
  sélection **29**
  stockage et restitution des données **27**
Registre pile opérationnelle (X, Y, Z, T) **7**
R→P conversion des coordonnées polaires ⟷ rectangulaires **38**
R/S run/stop **52**
RTN return **52**
Return **52**
RUN (commutateur W/PRGM-RUN) **53**
run/stop **52**

S

Saut de 2 pas de programme **93**
Service **130**
SF1 SF2 (voir flag)
Signe
  exposant **23**
  nombre **6**
SIN sinus/arc sinus **38**
Single step **66**
Sinus **38**
Sous-programme **74**
Soustraction
  degrés, minutes, secondes **38**
  pile opérationnelle **8**
  registres mémoire **30**
  touche – **6**
SST single step **66**
Stack (voir pile opérationnelle)
STK pile opérationnelle **17**
STO mise en mémoire **27**
Suppression d'un pas de programme **66**

T

[TAN] tangente/arc tangente 38
Tangente 38
Température de fonctionnement 127
Tests
  comparaison entre x et y ([x≠y], [x≤y], [x=y], [x>y]) 83
  flags 91
[TF1] [TF2] (voir flag)
Touches d'introduction de données 6
Touches de préfixation 7, 17

U

Utilisation d'un programme préenregistré 10

V

Valeur absolue 43

W

W/PRGM-RUN (commutateur) 11, 15, 53

X

[1/x] inverse 45
[√x] racine carrée/carré 45
[x⇄y] permutation de x et y 31
[x≠y], [x≤y], [x=y], [x>y] tests de comparaison entre x et y 83

Y

[y^x] exponentielle 45

172 points de vente dans 65 pays assurent le service après vente

Hewlett-Packard France:
Siège social: Quartier de Courtabœuf, boîte postale nº 6, 91401 Orsay, tél. (1) 907 78 25
Agence de Lyon: Chemin des Mouilles, boîte postale nº 12, 69130 Ecully, tél. (78) 83 65 25
Agence de Rennes: 63, avenue de Rochester, 35000 Rennes, tél. (99) 36 33 21
Agence de Strasbourg: 74, allée de la Robertsau, 67000 Strasbourg, tél. (88) 35 23 20/21
Agence de Toulouse: Zone Aéronautique, avenue Clément-Ader, 31770 Colomiers, tél. (61) 78 11 55
Agence de Marseille: Aéroport principal de Marseille-Marignane, 13721 Marignane, tél. (91) 89 90 10, postes 208/214
Pour la Belgique: Hewlett-Packard Benelux S.A., 1, avenue du Col-Vert, B-1170 Bruxelles, tél. (02/03) 672 22 40
Pour la Suisse romande: Hewlett-Packard (Schweiz) AG, 9, chemin Louis-Pictet, 1214 Vernier-Genève, tél. (022) 41 49 57
Pour les pays du bassin méditerranéen, Afrique du Nord et Moyen-Orient: 35, Kolokotroni Street – Platia Kefallariou, Kifissia-Athènes, Grèce, télex 21 6588, câble Hewpacksa Athènes, tél. 80 80 337/359/429 et 8018 693
Pour le Canada: Hewlett-Packard Canada, 275 Hymus Boulevard, Pointe-Claire, 730 Québec, tél. (518) 561 6520
Direction pour l'Europe: Hewlett-Packard S.A., 7, rue du Bois-du-Lan, 1217 Meyrin 1 – Genève, Suisse, tél. (022) 41 54 00

00065-90219 French

Printed in Switzerland