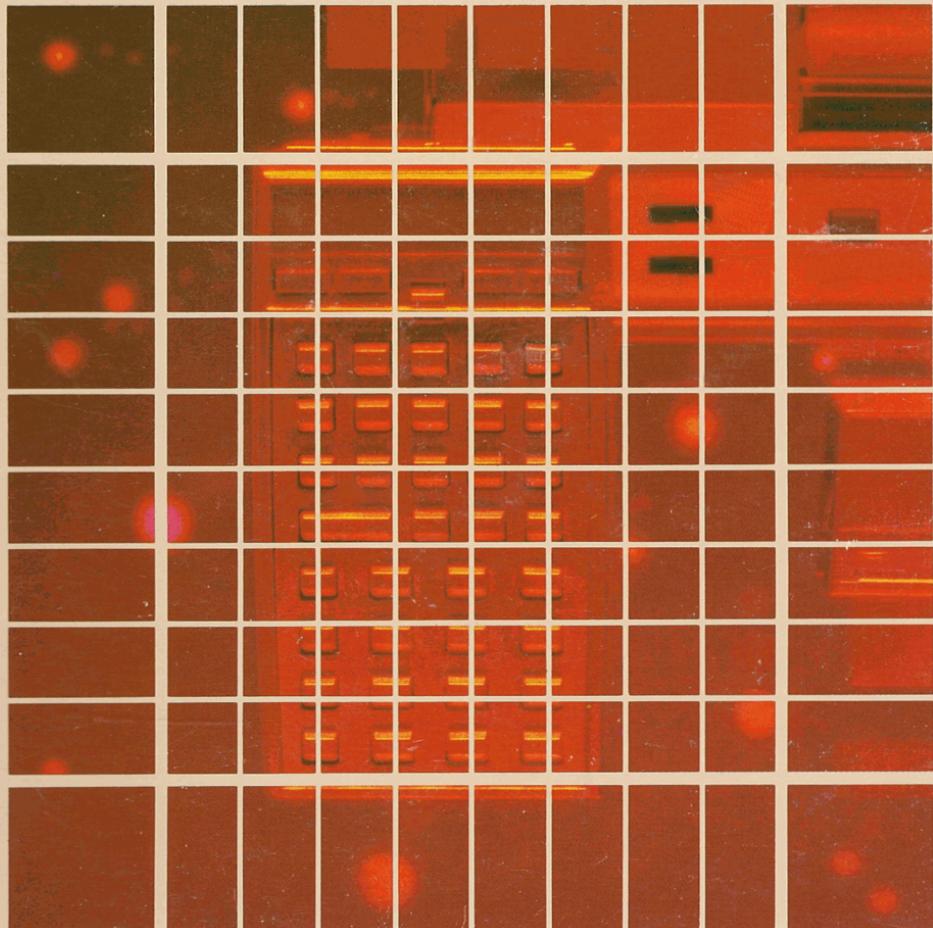


HEWLETT-PACKARD

ADVANTAGE PAC pour le HP-41

MODULE DE SOLUTIONS DE HAUT NIVEAU



CONVENTIONS TYPOGRAPHIQUES DANS LE PRESENT MANUEL

Exemple de notation	Description
$\Sigma+$ $\Sigma-$	Fonction primaire du clavier. Appuyez sur la touche $\Sigma+$. Fonction secondaire du clavier. Appuyez sur la touche jaune ■ , relâchez-la, puis appuyez sur la touche $\Sigma-$.
A (FX)	Fonction propre à un programme particulier. Appuyez sur $\Sigma+$. (Correspond à la touche portant la lettre «A» en bleu). FX est le libellé affiché pour A dans cet exemple.
XEQ TVM	Fonction non présente sur le clavier. Pour l'exécuter, appuyez sur XEQ ALPHA TVM ALPHA . Vous pouvez aussi affecter cette fonction à une touche du clavier personnel, puis exécuter la fonction en validant le clavier personnel (une pression sur la touche ALPHA) et en appuyant sur la touche).
SIZE 013 ABC	SIZE ALPHA SIZE ALPHA 0 1 3 Caractères du clavier alphabétique correspondants aux lettres bleues figurant sur les touches. Appuyez sur ALPHA avant de commencer à frapper les caractères et après avoir fini.
123	Caractères secondaires du clavier alphabétique (pour y accéder, appuyez d'abord sur la touche préfixe jaune ■). Ces caractères sont représentés au dos du calculateur.

NOTE

AFFICHAGE EUROPEEN

Le module affiche normalement les données et résultats en format américain (virgule pour séparer les groupes de trois chiffres, et point décimal). Pour obtenir le format d'affichage européen utilisé dans ce manuel (virgule décimale et point pour séparer les groupes de trois chiffres), il faut désarmer l'indicateur binaire 28 et armer l'indicateur binaire 29 :

CF 28 (virgule décimale)

SF 29 (groupes de 3 chiffres séparés par des points)



Advantage PAC pour le HP-41

Module de solutions de haut niveau

Août 1985

00041-90561

REMERCIEMENTS

Les opérations sur les matrices contenues dans le présent module sont basées sur la ROM CCD, écrite par la société W & W Software Products GmbH, 5060 Bergisch Gladbach 2, République fédérale d'Allemagne.

Les sous-programmes de résolution d'une équation et d'intégration numérique sont une adaptation par Firmware Specialists, Inc., 605 NW 5th Street #2A, Corvallis, Oregon 97330, Etats-Unis, de ceux qui équipent le calculateur HP-15C.

Le concept original du contenu et de l'interface utilisateur de ce module a été mis au point par Chris Bansen, Corvallis, Oregon, Etats-Unis.

L'annexe A, «Les taux d'intérêts», a été écrite par Marc Girard, Centre de formation de la profession bancaire, 7 rue du général Foy, 75008 Paris.

AVERTISSEMENT

1. Les informations contenues dans ce document peuvent faire l'objet de modifications sans préavis.
2. En raison de la complexité des techniques informatiques, ce document est remis au lecteur dans le seul but de faciliter sa compréhension du produit dont il traite. HPF décline en conséquence toute responsabilité pour tout dommage pouvant résulter des informations contenues dans ce document.
3. HPF ne garantit ni la fiabilité ni les conséquences de l'utilisation de ses produits logiciels lorsqu'ils sont utilisés sur des produits dont il n'a pas assuré la fourniture.
4. Les informations contenues dans ce document sont originales. Elles ont été conçues et mises au point par Hewlett-Packard. L'acheteur s'interdit en conséquence, sauf accord préalable et écrit de HPF :
 - de les divulguer ou d'en faciliter la divulgation ;
 - de les copier ou de les reproduire en tout ou en partie par n'importe quel moyen et sous n'importe quelle forme ;
 - de les traduire dans toute autre langue.

TABLE DES MATIERES

Introduction	5
Insérer et retirer des modules d'application	7
Comment utiliser ce manuel et le module	11
Le programme matriciel	19
Indique une méthode aisée pour convertir des matrices réelles ou complexes. Les opérations comprennent inversion, transposition, calcul du déterminant et résolution d'équations simultanées.	
Fonctions matricielles	30
Plus de quarante-cinq fonctions pour stocker et manipuler des matrices et des éléments matriciels. Comprend toutes les possibilités du programme MATRX, l'arithmétique matricielle et la recherche d'éléments spécifiques.	
Résolution d'une équation	61
Calcule les racines réelles d'une équation $f(x) = 0$.	
Résolution d'une équation polynomiale	71
Calcule les racines complexes et réelles d'un polynôme à coefficients réels de degré inférieur ou égal à 5.	
Intégration numérique	79
Calcule l'intégrale définie sur l'intervalle donné d'une équation $f(x) = 0$.	
Équations différentielles	87
Résout les équations différentielles du premier et second ordre.	
Opérations sur les nombres complexes	93
Arithmétique et autres opérations courantes sur les nombres complexes, y compris trigonométrie, valeur absolue, inverse, logarithmes en base 10 et népérien.	
Opérations vectorielles	101
Effectue les opérations courantes sur les vecteurs à 2 ou 3 dimensions : addition, soustraction, produit scalaire, produit vectoriel, distance (norme de la différence), angle, norme, et vecteur unitaire.	
Transformations de coordonnées	117
Transforme des coordonnées en trois dimensions, avec ou sans rotation.	
Changements de base numérique et logique booléenne	127
Effectue des conversions entre nombres binaires, octaux, décaux et hexadécimaux. Comprend en outre des fonctions de logique booléenne, et de vérification et de rotation de bits.	

4 Table des matières

Ajustement de courbes	133
Collecte un ensemble d'observations statistiques et ajuste à ces données une droite, une courbe logarithmique, une courbe exponentielle, une courbe de puissance, ou celle de ces quatre courbes qui s'ajuste le mieux aux données.	
Valeur de l'argent dans le temps	143
Résout des problèmes financiers impliquant le temps, l'argent, et les taux d'intérêt.	
Annexe A : Les taux d'intérêt	153
Index des programmes	intérieur de couverture, en fin de manuel

INTRODUCTION

Le module Advantage du HP-41 vous offre un choix de programmes et de fonctions vous permettant de résoudre des problèmes mathématiques et d'ingénierie de haut niveau, de réaliser des ajustements de courbes pour des données statistiques et de résoudre des problèmes financiers simples (valeur de l'argent dans le temps —TVM). Il constitue ainsi un ensemble puissant et varié de solutions, à la disposition des étudiants et professionnels scientifiques et techniques.

Un grand nombre des sous-programmes inclus dans ce module ont été rendus accessibles à l'utilisateur, de façon qu'il puisse les utiliser dans ses propres programmes.

Ce manuel décrit chaque programme ou jeu de fonctions en l'accompagnant des équations correspondantes, d'instructions d'utilisation détaillées, d'exemples montrant les séquences de touches requises pour parvenir à une solution et d'une description des sous-programmes accessibles à l'utilisateur.

Remarque : Avant d'enficher le module Advantage dans votre HP-41, éteignez le *calculateur* et respectez scrupuleusement les instructions du paragraphe «Insérer et retirer des modules d'application».

INSERER ET RETIRER DES MODULES D'APPLICATION

Avant d'insérer un module d'application dans votre calculateur, lisez ces quelques informations.

Quatre modules d'application peuvent être insérés ensemble dans les ports d'un HP-41. Les noms des programmes contenus dans ces modules sont listés dans le catalogue 2 (**CATALOG** 2).

AVERTISSEMENT

Avant d'insérer ou de retirer un module, *mettez le calculateur hors tension*. Le calculateur et le module pourraient être endommagés si vous omettez cette précaution.

Avant d'insérer un module :

Eteignez votre calculateur !



Retirez le capuchon protégeant le port choisi ; conservez-le pour le protéger lorsqu'il sera vide à nouveau.



Dans un HP-41C, insérez le module d'application dans n'importe quel port *suivant* immédiatement le dernier module inséré. Insérez le module de façon que l'étiquette soit lisible lorsque le calculateur est placé clavier vers le haut. S'il y a déjà un module mémoire dans le port 1, vous pouvez insérer un module d'application en 2, 3 ou 4 ; les ports sont numérotés sur un diagramme figurant sur le



8 Insérer et retirer des modules d'application

dos du calculateur. Mais n'insérez jamais un module d'application dans un port de nombre inférieur à celui d'un module mémoire.

Insérez n'importe quels modules d'application supplémentaires, en les plaçant également après le dernier module mémoire. Protégez les ports inutilisés.

Les modules d'application sont prêts à fonctionner.

Avant de retirer un module :

Eteignez le calculateur ! Si vous ne le faites pas, vous risquez d'endommager le calculateur et le module.

Saisissez le tenon du module et retirez-le comme indiqué ici.



Protégez le port vide par un capuchon.

Il en est des autres accessoires enfichables (tels le lecteur de cartes HP 82104A ou le lecteur optique HP 82153A) comme des modules d'application.

Des vides peuvent être laissés entre modules et accessoires : il est possible d'enficher un module mémoire dans le port 1 et un module d'application dans le port 4, les ports 2 et 3 étant laissés vides.

COMMENT UTILISER CE MANUEL ET LE MODULE

Que contiennent les chapitres ?

Chacun des chapitres de ce manuel couvre un programme ou un ensemble de fonctions différent. A l'exception des deux premiers chapitres sur les matrices, chacun des chapitres est indépendant des autres.

Au début de chaque chapitre vous trouverez une **description** du programme ou des fonctions auquel il est consacré. Les **équations** sur lesquelles le programme se fonde sont données et des **références** sont éventuellement données pour la recherche d'autres informations. Lorsque c'est nécessaire, les valeurs limites des données sont indiquées. Dans certains cas, le programme pourra outrepasser ses limites de validité, mais le résultat pourrait alors manquer de la précision nécessaire.

Tableau d'instructions d'utilisation

Au centre de chaque chapitre se trouvent les instructions (**Utilisation**) et le tableau **instructions d'utilisation**. Les instructions décrivent pas à pas l'utilisation du programme ou des fonctions. Elles indiquent quelles sortes de valeurs il faut saisir et quelles touches utiliser pour calculer les résultats.

Au dessus du tableau figure le nombre de registres nécessaires pour exécuter le programme. Lisez le paragraphe «Affecter des registres de mémoire», en page suivante.

Instructions	Frappez :	Affichage	Taille : 016
<p>3. Entrez vos couples de données : Répétez pour chaque couple</p> <p>⋮</p>	<p>y ENTER x A $(\Sigma+)$</p>	<p>$\Sigma+$ CLΣ FIT</p>	

Diagramme d'explication :

- Une flèche pointe vers la colonne "Instructions" (la première colonne) depuis la légende "Cette colonne décrit ce que vous avez à faire, y compris la sorte de saisie que vous avez à faire (valeurs des données)."
- Une flèche pointe vers la colonne "Frappez :" (la deuxième colonne) depuis la légende "Cette colonne indique les touches à actionner pour entrer votre saisie ou calculer un résultat."
- Une flèche pointe vers la colonne "Affichage" (la troisième colonne) depuis la légende "Celle-ci indique ce qui apparaîtra sur l'affichage du calculateur si vous suivez les instructions. Souvent, il s'agira d'un résultat, d'une sollicitation à fournir des données, ou d'un menu. Le *menu*, l'interface vers l'utilisateur utilisée par bien des programmes de ce module, est décrit en page suivante, sous «Le menu : l'interface-utilisateur»."

Après ce tableau viennent des **remarques** qui concernent l'un ou l'autre détail de fonctionnement et des explications ; chaque chapitre est illustré d'**exemples** d'utilisation.

Enfin, des **informations pour la programmation** sont disponibles pour accéder aux sous-programmes chaque programme, en vue de les utiliser dans un programme que le lecteur pourrait désirer créer.

Affecter des registres (instruction **SIZE**)

Le tableau des instructions indique le nombre minimum de registres de stockage requis pour l'exécution d'un programme donné. Pour allouer ces *nnn* registres de stockage, utilisez la fonction **SIZE** (appuyez sur **XEQ** **ALPHA** **SIZE** **ALPHA** *nnn*). Vous trouverez dans le Manuel d'utilisation du HP-41 des informations sur cette fonction. Si vous essayez d'exécuter un programme, mais obtenez en retour le message

SIZE $>= nnn$

il vous faut régler la *taille* (**SIZE**) de la mémoire au minimum à *nnn*. Appuyez ensuite sur **R/S** pour poursuivre l'exécution du programme.

Les touches du calculateur : notation

On distingue dans le Manuel d'utilisation du HP-41, les *fonctions du clavier* et les *fonctions non présentes sur le clavier*. Ces deux types de fonctions sont invoquées (exécutées) de deux manières différentes. Les fonctions du clavier ont leurs propres touches sur le clavier (telle **TAN** et **x^2**). Pour exécuter les *fonctions non présentes sur le clavier*, il suffit de frapper leur nom après avoir appuyé sur **[XEQ]**.*

Exemple de notation	Touches à utiliser
[$\Sigma+$]	[$\Sigma+$] Ceci est une fonction du clavier.
[$\Sigma-$]	[$\Sigma+$] (Appuyez séquentiellement sur ces touches et non simultanément). Ceci est une fonction secondaire du clavier.
[A] (FX)	[$\Sigma+$] ($\Sigma+$ est imprimé sur le dessus de la touche ; A, sur sa face inclinée). Il s'agit ici d'une fonction «sur mesure» liée à un programme précis. FX est ce qui apparaît sur l'écran au-dessus de [A] . FX est le <i>libellé de menu</i> correspondant à [A] .
[XEQ] [TVM]	[XEQ] [ALPHA] [TVM] [ALPHA] (La touche [ALPHA] , touche «en bascule», place le calculateur en mode Alpha). Ceci est une fonction <i>non</i> présente au clavier. Elle peut aussi être exécutée en mode «Personnel» (User). Relisez le <i>Manuel d'utilisation</i> du HP-41.
[XEQ] [SIZE] 013	[XEQ] [ALPHA] [SIZE] [ALPHA] [0] [1] [3]

Ce module utilise les touches des deux premiers rangs et leur donne des fonctions spéciales, redéfinies. Elles sont représentées en tant que touches de **[A]** à **[J]**, non comme **[$\Sigma+$]** etc.

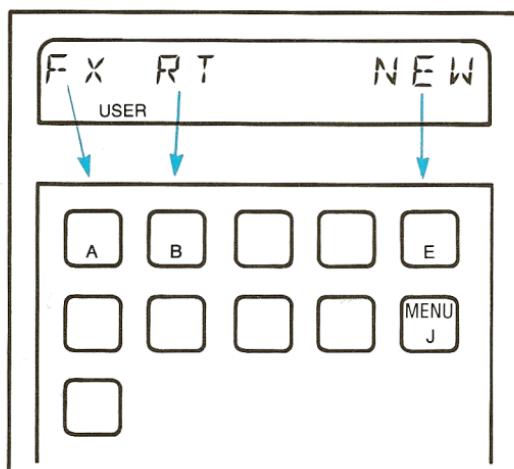
Le menu : l'interface-utilisateur

Ce module fournit à la fois des *fonctions* et des *programmes*. Les fonctions se comportent comme d'autres fonctions-non-présentes-au-clavier du HP-41. Les programmes sont plus sophistiqués - et aussi plus faciles à utiliser. Ils combinent plusieurs nouvelles fonctions et une *interface vers l'utilisateur*, comportant messages de sollicitation et menus.

* L'autre solution pour exécuter une fonction non présente au clavier - plus rapide - est d'affecter son nom à une des touches du clavier Personnel (User). Consultez le *Manuel d'utilisation* du HP-41 (Clavier Personnel, Redéfinition du clavier Personnel).

Il est important de noter que certains programmes redéfinissent certaines touches des deux premiers rangs pour leur permettre d'effectuer certaines opérations des programmes. *Pour qu'un programme fonctionne comme prévu, il faut que vous ayez effacé toutes les redéfinitions de touches existant sur ces deux rangées.* Pour utiliser ces touches redéfinies, le clavier Personnel (mode User) doit être activé. Tous les programmes de ce module possédant cette caractéristique activent *automatiquement* le clavier Personnel lorsqu'ils sont exécutés. Si, pour une raison quelconque, vous désactivez le clavier Personnel, il vous faudra le réactiver manuellement (à l'aide de la touche **USER**) pour utiliser les touches redéfinies. Les menus identifient ces touches : en voici un exemple.

Le menu de résolution des équations polynomiales



Le libellé **FX** indique que, lorsque le programme **PLY** est exécuté, la touche supérieure gauche, **A**, est redéfinie pour évaluer le polynôme $f(x)$ à x . **B** (identifié par le libellé **RT**) est redéfini pour calculer la racine et **E** (identifié par le libellé **NEW**) initialise le programme pour lui faire accepter un nouveau polynôme.

La touche **[J]** a, avec ce module, une signification particulière. Pour tous les programmes et fonctions de ce module, un appui sur **[J]** a pour effet d'afficher le menu à l'écran.* Ceci peut être fait à tout moment et aussi souvent que nécessaire ; le menu est simplement une aide qui permet d'identifier plus clairement les touches.

Messages d'erreur. Si vous obtenez un message d'erreur sous un programme, vous pouvez appuyer sur **[J]** après avoir traité la cause de l'erreur pour ré-afficher le menu. Pour la définition des messages d'erreur, relisez le manuel d'utilisation de votre HP-41. Les messages d'erreur particuliers aux opérations sur les matrices sont décrits dans le chapitre «Fonctions de matrices».

Si le calculateur s'éteint pendant que vous travaillez sous un programme, vous trouverez l'affichage changé lorsque vous «réveillerez» votre calculateur. Il montrera le registre X, sans aucun des messages de sollicitation ou sans le menu qui se trouvait affiché avant l'extinction de l'affichage. Le programme est toujours actif, mais il est préférable de le relancer un appuyant sur **[J]** pour rappeler le menu.

Si vous exécutez un programme qui ne possède *pas* de menu, il sera probablement nécessaire de réarmer l'indicateur binaire 21 pour rétablir un affichage adéquat des résultats.

L'indicateur d'exécution

Si ce petit signe (\triangleright) vous est encore inconnu, vous ferez rapidement sa connaissance. Il apparaît sur l'écran chaque fois qu'un programme «travaille». Il vous indique, lorsque vous exécutez une opération sous un programme, qu'un calcul est en cours d'exécution.

Liste du contenu du module

Catalogue 2 vous indique tous les programmes et sous-programmes de ce module - ou de tout autre module enfiché dans votre HP-41. Frappez simplement **CATALOG** 2.

Programmes et sous-programmes

Il vous est possible d'appeler certains programmes (et sous-programmes) de ce module en tant que sous-routines de vos propres programmes chargés en mémoire du HP-41. Voyez «Informations pour la programmation» à la fin de plusieurs chapitres.

* La touche **[J]** est la même que la touche **[TAN]**. Nous utilisons la lettre de manière à ne pas confondre l'«ancienne» identité de la touche (tangente) avec la nouvelle.

Utiliser une imprimante

Si vous avez relié une imprimante HP-IL à votre HP-41 lorsque vous utilisez ce module, placez-la en mode MAN pour obtenir une impression lisible de vos saisies et des résultats. Certains programmes nécessitent le mode NORMAL. Ce mode liste toutes les valeurs saisies et toutes les frappes effectuées.

Copie de programmes

Bien des programmes de ce module peuvent être copiés en utilisant la commande **COPY**. Cependant, *il n'est pas nécessaire de copier un programme en mémoire principale pour l'utiliser*. De même, il n'est pas nécessaire de copier un sous-programme pour y accéder via l'un de vos propres programmes.

Utiliser des libellés

Dans vos programmes personnels, évitez d'utiliser des libellés identiques à ceux de ce module d'application. En cas de conflit, c'est le libellé présent en mémoire programme qui a le pas sur le libellé du module d'application. Vous trouverez une liste des libellés utilisés dans ce module sous Catalogue 2.

Conflits avec d'autres modules d'application

Remarque : Ne placez pas ensemble le module HP-41 Advantage Pac et le module de développement HP-IL (HP-IL Development Module). Ces deux modules partagent les mêmes numéros d'identification de ROM et leur utilisation conjointe posera des problèmes de fonctionnement.

Certains noms de fonctions utilisés par le module Advantage sont aussi utilisés par le module mathématiques HP-41 (Math Pac) et le programme de gestion immobilière HP-41 Real Estate Pac. Avant d'utiliser ces fonctions, retirez les modules dont les fonctions ne doivent pas être utilisées.

Redondance de fonctions

Module mathématiques

Toutes les fonctions sur les nombres complexes.

Toutes les fonctions de DIFEQ.

Module Real Estate

N, PV, PMT, FV, et *I

Où trouver de l'aide

Si vous avez des questions concernant l'utilisation de ce calculateur, référez-vous au manuel d'utilisation du calculateur HP-41. Si vous avez des problèmes techniques que ce manuel ne peut résoudre, appelez l'assistance technique à la clientèle Hewlett-Packard ou écrivez-nous. Vous trouverez adresses et numéros de téléphone à la fin de votre manuel d'utilisation.

NOTE

AFFICHAGE EUROPEEN

Le module affiche normalement les données et résultats en format américain (virgule pour séparer les groupes de trois chiffres, et point décimal). Pour obtenir le format d'affichage européen utilisé dans ce manuel (virgule décimale et point pour séparer les groupes de trois chiffres), il faut désarmer l'indicateur binaire 28 et armer l'indicateur binaire 29 :

CF 28 (virgule décimale)

SF 29 (groupes de 3 chiffres séparés par des points)

LE PROGRAMME DE CALCUL MATRICIEL

Le module Advantage offre des possibilités étendues de création, stockage et calcul sur les matrices réelles ou complexes. Ces fonctions vous sont accessibles soit en tant que *fonctions individuelles*, soit en tant que *programmes* avec menus et messages de sollicitation. C'est aussi le cas de plusieurs autres matières traitées dans ce module. A cause de la complexité et de la quantité des matières traitées, les matrices font l'objet de deux chapitres séparés.

Ces chapitres décrivent le programme sur les matrices, MATRX, d'utilisation aisée, permettant d'effectuer les opérations les plus courantes sur une matrice nouvellement créée. Pour utiliser MATRX, il n'est pas utile de savoir comment le calculateur stocke et traite les matrices dans sa mémoire. Le chapitre suivant, «Fonctions matricielles» dresse la liste - et définit - chaque fonction du module concernant les matrices, y compris celles invoquées par le programme MATRX. L'utilisation de ces fonctions de manière autonome nécessite une connaissance approfondie de la manière dont le calculateur stocke les matrices.

Possibilités du programme

Considérons le système :

$$3,8x_1 + 7,2x_2 = 16,5$$

$$1,3x_1 - 0,9x_2 = -22,1$$

pour lequel il faut déterminer les valeurs de x_1 et de x_2 . Ces équations peuvent être exprimées sous forme matricielle, $\mathbf{AX} = \mathbf{B}$, où

$$\mathbf{A} = \begin{bmatrix} 3,8 & 7,2 \\ 1,3 & -0,9 \end{bmatrix}, \quad \mathbf{X} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \quad \text{et} \quad \mathbf{B} = \begin{bmatrix} 16,5 \\ -22,1 \end{bmatrix}.$$

A est la *matrice coefficient* pour le système, **B** est la *colonne ou matrice constante* et **X** est la *solution ou matrice résultat*.

Pour un tel système, le programme MATRX crée (dimensionne) une matrice carrée réelle ou complexe, **A**, et une matrice colonne, **B**. Vous pouvez alors :

- Introduire, changer («éditer») ou simplement lire les éléments de **A** et de **B**.
- Inverser **A**.
- Transposer **A** si **A** est réelle.

20 Le programme de calcul matriciel

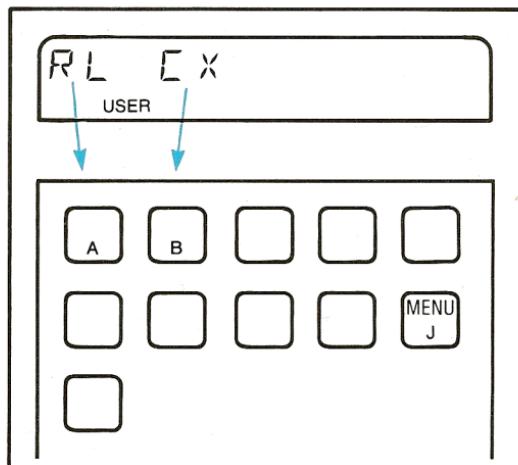
- Trouver le déterminant de **A** si **A** est réelle ;
- Résoudre le système d'équations simultanées en trouvant la solution de **AX = B**.

C'est la mémoire disponible qui est la seule limitation à la taille de vos matrices. Chaque matrice réelle nécessite un registre plus un registre pour chaque élément. Si vous désirez stocker plus d'une matrice, vous devrez utiliser la fonction **MATDIM**, décrite au chapitre suivant. Le programme **MATRX** ne stocke ni ne rappelle de matrices ; il fonctionne avec une matrice carrée simple **A** et une matrice à colonne unique **B**. Lorsque vous introduisez de nouveaux éléments dans **A** vous détruissez ses anciens éléments.

Utilisation

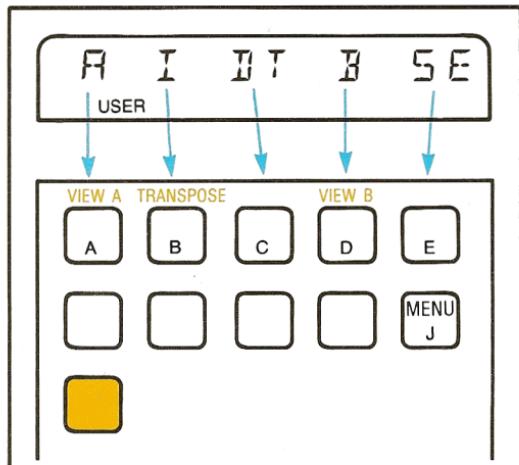
MATRX possède deux menus qui indiquent les touches qui correspondent à chaque fonction. Le premier menu, le *menu initial*, permet de choisir entre matrice réelle ou complexe :

Menu initial



Après avoir fait votre choix, entrez l'ordre des matrices et appuyez sur **R/S**, et vous lirez le *menu principal* :

Menu principal



A = introduction ou «édition»
(correction) de la matrice
A

I = inverse A en A^{-1}

D T = déterminant de A

B = introduction ou édition
de la matrice B

S E = «Solve Equation»
Résout l'équation

VIEW A = Voir A

TRANSPOSE = Transposer A

VIEW B = Voir B

Ce menu présente les choix disponibles dans le programme MATRIX. Appuyez sur **J** pour rappeler ce menu chaque fois qu'il sera nécessaire. Cette action ne dérangera en aucune façon votre programme.

Pour effacer le menu, appuyez sur **⬅**. Ceci vous montre le contenu du registre X, mais ne met pas fin au programme. Vous pouvez effectuer des calculs, puis rappeler le menu en appuyant sur **J**. Vous ne n'avez cependant pas besoin d'effacer l'affichage du programme avant d'effectuer un calcul.

- Le programme commence par vous demander une nouvelle matrice. Il vous demande de préciser s'il s'agit d'une matrice réelle ou complexe ainsi que les dimensions d'une matrice carrée pour **A**.
- Le programme n'efface pas les données de matrices précédentes, de sorte que les données précédentes—peut-être des données complètement obsolètes—se trouveront dans les nouvelles matrices **A** et **B** jusqu'à ce que vous les remplacez par de nouvelles valeurs.
- Chaque élément d'une matrice complexe a deux valeurs (une partie réelle et une partie imaginaire) et demande quatre fois plus d'espace-mémoire que le fait de stocker un élément d'une matrice réelle. Les sollicitations pour les parties réelles x_{11} , x_{12} , etc. sont **1:1= ?**, **1:2= ?**, etc. Les messages de sollicitation pour les parties complexes $x_{11} + iy_{11}$, $x_{12} + iy_{12}$, etc. sont **RE.1:1= ?**, **IM.1:1= ?**, **RE.1:2= ?**, **IM.1:2= ?**, etc.

Le chapitre suivant comporte une discussion exhaustive du problème du «Stockage des matrices». Il n'est cependant pas nécessaire de comprendre tout cela en détail pour utiliser le programme, parce que le programme vous sollicite et vous demande le format de mémoire adéquat, par le message **SIZE> =nnn**, si la taille de la mémoire n'est pas suffisante. Vous utiliserez alors **SIZE nnn** pour donner à la mémoire la taille convenable.

Le tableau suivant indique les frappes et séquences de frappe à utiliser pour exécuter des opérations sur les matrices du programme **MATRIX**. Toutes ces fonctions sont également disponibles en tant que fonctions du HP-41, décrites dans le chapitre suivant.

Instructions d'utilisation de **MATRIX**

Instructions	Frappez	Taille : variable*
		Affichage
1. Lancez le programme MATRIX .	XEQ MATRIX †	RL CX
2. Choisissez : matrice réelle (RL) ou complexe (CX).	A (RL) ou B (CX)	ORDER=?
3. Entrez la dimension, n , de votre matrice carée. A.	n R/S	A I DT B SE
4. Entrez les éléments de A. Le point d'interrogation vous demande de changer l'élément en cours, si vous le désirez. Appuyez ensuite sur R/S pour accéder à l'élément suivant.	A R/S ⋮ R/S R/S	$1:1=a_{11}$? ou RE.1:1=a₁₁ ? $1:2=a_{12}$? ou IM.1:1=y₁₁ ? ⋮ $n:n=a_{nn}$? ou IM.n:n=y_{nn} ? A I DT B SE
Pour réviser et éditer la matrice A, répétez simplement le processus. Pour laisser la saisie existante inchangée, appuyez sur R/S .		
5. Pour éditer un élément spécifique $a_{i,j}$, entrez en mode d'édition en appuyant sur A , puis spécifiez l'élément sous la forme $iii.jjj$. ‡	A $iii.jjj$ A ⋮ R/S	$i:j=a_{i,j}$? ou RE.i:j=a_{i,j} ? ⋮ A I DT B SE
Si $iii.jjj$ n'existe pas, l'éditeur se termine et renvoie au menu principal.		
Utilisez R/S pour passer aux éléments suivants et quitter l'éditeur.		

Instructions d'utilisation de MATRX (Suite)

Instructions	Frappez :	Affichage
6. Pour examiner simplement la matrice A :	[A] [R/S] § ⋮ [R/S] §	1:1= a_{11} OU RE.1:1= a_{11} 1:2= a_{12} OU IM.1:1= y_{11} ⋮ A I DT B SE
Pas de ? ; ces valeurs ne peuvent être changées.		
7. Pour entrer, modifier et voir la matrice colonne, B, suivez les étapes 4, 5 et 6, mais utilisez [D] (B) et [D] ([B]). B est dimensionnée automatiquement à une colonne sur n lignes (étape n° 3).		
8. Pour terminer l'édition et revenir au menu :	[J]	A I DT B SE
9. Exécutez une opération sur les matrices :		
■ Inversez A en A^{-1} . Ceci remplace la matrice A.	[B] (I)	A I DT B SE
Visualisez A^{-1} .	[A] [R/S] § ⋮ [B]	A I DT B SE
■ Transposez A (si elle est réelle) en A^T . Ceci remplace la matrice A (si vous aviez inversé A, ré-inversez la d'abord ; si vous aviez trouvé $\det(A)$ ou résolu pour X, il vous faudra inverser A deux fois pour la rétablir avant de la transposer. Voyez «Remarques» dans cette section pour plus d'informations).	[A] [R/S] § ⋮ [A]	A I DT B SE
Visualisez A^T .	[A] [R/S] § ⋮ [C] (DT) [R/S] §	DET=résultat A I DT B SE
■ Déterminante de A (si elle est réelle), $\det(A)$. (Si vous avez inversé A, ré-inversez-la d'abord). Cette opération remplace A par sa forme décomposée LU. Voyez «Remarques»		

Instructions d'utilisation de MATRX (suite)

Instructions	Fappez	Affichage
<p>■ Résolvez le système d'équations décrit par $AX = B$. Ceci détermine X, qui remplace B (et remplace A par sa forme décomposée LU ; voir «Remarques»).</p> <p>Visualisez X (qui remplace B).</p>	<p>E (SE)</p> <p>D (B) R/S § ⋮</p>	A I DT B SE

* La taille de ce programme dépend de la taille des matrices qui y sont impliquées. Il est de $(ordre^2 + ordre + 2)$ pour les matrices réelles **A** et **B** ; $[4(ordre^2) + 2(ordre) + 2]$ pour les matrices complexes **A** et **B**. Notez cependant que le programme vous indiquera la taille de mémoire à choisir si elle est insuffisante.

† Pour exécuter un programme, appuyez sur **[XEQ]** **[ALPHA]** nom alpha du prog. **[ALPHA]** ou utilisez une touche redéfinie.

‡ Vous pouvez éliminer les zéros avant la virgule dans la partie *i* et les zéros après la virgule dans la partie *j*. Les zéros deviennent 1 par défaut : exemple : 0,000 devient 1,001.

§ Si une imprimante est reliée, l'affichage revient automatiquement au menu principal après impression des résultats.

Une liste des messages d'erreur concernant les opérations sur les matrices se trouve dans «Messages d'erreur», dans le chapitre suivant.

Remarques

Modification de la matrice originelle. La matrice qui a été entrée, **A**, est modifiée par les opérations pour en trouver l'inverse, le déterminant, la forme transposée et la solution de l'opération sur les matrices. Vous pouvez ré-inverser A^{-1} et re-transposer A^T pour retrouver la forme originale de **A**. Si cependant vous avez calculé la déterminante ou la matrice solution, alors **A** est dans sa forme décomposée LU . Pour retrouver **A**, inversez-la deux fois. La décomposition LU n'interfère pas avec une opération MATRX subséquente, sauf une transposition ou «édition»*. Pour plus de renseignements sur le décomposition LU , lisez «La décomposition LU» dans le chapitre suivant («Fonctions sur les matrices»).

* N'essayez pas d'éditer une matrice décomposée LU à moins d'en changer tous les éléments.

Stockage des matrices. Le programme MATRX stocke une matrice **A** en commençant au R₀ de la mémoire principale ; il est nommé **R0**. Sa matrice colonne **B** est stockée ensuite et la matrice résultat **X** écrase **B**. Faites référence au chapitre «Fonctions matricielles» et lisez les explications sur la manière dont les matrices sont nommées et stockées et l'espace qui leur est nécessaire.

MATRX ne peut accéder à aucune autre matrice, à l'exception de la **R0** et la matrice colonne correspondante.

Touches redéfinies. Ce programme utilise des labels alpha locaux (comme expliqué dans le manuel d'utilisation du HP-41) assignés aux touches **[A]-[E]**, **[J]**, **[A]**, **[B]** et **[D]**. Ces définitions locales sont outrepassées par n'importe quelle redéfinition que vous aurez faite de l'une ou l'autre de ces touches, redéfinitions qui prennent le pas sur le programme. *Assurez-vous donc bien d'annuler toute redéfinition de touches avant d'utiliser ce programme*, et à l'avenir, essayez d'éviter de redéfinir ces touches.

Exemples

Dans le système d'équations en début de chapitre, nous avons l'équation matricielle **AX = B** ou

$$\begin{bmatrix} 3,8 & 7,2 \\ 1,3 & -0,9 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 16,5 \\ -22,1 \end{bmatrix}$$

Trouvez l'inverse, la déterminante et la transposée de **A**, puis trouvez la matrice solution, **X**.

Frappez

[FIX] 4

[XEQ] [SIZE] 008

[XEQ] [MATRX]

[A] (RL)

2 [R/S]

Affichage

RL CX

ORDER=?

A I DT B SE

Définit le format d'affichage utilisé ici. Optionnel—définit le nombre de registres de stockage nécessaires pour le programme. Ce n'est pas nécessaire si vous avez déjà affecté **SIZE ≥ 008** .

Lance le programme **MATRX**.

Sélectionne une matrice réelle.

Dimensionne une matrice carrée 2×2 .

Frappez :**[A]**

3.8 **[R/S]**
 7.2 **[R/S]**
 1.3 **[R/S]**
 0.9 **[CHS]** **[R/S]**

[A]

[R/S] *
[R/S] *
[R/S] *
[R/S] *
[R/S] *

[B] (I)

[A]
[R/S] *
[R/S] *
[R/S] *
[R/S] *
[R/S] *

[B] (I)

[B]
[A]
[R/S] *
[R/S] *
[R/S] *
[R/S] *
[B]

[C] (DT)
[D] (B)

16.5 **[R/S]**22.1 **[CHS]** **[R/S]**

[D] (**[B]**)
[R/S] *
[R/S] *
[E] (SE)

Affichage**1:1 = a_{11} ?****1:2 = a_{12} ?****2:1 = a_{21} ?****2:2 = a_{22} ?****A I DT B SE****1:1 = 3,8000****1:2 = 7,2000****2:1 = 1,3000****2:2 = -0,9000****A I DT B SE****A I DT B SE****1:1 = 0,0704****1:2 = 0,5634****2:1 = 0,1017****2:2 = -0,2973****A I DT B SE****A I DT B SE****A I DT B SE****1:1 = 3,8000****1:2 = 1,3000****2:1 = 7,2000****2:2 = -0,9000****A I DT B SE****A I DT B SE****DET = -12,7800****1:1 = b_{11} ?****2:1 = b_{21} ?****A I DT B SE****1:1 = 16,5000****2:1 = -22,1000****A I DT B SE****A I DT B SE**

Lance l'édition de **A** et affiche l'ancien élément a_{11} .

Introduit 3,8 pour a_{11} .

Introduit a_{22} et renvoie au menu principal.

Affiche le contenu de **A** pour examen.

Inverse **A**.

Affiche le contenu de **A**, maintenant A^{-1} .

Ré-inverse A^{-1} en la valeur originale de **A**.

Transpose **A**.

Affiche le contenu de **A**, maintenant A^T .

Retranspose A^T en **A** originale.

Det(**A**).

Introduit l'éditeur de **B** et affiche l'ancien élément b_{11} .

Introduit 16,5 pour b_{11} .

Introduit b_{21} et renvoie au menu principal.

Affiche le contenu de **B** pour examen.

Résout le système **AX = B**, plaçant **X** en **B**.

Frappez

(B)
 R/S *
 R/S *

Affichage

1:1 = -11,2887
 2:1 = 8,2496
 A I DT B SE

Affiche la matrice solution (dans B).

Trouvez l'inverse de cette matrice complexe :

$$\begin{bmatrix} 1 + 2i & 3 + 3i \\ 4 + 5i & 6 + 7i \end{bmatrix}$$

Frappez :

XEQ SIZE 017

XEQ MATRIX

B (CX)

2 R/S

A

1 R/S

2 R/S

3 R/S

4 R/S

1.002 A

R/S

3 R/S

Affichage

RL CX

ORDER = ?

A I DT B SE

RE.1:1 = a_{11} ?

IM.1:1 = y_{11} ?

RE.1:2 = a_{12} ?

IM.1:2 = y_{12} ?

RE.2:1 = a_{21} ?

RE.1:2 = 3,0000 ?

IM.1:2 = 4,0000 ?

RE.2:1 = a_{21} ?

Pour une matrice complexe A,
 $2^2 \times 4 + 1 = 17$.

Recommence le programme.

Dimensionne une matrice complexe 2×2 .

Hum ! Erreur de saisie pour y_{12} . Il fallait frapper un 3 et non un 4.

Replace l'éditeur en a_{12} .

La partie imaginaire. (Valeur erronée)

La valeur correcte pour y_{12} est introduite. Continuez la saisie des données.

* R/S cette frappe n'est pas nécessaire si une imprimante est reliée au calculateur.

Frappez4 5 6 7 (I)2.002 (ou **Affichage****IM.2:1** = y_{21} ?**RE.2:2** = a_{22} ?**IM.2:2** = y_{22} ?**A I DT B SE****A I DT B SE****RE.1:1** = $-0,9663$ **RE.2:2** = $-0,2360$ **IM.2:2** = $-0,0225$ **A I DT B SE**

Introduit le dernier élément et renvoie au menu principal.

Inverse A.

Visualisation de A^{-1} .*Affiche $a_{22} + iy_{22}$.*

Sort de l'éditeur.

* Si une imprimante est reliée au système, l'opération de visualisation provoque automatiquement l'impression de la matrice entière et le ré-affichage du menu.

FONCTIONS MATRICIELLES

Sommaire

Etablir une matrice	31
Nommer une matrice	32
Dimensionner une matrice	32
Stocker une matrice	33
Utiliser des éditeurs de matrices	34
Comment spécifier une matrice	36
Paramètres de matrices choisis par défaut	37
Messages d'erreur	38
Stocker et rappeler des éléments matriciels	
isolés	39
Accès direct à chaque élément	39
Accès séquentiel aux éléments	41
Fonctions matricielles	42
Arithmétique matricielle	43
Principales opérations	44
Autres fonctions «utilitaires»	46
Matrices complexes	50
Représentation des éléments complexes	50
Fonctions et matrices complexes	51
Décomposition LU	52
Exemples	53
Tableau alphabétique des fonctions	58

FONCTIONS MATRICIELLES

Ce chapitre fait pendant au chapitre précédent, «Le programme de calcul matriciel». Ce chapitre couvre toutes les fonctions disponibles offertes par ce module à l'utilisateur expérimenté.

Il est possible de créer, manipuler et stocker des matrices réelles et complexes. C'est la quantité de mémoire disponible qui sera la seule limitation au nombre de matrices. Si vous possédez de la *mémoire étendue* (c.à.d. un module mémoire et de fonctions étendues HP 82180A ou un HP-41CX), vous pouvez aussi y stocker des matrices.

Les opérations disponibles dans ce module comprennent inversion, transposition, identification du déterminant, résolution d'un système d'équations et arithmétique matricielle. De plus, des éléments indépendants, à l'intérieur des matrices et entre elles, peuvent être manipulés.

Créer une matrice

Tout ce qui est nécessaire pour créer une matrice est son nom et ses dimensions. La fonction **MATDIM** utilise le nom (dans le registre alphabétique) et les dimensions *mmm.nnn* dans le registre *X* pour créer une matrice.

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & a_{ij} & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix}$$

Cette fonction n'efface pas les éléments d'une nouvelle matrice en mémoire principale, mais retient les éléments de la matrice ou des registres précédents. Elle efface cependant les éléments d'une nouvelle matrice dans la mémoire étendue.

Vous saisissez alors les valeurs—numériques ou alphabétiques—dans la matrice au moyen d'un «éditeur» de matrice (Cf. page 34).

Nommer une matrice

Le nom que vous aurez donné à une matrice déterminera l'endroit où elle sera stockée. Une matrice qui doit être stockée en mémoire principale (*non-étendue*) doit être nommée

R_{xxx},

où *xxx* représente trois chiffres. Les zéros de tête peuvent être éliminés ; La matrice sera stockée en commençant à R_{xxx} . Par exemple, **R007** est similaire à **R7** et amènera le stockage de ce nom de matrice en R_{07} .

Si vous spécifiez une matrice **R**, son nom et son emplacement seront **R0**.

Une matrice qui devrait être stockée en *mémoire étendue* peut être nommée avec un maximum de sept caractères alphabétiques, à l'exception de la lettre X, réservée au nom du registre, et de la lettre R suivie de trois chiffres, qui est réservée au stockage en mémoire principale. Nul besoin de spécifier le type de fichier, un type de fichier unique aux matrices sera automatiquement attribué.

Utilisez le registre alphabétique pour spécifier les noms de matrices. Lorsque plusieurs noms sont spécifiés (en tant que paramètres de certaines fonctions), séparez-les par des virgules.

MNAME? renvoie le nom de la matrice se trouvant dans le registre alphabétique.

Dimensionner une matrice

Définissez les dimensions d'une nouvelle matrice en tant que *mmm.nnn*, où *m* est le nombre de lignes et *n* le nombre de colonnes. Les zéros avant la virgule, pour *m*, et les zéros après la virgule pour *n* peuvent être omis.

Pour une matrice complexe, spécifiez *mmm.nnn* en tant que *deux fois* le nombre de lignes et *deux fois* le nombre de colonnes (relisez «Travailler avec des matrices complexes»).

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$$

$$mmm.nnn = 2,003$$

$$\begin{bmatrix} 1 + i & 2 + 3i \\ 4 + 5i & 6 + 7i \end{bmatrix}$$

$$mmm.nnn = 4,004$$

Une partie nulle prend par défaut la valeur 1, zéro est donc équivalent à 1,001, 3 à 3,001, 0,023 à 1,023.

MATDIM dimensionne une nouvelle matrice ou redimensionne une matrice existante à de nouvelles dimensions.

DIM? renvoie dans le registre X les dimensions de *mmm.nnn* de la matrice spécifiée dans le registre alphabétique ; un registre alphabétique vide spécifie la matrice en cours).

Stocker une matrice

Les éléments d'une matrice sont stockés en mémoire dans un ordre de gauche à droite sur chaque ligne, de la première à la dernière. Chaque élément occupe un registre de stockage de données. Un nombre complexe nécessite quatre registres pour y stocker les parties qui le composent.

Espace mémoire. Une matrice en mémoire principale occupe $(m \times n) + 1$ registres de stockage de données, où m est le nombre de lignes de la matrice complexe et n le nombre de colonnes de la matrice complexe.

Remarque : Pour dimensionner correctement une matrice en mémoire principale, la taille (SIZE) de la mémoire doit être suffisante. Si ce n'est pas le cas, le message **NONEXISTENT** répondra à toute tentative d'exécution de **MATDIM**. Allouez plus de registres de stockage par l'instruction (**SIZE** *nnn*) et refaites l'essai.

Une matrice se trouvant en mémoire principale représente une longueur de fichier de $m \times n$. ($2m \times 2n$ pour une matrice complexe). Son fichier-type est unique aux matrices. N'utilisez pas la fonction **CLFL** avec une matrice stockée en mémoire principale : cela détruirait une partie des informations contenues dans l'en-tête du fichier. Utilisez en lieu et place **PURFL** pour «purger», effacer la matrice entière.

Modifier les dimensions. Si vous redimensionnez une matrice, les éléments existants sont réassignés à de nouveaux éléments en fonction des nouvelles dimensions. Les éléments anciens superflus sont perdus ; d'autres éléments prennent les valeurs déjà présentes dans les nouveaux registres - sauf en mémoire étendue, où de nouveaux éléments reçoivent la valeur zéro.

AVERTISSEMENT

Lorsque **MATDIM** est utilisée pour redimensionner une matrice stockée en mémoire étendue, la position du pointeur n'est pas rajustée. S'il devait se produire que le pointeur se trouve placé sur un élément situé au-delà des nouvelles limites de la matrice redimensionnée, elle doit être replacée dans ces nouvelles limites en exécutant soit **MSIJ**, soit **MSIJA** avec des indices valables avant que le pointeur puisse à nouveau être utilisé.

Les matrices existantes en mémoire étendue ne peuvent être redimensionnées pour remplir complètement la mémoire étendue. Une *nouvelle* matrice peut cependant être redimensionnée pour remplir complètement la mémoire étendue.

Redimensionner 2×3 en 2×2

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \text{ perdus : } 5 \quad 6$$

Redimensionner 2×3 en 2×4

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & ? & ? \end{bmatrix}$$

Voici ce qui se produit lors de chaque dimensionnement de nouvelle matrice, puisque les éléments de la matrice précédente demeurent jusqu'à la prochaine modification.

Utiliser des éditeurs de matrices

Il existe deux éditeurs de matrices : **MEDIT** pour matrices réelles et **CMEDIT** pour matrices complexes. Ils sont par ailleurs fort semblables.

Les éditeurs ont trois utilisations :

- Saisir de nouvelles valeurs dans les éléments d'une matrice.
- Réviser et modifier («éditer») les éléments d'une matrice, soit par ordre séquentiel, soit par accès aléatoire à certains éléments pris isolément.
- Visualiser (sans pouvoir les changer) les éléments d'une matrice (indicateur binaire 08 armé).

Lorsque vous exéutez **MEDIT** ou **CMEDIT**, l'éditeur affiche l'élément 1,1 de la matrice spécifiée dans le registre alphabétique, ou de la matrice en cours si le registre alphabétique est vide. L'appui sur la touche **R/S** fait avancer l'affichage d'élément en élément ; dans le cas d'une matrice complexe, chaque partie de l'élément complexe est montrée séparément.

Fonction	Affichage	Fonction	Affichage
MEDIT	1:1=1,0000?	CMEDIT	RE.1:1=1,0000?
R/S	1:2=2,0000?	R/S	IM.1:1=1,0000?
:	:	:	:
R/S	<i>(registre X)</i>	R/S	<i>(registre X)</i>

Le signe **?** à la fin de la ligne d'affichage indique que cette valeur peut être modifiée. En fait, le programme vous demande si ceci est bien la valeur désirée. Si ce n'est pas le cas, saisissez la nouvelle valeur et appuyez sur la touche **R/S**. Vous ferez ceci dans le cas d'une matrice neuve comme dans celui de la modification d'un élément d'une matrice.

Si vous appuyez sur **R/S** sans saisir de nouvelle valeur, la valeur en cours reste inchangée.

Visualiser sans modification. Si vous armez l'indicateur 08, l'éditeur vous permettra d'examiner les éléments, mais non de les modifier. L'affichage dans ce cas ne présente pas de signe **?** à la fin de la ligne.

1:1=1,0000

Si une imprimante est reliée, et que l'indicateur 08 est armé, elle imprimera tous les éléments de la matrice sans faire de pause.

Accéder directement à l'un des éléments. *Lorsque l'éditeur est actif*, l'accès à n'importe quel élément spécifique est possible et le clavier personnel est également actif. Pour accéder à l'élément se trouvant sur la *i*ème ligne et la *j*ème colonne, frappez *iii.jjj* et appuyez sur **A**, comme sous le programme **MATRIX**. Les zéros avant la virgule pour *iii* et après la virgule pour *jjj* peuvent être éliminés.

Pour une matrice complexe, il est possible d'accéder directement à la *partie réelle* de l'élément *i* et *j*.

Utilisez ensuite **R/S** pour accéder à la partie imaginaire.

Frappez

ALPHA nom de matrice **ALPHA**
XEQ **MEDIT**
3.003 **A**

ALPHA nom de matrice complexe **ALPHA**
XEQ **CMEDIT**
3.003 **A**
R/S

Affichage

1:1=1,0000?

3:3=6,0000?

RE.1:1=1,0000?

RE.3:3=6,0000?

IM.3:3=7,0000?

Les zéros avant la virgule peuvent être supprimés dans la partie i et les zéros *après* la virgule dans la partie j . Une partie nulle devient 1 par défaut.

Sortir de l'éditeur. Pour quitter l'éditeur avant qu'il ait atteint le dernier élément, soit :

- Appuyez sur **J**.
- Soit essayez d'accéder à un élément nonexistant. Par exemple, dans une matrice 4×4 , appuyez sur 5 **A**.

Comment spécifier une matrice

Dans l'opération de multiplication $\mathbf{AB} = \mathbf{C}$, \mathbf{A} et \mathbf{B} sont connus et l'on recherche la matrice-produit, \mathbf{C} . Au cours de l'opération, le calculette doit recevoir l'identité des matrices \mathbf{A} et \mathbf{B} et posséder les instructions nécessaires pour placer la matrice-résultat, \mathbf{C} , à l'endroit désiré. Cependant, la matrice résultat peut être la même que l'une des matrices introduites. *Toutes* les matrices données doivent déjà exister en tant que matrices nommées et dimensionnées. Naturellement, seules \mathbf{A} et \mathbf{B} doivent contenir des données valables.

Quelques fonctions utilisent une seule matrice d'introduction et certaines fonctions utilisent *automatiquement* l'une des matrices introduites comme véhicule de sortie de données. Le nombre minimum de matrices à spécifier est *un* et le maximum *trois*.

Une fonction matricielle recherche dans le registre alphabétique les noms des matrices dont elle a besoin pour les données à introduire et les résultats produits. Avant d'exécuter cette fonction, il vous faut *spécifier* tous les paramètres nécessaires dans une ligne du registre alphabétique, en séparant chacun d'entre eux par une virgule :

Registre alpha

matrice introduite[,matrice introduite][,matrice résultat]

Par exemple,

Registre alphabétique	A,B,C
	XEQ M*M

multipliera les matrices **A** et **B**, plaçant le résultat dans une matrice existante, **C**.

Opérations scalaires. Entrées et sorties scalaires doivent se trouver dans le registre **X** et cet emplacement ne doit pas être spécifié à moins que la fonction en question puisse utiliser *soit* un scalaire *soit* une matrice pour le même paramètre introduit. Pour spécifier le registre **X**, utilisez **X**.

Par exemple, **MATDIM** nécessite une entrée scalaire et un nom de matrice, nul besoin donc de spécifier le registre **X**. Par ailleurs, les fonctions scalaires arithmétiques, telles **MAT***, peuvent utiliser indifféremment deux matrices ou un scalaire et une matrice pour l'introduction des données. Il faut donc, si vous désirez l'utiliser, spécifier **X**.

La matrice en cours. Par *matrice en cours*, on entend celle à laquelle on a accédé - ou que l'on a utilisé - en dernier lieu. Si le registre alphabétique est vide et que l'on exécute une fonction de matrice qui requiert une spécification de matrice, c'est la matrice courante qui est utilisée par défaut. S'il n'y a pas de matrice courante, le résultat sera l'affichage de **UNDEF ARRAY**.

La matrice résultat d'une fonction matricielle devient, après cette opération, la matrice en cours.

Pour trouver le nom de la matrice en cours, exécutez **MNAME?**. Son nom est renvoyé dans le registre alphabétique.

Paramètres de matrices choisis par défaut

Si vous ne spécifiez pas l'une ou toutes les matrices nécessaires à la fonction, certains paramètres *par défaut* existent (les paramètres par défaut sont ceux qui sont automatiquement attribués si vous n'en spécifiez pas). Les valeurs par défaut les plus fréquemment rencontrées seront celles de la matrice courante. Si vous ne spécifiez pas de nom de matrice particulier et que le registre alphabétique est vide, la matrice en cours devient également matrice par défaut.

Pour les opérations demandant jusqu'à trois noms de matrices en registre alphabétique, le tableau suivant illustre les conventions de notation des paramètres.

Spécifications

Contenu du registre alphabétique	Matrices spécifiées
A,B,C	A, B, C
A,B	A, B, B
A	A, A, A
A,,B	A, A, B
,A,B	en cours, A, B
,A	en cours, A, A
,,A	en cours, en cours, A
X,A,B	Registre X, A, B
X,A	Registre X, A, A
A,X	A, Registre X, A
A,,X	A, A, A (ignore X)
X	Registre X, en cours, en cours
(vide)	en cours, en cours, en cours

Messages d'erreur

Référez-vous à la documentation de votre HP-41 pour ce qui est des messages d'erreur que vous ne trouvez pas ici.

ALPHA DATA : la matrice spécifiée contient des données alphabétiques et ne peut donc être utilisée. La matrice est inchangée

DATA ERROR : la valeur dans le registre X, Y ou Z n'est pas valable.

DATA ERROR X : la valeur du registre X n'est pas valable.

DATA ERROR Y : la valeur du registre Y n'est pas valable.

DIM ERROR : la matrice spécifiée n'a pas été dimensionnée en fonction de l'opération en cours.

END OF ARRAY : une fonction utilisant le pointeur a été tentée alors que le pointeur se trouve en dehors de ses limites.

NAME ERROR : un nom de matrice non valable a été choisi, tel que «X», ou le nombre de noms de matrices est incompatible avec cette fonction.

NO ROOM : il n'y a plus assez d'espace pour loger une matrice en mémoire étendue.

NO X-MEMORY : on ne peut créer une matrice en mémoire étendue...si le calculateur utilisé n'en possède pas : par exemple un HP-41C/CV sans module HP 82180A, mémoire/fonctions étendues.

NONEXISTENT : il n'y a pas assez de registres de stockage en mémoire principale pour stocker la matrice. Modifiez la taille de la mémoire avec (`SIZE`*nnn*) pour pouvoir y placer la nouvelle matrice.

NOT ARRAY FL : une opération sur les matrices a été tentée sur un fichier en mémoire étendue qui n'est *pas* un fichier matriciel.

NOT CPX : apparaît si vous décidez d'utiliser `CMEDIT` avec une matrice réelle d'ordre impair (pas de $2m \times 2n$).

TRY AGAIN : (faites un autre essai) apparaît si vous essayez d'exécuter `MATDIM` avec moins de deux registres de mémoire programme disponible. Re-taillez la mémoire de stockage de données avec *moins* de registres de mémoire ou utilisez la commande `CLP` pour éliminer un programme.

UNDEF ARRAY : apparaît si vous exécutez une fonction nécessitant une spécification de matrice et si le registre alphabétique ne contient pas de spécification de matrice valide.

Stocker et rappeler des éléments matriciels isolés

L'éditeur de matrices offre une méthode de stockage et de révision des éléments matriciels. Pour la programmation, vous pouvez utiliser les instructions suivantes pour manipuler les éléments matriciels individuels.

Un élément spécifique est identifié par la valeur *iii.jjj* en fonction de son emplacement dans la *i*ème ligne et la *j*ème colonne. Les zéros avant la virgule peuvent être éliminés dans l'index *i* et les zéros après la virgule, dans l'index des *j*.

Accès direct à chaque élément

Pour stocker, puis rappeler un élément isolé, déterminer d'abord la valeur *iii.jjj*, (les coordonnées, ligne et colonne) utilisée par le pointeur, puis stocker ou rappeler la valeur de l'élément de, ou dans le registre X. Pour passer à un autre élément, modifier le pointeur et ses valeurs *iii.jjj*.

C'est la valeur du pointeur qui détermine l'élément en cours.

Comment régler et rappeler le pointeur

Fonction	Résultat
[MSIJA] (règle le pointeur par le registre alphabétique)	Règle en <i>iii.jjj</i> le pointeur de la matrice spécifiée. Données : nom de matrice dans le registre alphabétique. <i>iii.jjj</i> dans le registre X.
[MSIJ] (règle le pointeur)	Règle le pointeur de la matrice en cours à <i>iii.jjj</i> . Données : <i>iii.jjj</i> dans le registre X.
[MRIJA] (rappelle le pointeur par le registre alphabétique)	Rappelle dans le registre X le pointeur de la matrice spécifiée. Données : nom de matrice en registre alphabétique. Résultat : <i>iii.jjj</i> dans le registre X.
[MRIJ] (rappelle le pointeur)	Rappelle dans le registre X le pointeur de la matrice en cours . Résultat : <i>iii.jjj</i> dans le registre X.

Les fonctions suivantes déplacent le pointeur d'éléments sur les lignes (*iii*) ou les colonnes (*jjj*). Une fois la fin d'une colonne atteinte (par l'index *i*) ou la fin d'une ligne (par l'index *j*) l'index avance jusqu'à la ligne ou la colonne suivante la plus grande et arme l'indicateur binaire 09. Si l'index avance au-delà de la taille de la matrice, les indicateurs 09 et 10 sont armés. Si ces situations ne se produisent pas, les indicateurs sont effacés chaque fois que les fonctions sont exécutées.

Augmenter et diminuer les coordonnées du pointeur

Fonction	Résultat
[I+]	Augmente la valeur <i>iii</i> du pointeur de un.
[I-]	Diminue la valeur <i>iii</i> de un.
[J+]	Augmente la valeur <i>jjj</i> du pointeur de un.
[J-]	Diminue la valeur <i>jjj</i> de un.

Stocker et rappeler la valeur de l'élément

Fonction	Résultat
[MS] (=matrix store) : stocker matrice	Stocke les valeurs du registre X dans l'élément concerné de la matrice en cours. Données : valeurs dans le registre X.
[MR] (=matrix recall) : rappeler matrice.	Rappelle dans le registre X la valeur de l'élément en cours dans la matrice en cours. Résultat : valeur dans le registre X.

Accès séquentiel aux éléments

Les fonctions suivantes offrent une manière plus rapide et plus automatique d'ajuster le pointeur de manière à accéder à chaque élément. Elles utilisent le stockage et le rappel de valeurs avec l'augmentation et la diminution des index i ou j , de sorte que le pointeur est automatiquement placé sur l'élément suivant.

Stocker et rappeler les valeurs des éléments

Fonction	Résultat
MSC+ (<i>matrix store by column</i> : stocke les matrices par colonne)	Stocke les valeurs du registre X dans l'élément en cours et avance le pointeur sur l'élément suivant de la colonne. Données : valeurs dans le registre X.
MSR+ (<i>matrix store by row</i> : stocke les matrices par ligne)	Stocke les valeurs du registre X dans l'élément en cours et avance le pointeur sur l'élément suivant de la ligne. Résultat : valeurs dans le registre X.
MRC+ (<i>matrix recall by column</i> , rappel des matrices par colonne)	Rappelle les valeurs dans le registre X à partir de l'élément courant et ensuite avance le pointeur vers l'élément suivant de la colonne. Résultat : valeur dans le registre X.
MRR+ (<i>matrix recall by row</i> , rappel des matrices par ligne)	Rappelle la valeur du registre X au départ de l'élément courant et avance le pointeur sur l'élément suivant de la ligne. Résultat : valeur dans le registre X.
MRC- (<i>matrix recall backwards by column</i> , rappel régressif des matrices par colonne)	Rappelle la valeur vers le registre X à partir de l'élément en cours et ensuite ramène le pointeur à l'élément précédent de la colonne. Résultat : valeurs dans le registre X.
MRR- (<i>matrix recall backwards by row</i> , rappel régressif des matrices par ligne).	Rappelle la valeur vers le registre X à partir de l'élément en cours puis ramène le pointeur à l'élément précédent de la ligne. Résultat : valeurs dans le registre X.

Lorsque la fin de la colonne ou de la ligne est atteinte, l'index du pointeur avance vers la colonne ou la ligne suivante (ou précédente). S'il est déplacé en dehors des limites de la matrice, il ne peut y être ramené à l'aide de ces fonctions. Il faut alors utiliser **MSIJ** ou **MSIJA**.

La séquence de frappe suivante créera la matrice **ABC** (en mémoire étendue).

$$\mathbf{ABC} = \begin{bmatrix} 5 & 6 & 7 \\ 8 & 9 & 10 \end{bmatrix}$$

Frappez**FIX** 4

ALPHA ABC ALPHA

2.003 XEQ MATDIM

0 XEQ MSIJ

5 XEQ MSR+

6 XEQ MSR+

7 XEQ MSR+

8 XEQ MSR+

9 XEQ MSR+

10 XEQ MS

SF 08

XEQ MEDIT

Affichage

2,0030

0,0000

5,0000

6,0000

7,0000

8,0000

9,0000

10,0000

1:1=5,0000

1:2=6,0000

1:3=7,0000

2:1=8,0000

2:2=9,0000

2:3=10,0000

Etablit le format d'affichage.

Nom de la matrice en mémoire étendue.

Dimensionne la matrice ABC : 2 lignes x 3 colonnes.

Règle le pointeur sur 1,001.

Introduit l'élément et avance le pointeur jusqu'à la colonne suivante pour l'entrée suivante, plaçant l'indicateur binaire sur 09.

Le pointeur avance automatiquement vers la deuxième ligne.

Ceci place l'éditeur en mode d'affichage ; si une imprimante est reliée, ce système assure un examen rapide des éléments.

Examinons ABC à présent (ABC se trouve toujours dans le registre alphabétique). Si une imprimante n'est pas reliée, appuyez sur R/S pour examiner chaque élément un à un.

Sort de l'éditeur.

Fonctions matricielles

Cette partie est consacrée à une définition des fonctions sur les matrices autres que le dimensionnement, le stockage et le rappel des fonctions dont nous venons de parler. Vous trouverez en page 58 une liste de toutes les fonctions concernant les matrices contenues dans ce module.

Remarquez que la plupart de ces fonctions ne sont pas valables pour les matrices comportant des données alphabétiques et que la plupart ne sont pas applicables aux matrices complexes. En tous cas, une matrice complexe est considérée en tant que matrice réelle pour la plupart des fonctions excepté **CREDIT**. Faites référence à «Travail avec des matrices complexes».

Arithmétique matricielle

Ces fonctions sont : addition, soustraction, multiplication et division avec un scalaire ainsi que multiplication matricielle. Elles peuvent utiliser deux matrices en tant qu'opérandes ou un scalaire et une matrice. Lorsque deux matrices sont utilisées, elles ne doivent pas être de la même dimension, mais le nombre total d'éléments doit être le même dans chacune. Ces règles s'appliquent aussi à la matrice résultat. Remarquez que la notation $i-j$ dans le tableau suivant suppose que les dimensions des matrices sont identiques. Si ce n'est pas le cas, la notation $i-j$ ne serait pas d'application.

La multiplication de matrices, d'autre part, calcule chaque nouvel élément en faisant la somme des produits des éléments de la ligne de la première matrice avec les éléments de la colonne de la seconde. Le nombre de colonnes de la première matrice doit être égal au nombre de lignes de la seconde. La matrice résultat doit avoir le même nombre de lignes que la première matrice et le même nombre de colonnes que la seconde.

S'il y a une opérande scalaire, elle doit se trouver dans le registre X et X doit être spécifié dans le registre alphabétique.

Fonction	Résultat
MAT+ (matrix add) : addition matricielle	<p style="text-align: center;">Arithmétique scalaire</p> <p>Ajoute un élément scalaire ou matriciel à chaque élément.</p> <p>Entrée : nom de matrice A or X, nom de matrice B ou X, nom de matrice résultat C dans le registre alpha.</p> <p>Résultat : $c_{ij} = a_{ij} + x$ ou $c_{ij} = x + b_{ij}$ or $c_{ij} = a_{ij} + b_{ij}$ for all i, j in C.</p>
MAT- (matrix subtract) : soustraction matricielle	<p>Soustrait un élément scalaire ou matriciel de chaque élément.</p> <p>Entrée : nom de matrice A ou X, nom de matrice B ou X, nom de matrice résultat C dans le registre alpha.</p> <p>Résultat : $c_{ij} = a_{ij} - x$ ou $c_{ij} = x - b_{ij}$ ou $c_{ij} = a_{ij} - b_{ij}$ pour tout i, j dans C.</p>

Fonction	Résultat
MAT* (scalar matrix-multiply) multiplication scalaire-matrice	<p>Multiplie un élément scalaire ou matriciel par chaque élément.</p> <p>Entrée : nom de matrice A ou X, nom de matrice B ou X, nom de matrice résultat C dans le registre alphabétique.</p> <p>Résultat : $c_{ij} = a_{ij} \times x$ ou $c_{ij} = x + b_{ij}$ ou $c_{ij} = a_{ij} \times b_{ij}$ pour tout i, j dans C.</p>
MAT/ (matrix divide) : division matricielle	<p>Divise un élément scalaire ou matriciel dans chaque élément.</p> <p>Entrée : nom de matrice A ou X, nom de matrice B ou X, nom de matrice résultat C dans le registre alphabétique.</p> <p>Résultat : $c_{ij} = a_{ij} \div x$ ou $c_{ij} = x \div b_{ij}$ ou $c_{ij} = a_{ij} \div b_{ij}$ pour tout i, j dans C.</p>
M*M (matrix multiplication) : mult. matricielle	<p>Arithmétique non scalaire</p> <p>Calcule chaque nouvel élément de i et j en multipliant la ième ligne de A par la jème colonne de B.</p> <p>Entrée : nom de matrice A, nom de matrice B, nom de matrice résultat C dans le registre alphabétique, où C doit être différent de A et de B.</p> <p>Résultat : $c_{ij} = \sum_{k=1}^p a_{ik} \times b_{kj}$, où A a p colonnes et B, p lignes.</p>

Principales opérations

Ces quatre opérations principales sont : l'inversion, l'identification du déterminant, la transposition et la résolution d'un système d'équations linéaires.

Un système d'équations linéaires

$$a_{11}x_1 + a_{12}x_2 = b_1$$

$$a_{21}x_1 + a_{22}x_2 = b_2$$

peut être représentée par l'équation matricielle $\mathbf{AX} = \mathbf{B}$, où

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}, \quad \mathbf{X} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \quad \text{et} \quad \mathbf{B} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$$

A est la *matrice coefficient*, **B** la *matrice constante ou colonne* et **X** est la *matrice solution*. La matrice **B** est écrasée par la matrice **X** lors de la résolution de ce système.

Fonction	Résultat
MDET (déterminant)	Trouve le déterminant de la matrice réelle carrée donnée. Entrée : nom de matrice dans le registre alphabétique. Résultat : déterminant dans le registre X (remplace la matrice en forme LU décomposée).
MINV (inverse)	Inverse et remplace la matrice carrée donnée. Entrée : nom de matrice dans le registre alphabétique. Résultat : Remplace la matrice par son inverse.
MSYS (système d'équations)	Résout un système d'équations linéaires. Entrée : nom de matrice A, nom de matrice B dans le registre alphabétique. Résultat : la matrice solution X remplace B dans le système défini par l'équation matricielle $AX = B$. (Remplace A par sa forme décomposée LU).
TRNPS (transposée)	Transpose et remplace la matrice réelle donnée. Entrée : nom de matrice dans le registre alphabétique. Résultat : Remplace la matrice par sa forme transposée.

Remarque : On ne peut transposer ou modifier un quelconque élément d'une matrice *A* dont on a trouvé le déterminant ou dont on a trouvé la matrice solution parce que **MDET** et **MSYS** transforment la matrice *A* introduite en sa forme *LU décomposée* (Cf. «Décomposition LU» pour plus de détails). Il est cependant possible de retrouver la forme originelle de *A* à partir de sa forme décomposée en l'inversant *deux fois* (en exécutant **MINV** deux fois). La décomposition LU n'interfère pas avec les calculs de **MINV**, **MSYS** ou **MDET**.

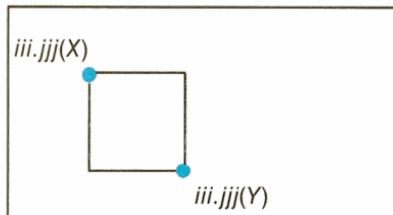
Autres fonctions «utilitaires»

Les fonctions restantes, également nommées *utilitaires*, sont celles destinées à permettre la copie et l'échange de parties de matrices et quelques fonctions diverses d'arithmétique : détermination de sommes, de normes, de maximums et de minimums et la réduction de matrices.

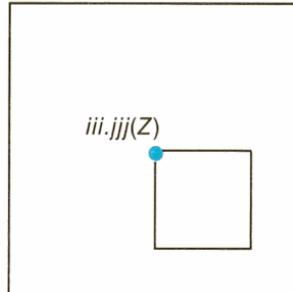
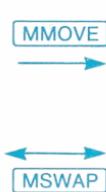
Déplacer et échanger des parties de matrices

Fonction	Résultat
C<>C (échanger des colonnes)	Echange les colonnes k et l d'une matrice. Entrée : nom de matrice dans le registre alphabétique. $kkk.lli$ dans le registre X.
R<>R (échanger des lignes)	Echange les lignes k et l dans une matrice. Entrée : nom de matrice dans le registre alphabétique. $kkk.lli$ dans le registre X.
MMOVE (matrix move, déplacement)	Copie la sous-matrice définie par les pointeurs dans la matrice source dans la zone définie par un pointeur dans la matrice de destination. Entrée : nom de matrice source A, nom de matrice destination B dans le registre alphabétique. dans le registre X : $iii.jjj$ pour l'élément initial de A ; dans le registre Y : $iii.jjj$ pour l'élément final de A ; dans le registre Z : $iii.jjj$ pour l'élément initial de B.
MSWAP (matrix swap, échange)	Echange la sous-matrice définie par les pointeurs en une matrice source dans une zone définie par le pointeur dans la matrice destination. Entrée : nom de matrice A, nom de matrice B dans le registre alphabétique. dans le registre X : $iii.jjj$ pour l'élément initial de A ; dans le registre Y : $iii.jjj$ pour l'élément final de A ; dans le registre Z : $iii.jjj$ pour l'élément initial de B.

Lors de l'exécution de **MMOVE** et de **MSWAP**, si A et B sont une même matrice et la sous-matrice source recouvre en partie la sous-matrice destination, les éléments sont traités dans l'ordre suivant : ordre inverse des colonnes (on commence par la dernière), puis ordre inverse des éléments (le dernier en premier lieu) au sein de chaque colonne.



Matrice source (A)



Matrice destination (B)

Lorsqu'une saisie sous la forme *iii.jjj* est attendue dans le registre *X*, une valeur zéro, soit pour la partie *i* soit pour la partie *j* est interprétée en tant que 1 (un zéro isolé est égal à 1,001). Ceci est vrai pour les valeurs *iii.jjj* que **MMOVE** et **MSWAP** attendent dans les registres *X* et *Z*, *mais non pour la valeur du pointeur dans le registre Y*.

Pour la saisie dans le registre *Y*, une valeur zéro pour la partie *i* est interprétée en tant que *m*, la dernière ligne, alors qu'une valeur zéro pour la partie *j* est interprétée en tant que *n*, la dernière colonne. Par exemple, dans une matrice 4×5 ,

Registre Y	Valeur du pointeur
0,000	4,005
3,000	3,005
0,003	4,003

Cette convention facilite la copie (ou l'échange) de matrices entières parce que simplement en effaçant la pile (**CLST**) ou en saisissant trois zéros, vous spécifiez les éléments 1,001 (*X*) et *mmm.nnn* (*Y*) pour la première matrice et l'élément 1,001 (*Z*) pour la seconde matrice, définissant ainsi deux matrices entières.

Les instructions suivantes copieraient la matrice **R0** de dimensions non définies en une nouvelle matrice, **R30** :

```

ALPHA R0 ALPHA
XEQ DIM?
ALPHA R30 ALPHA
XEQ MATDIM
ALPHA R0,R30 ALPHA
XEQ CLST
XEQ MMOVE

```

Fonctions arithmétiques diverses

Fonction	Résultat
	Maxima et Minima
MAX (maximum)	Trouve l'élément maximum dans la matrice et place dessus le pointeur. Entrée : nom de matrice dans le reg. alpha. Résultat : valeur maximum dans le reg. X.
MIN (minimum)	Trouve l'élément minimum dans la matrice et place dessus le pointeur. Entrée : nom de matrice dans le reg. alpha. Résultat : valeur minimum dans le reg. X.
MAXAB (valeur absolue maximum)	Trouve la valeur absolue maximum dans la matrice et y place le pointeur d'éléments. Entrée : nom de matrice dans le reg. alpha. Résultat : valeur absolue maximum dans le reg. X.
CMAXAB (valeur maximum absolue de la colonne)	Trouve la valeur maximum absolue et y place le pointeur d'éléments. Entrée : nom de matrice dans le reg. alpha. Résultat : valeur absolue maximum dans le reg. X.
RMAXAB (valeur absolue maximum de la ligne)	Trouve la valeur maximum dans la kème ligne et y place le pointeur d'éléments. Entrée : nom de matrice dans le reg. alpha. Résultat : valeur absolue maximum dans le reg. X.
CNRM (norme colonne)	Normes Trouve la somme la plus élevée des valeurs absolues des éléments de chaque colonne et place le pointeur d'éléments sur le premier élément de la colonne ayant la somme la plus élevée. Entrée : nom de matrice dans le reg. alpha. Résultat : norme colonne dans le reg. X.
FNRM (norme de Frobenius)	Trouve la racine carrée de la somme des carrés de tous les éléments de la matrice. Entrée : nom de matrice dans le reg. alpha. Résultat : norme de Frobenius dans le reg. X.
RNRM (norme de ligne)	Trouve la somme la plus élevée des valeurs absolues des éléments de chaque ligne de la matrice et place le pointeur d'éléments sur le premier élément de la ligne ayant la somme la plus élevée. Entrée : nom de matrice dans le reg. alpha. Résultat : norme de ligne dans le reg. X.

Fonctions arithmétiques diverses (suite)

Fonction	Résultat
SUM	Sommes Somme des éléments d'une matrice. Entrée : nom de matrice dans le registre alpha. Résultat : somme dans le registre X.
SUMAB (somme des valeurs absolues)	Fait la somme des valeurs absolues des éléments de la matrice. Entrée : nom de matrice dans le registre alpha. Résultat : somme des val. absolues dans le reg. X.
CSUM (somme des colonnes)	Trouve les sommes de chaque colonne, les stocke dans le vecteur résultat. Entrée : nom de matrice, nom de matrice résultat dans le reg. alpha. Le nombre des éléments dans la matrice résultat doit être égal au nombre de colonnes dans la matrice introduite au départ.
RSUM (somme des lignes)	Fait la somme de chaque rangée et stocke les sommes obtenues dans le vecteur résultat. Entrée : nom de matrice, nom de matrice résultat dans le reg. alpha. Le nombre d'éléments dans la matrice résultat doit être égal au nombre de lignes dans la matrice de départ.
YC+C (Y fois colonne plus colonne)	Multiplie chaque élément de la colonne k de la matrice par la valeur du reg. Y, et l'ajoute à l'élément correspondant de la colonne l , changeant ainsi ses éléments. Ainsi, la commande convertit a_{il} en $a_{il} + y \times a_{ik}$. Entrée : nom de matrice dans le registre alpha. kkk./// dans le registre X. y dans le registre Y.
PIV (pivot)	Trouve la valeur pivot de la colonne k ; c'est-à-dire la valeur absolue maximum d'un élément au-dessous de, ou sous la diagonale. Entrée : nom de matrice dans le registre alpha. kkk dans le registre X. Résultat : valeur pivot dans le reg. X ; le pointeur est placé sur l'élément pivot.
R>R? (compare les lignes)	Compare les éléments des lignes k et l . Si le premier élément non-égal dans k est plus grand que son correspondant dans l , la comparaison est positive pour la règle de programmation «exécution si la condition est vraie». Entrée : nom de matrice dans le registre alpha. kkk./// dans le registre X. Résultat : YES si le premier élément non-égal de la rangée k est plus grand que l'élément de la rangée l . NO dans tous les autres cas.

Fonctions arithmétiques diverses (suite)

Fonction	Résultat
AIP (rappel alph. de la partie entière)	Ajoute la partie entière du nombre dans le registre X au contenu du registre alphabétique. Pour $x < 0$, AIP ajoute la valeur absolue.
MP (rappel alph. de la sollicitation matr.)	Ajoute un message de sollicitation de matrice $rrr.ccc =$ au contenu du registre alphabétique.

Matrices complexes

Dans le calculateur, une matrice complexe est simplement une matrice réelle qui compte quatre fois plus d'éléments. Seul le programme MATRX et l'éditeur de matrices complexes (**CMEDIT**) «reconnaissent» une matrice en tant que matrice complexe et traite ses éléments en conséquence. Toutes les autres fonctions traitent les parties réelles et imaginaires des éléments complexes en tant qu'éléments réels séparés.

Représentation des éléments complexes

Dans sa représentation interne, une matrice complexe a deux fois plus de colonnes et deux fois plus de lignes qu'elle en comporte «normalement».

Le nombre complexe $100 + 200i$ est stocké sous la forme

$$\begin{bmatrix} 100 & -200 \\ 200 & 100 \end{bmatrix}$$

La matrice complexe 2×1

$$\begin{bmatrix} 1 + 2i \\ 3 - 4i \end{bmatrix} \text{ est stocké sous la forme } \begin{bmatrix} 1 & -2 \\ 2 & 1 \\ 3 & 4 \\ -4 & 3 \end{bmatrix}$$

Une importante exception : la matrice colonne (un vecteur) dans un système d'équations simultanées;

Résoudre des équations simultanées complexes. La manière la plus simple de travailler avec des matrices complexes est d'utiliser le programme MATRX. Il dimensionne automatiquement les matrices complexes. Toutefois, **MSYS** peut résoudre des systèmes d'équations plus complexes que MATRX.

En plus, une matrice résultat complexe issue du programme MATRX ne peut être utilisée pour de nombreuses opérations impliquant des matrices complexes *en dehors* de MATRX, et ceci parce que MATRX dimensionnera une matrice colonne complexe différemment de $2m \times 2$. Il utilise les dimensions $2m \times 1$, dans lesquelles les parties réelle et imaginaire d'un nombre deviennent des éléments successifs dans une seule colonne.

Cette forme présente l'avantage d'épargner l'espace mémoire et d'accélérer les opérations. L'éditeur de matrices complexes et **MSYS** peuvent aussi utiliser cette forme $2m \times 1$, bien que ce ne soit pas obligatoire. Ce qui signifie que **MSYS** peut être utilisée sur un système matriciel issu de MATRX.

Une matrice complexe de $2m \times 2$ peut être convertie en la forme $1 \times 2m$ en la transposant, la redimensionnant et en la retransposant ensuite.

Accéder aux éléments complexes. Si l'éditeur de matrices complexes est utilisé (**CMEDIT** ou l'éditeur du programme MATRX), les éléments complexes sont accessibles comme s'ils étaient des nombres complexes. Dans les autres cas (lorsque l'on utilise des fonctions déplaçant le pointeur), on accède aux éléments complexes comme à des éléments réels stockés selon le principe $2m \times 2n$ décrit plus haut.

Espace de stockage en mémoire. Puisque les dimensions nécessaires pour une matrice complexe sont quatre fois plus grandes que le nombre d'éléments complexes (une matrice complexe $m \times n$ étant dimensionnée en tant que $2m \times 2n$), le nombre de registres occupés par une matrice complexe en mémoire est quatre fois plus grand que celui occupé par une matrice réelle ayant le même nombre d'éléments. En d'autres termes, ce sont les dimensions **MATDIM** ou **DIM?** qui sont importantes pour le stockage d'une matrice complexe, non le nombre d'éléments complexes qu'elle comporte.

Fonctions et matrices complexes

La plupart des fonctions ne sont pas très significatives lorsqu'elles sont appliquées aux matrices complexes : puisqu'elles ne reconnaissent pas les différentes parties d'un nombre complexe comme un nombre isolé, les résultats obtenus ne seront pas ceux auxquels on pourrait s'attendre dans le cas d'entrées complexes.

Opérations complexes valides. Certaines fonctions travaillent aussi bien avec des matrices complexes qu'avec des matrices réelles. Ce sont :

MSYS résolution d'équations simultanées

MINV inverse la matrice

MAT+ ajoute la matrice

MAT- soustrait la matrice

MAT* multiplie la matrice par le scalaire, *mais seulement par un scalaire réel en registre X.*

M*M multiplication de matrices

Matrices introduites comme matrices résultat doivent être complexes.

Décomposition LU

La décomposition LU est une forme totalement non-reconnaissable d'une matrice, contenant le plus souvent des données alphabétiques. Cette transposition s'opère lors du processus où l'on trouve :

- La solution à un système d'équations ; (**MSYS** ; **SE** dans le programme MATRX).
- Le déterminant (**MDET** ; **DT** dans le programme MATRX).
- L'inverse (**MINV** ; **I** dans le programme MATRX).

Les deux premières opérations convertissent la matrice introduite dans sa forme décomposée LU et s'arrête là, alors que l'inversion laisse la matrice dans sa forme inversée. Lorsque vous utilisez des fonctions qui laissent la matrice en forme LU décomposée, il est important de noter plusieurs choses :

- Vous ne pouvez éditer (corriger, modifier) une matrice décomposée LU sans en avoir édité chacun des éléments (Cf. «Editer et examiner une matrice décomposée LU», ci-dessous, pour plus de détails).
- Vous ne pouvez effectuer aucune opération qui modifiera la matrice (autre que **MINV**) parce que l'état LU de la matrice sera effacé et elle ne sera plus reconnaissable. Les opérations qui ont cet effet sont : **R<>R** , **C<>C** , **MS** , **MSR+** , **MSC-** , **MSC+** , **MMOVE** (intramatrice), **MSWAP** et **TRNPS** .
- Une matrice décomposée LU doit être examinée avec grand soin. Certaines opérations peuvent modifier des éléments de façon insidieuse (Cf. «Editer et examiner une matrice décomposée LU», ci-dessous).
- La décomposition LU détruit la forme originale de la matrice. Si **MSYS** ou **MDET** sont exécutées, puis que la matrice introduite est examinée (**A** dans le programme MATRX), on ne retrouve plus que la forme décomposée de la matrice.
- On ne peut calculer la transposée (**TRNPS** ; **AB** dans le programme MATRX) d'une matrice en forme décomposée LU. La décomposition LU n'influence pas le calcul correct de l'inverse, de la déterminante ou de la matrice solution, puisque toutes ces opérations nécessitent de toute manière une décomposition LU.

Renverser la décomposition LU. Pour retrouver une matrice à partir de sa forme décomposée LU, *inversez-la deux fois* (en fait : trouvez son inverse puis ré-inversez la en sa forme originale. Pour que ceci fonctionne, il doit évidemment être possible d'inverser la matrice. Le résultat peut différer légèrement de l'original, puisqu'au cours des opérations on «arrondit» certains résultats.

Éditer et visualiser une matrice décomposée LU. Les matrices décomposées LU sont stockées dans une autre forme que les matrices normales :

- Certains éléments contiennent des données alphabétiques.
- Le registre d'état de la matrice est modifié pour indiquer que la matrice est dans sa forme LU.

Le fait d'éditer un *quelconque* élément de la matrice effacera l'indicateur binaire dans le registre d'état, ce qui rendra la matrice non reconnaissable par le programme. A cause de cela, si vous éditez un désélément, il vous faut les éditer *tous* si vous désirez utiliser votre programme une fois encore. Notez que la matrice ne se trouvera plus après cela dans sa forme décomposée LU.

Le contenu d'une matrice décomposée LU peut être examiné ainsi :

- Partant du menu principal, frappez A (Voir A) pour visualiser certains éléments sans les modifier.
- Armez l'indicateur 08 (SF 08) avant d'exécuter **MEDIT** ou **CMEDIT**. Ceci vous permettra d'examiner les éléments sans les modifier.

Exemples

Trouvez le déterminant de l'inverse de la transposée de la matrice ci-dessous :

$$\begin{bmatrix} 6 & 3 & -2 \\ 1 & 4 & -3 \\ 2 & 3 & -1 \end{bmatrix}$$

La taille de mémoire doit être au moins de 10 registres (**SIZE** 010).

54 Fonctions matricielles

Frappez

FIX 4

[XEQ **SIZE** 010

ALPHA **R0** **ALPHA**

3.003 **[XEQ** **MATDIM**

[CF 08

[XEQ **MEDIT**

6 **[R/S**
3 **[R/S**
2 **[CHS** **[R/S**
1 **[R/S**
4 **[R/S**
3 **[CHS** **[R/S**
2 **[R/S**
3 **[R/S**
1 **[CHS** **[R/S**
[XEQ **TRNPS**
[XEQ **MINV**

[XEQ **MDET**

Affichage

3,0030
1:1 = ?
1:2 = ?
1:3 = ?
2:1 = ?
2:2 = ?
2:3 = ?
3:1 = ?
3:2 = ?
3:3 = ?
0,0400

Règle le format d'affichage.

Nomme la matrice **R0**, qui doit être stockée en mémoire principale depuis R_{00} – R_{10} .

Dimensionne **R0** à 3×3 .

Règle l'éditeur pour qu'il permette les modifications.

L'éditeur vous demande de nouveaux éléments, vous montrant d'anciens éléments ou le contenu précédent des registres.

Sort de l'éditeur.

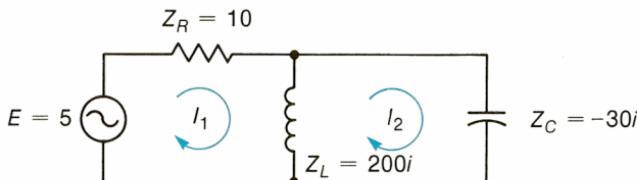
R0 est transposée.

R0 (qui était transposée) est inversée.

Le déterminant de l'inverse de la transposée de la matrice originale.

Notez que si vous désirez trouver la transposée de la matrice originale après avoir trouvé son déterminant, il vous faudra inverser la matrice deux fois pour changer la forme décomposée LU en la matrice d'origine.

Trouvez les courants I_1 et I_2 dans le circuit électrique illustré ci-dessous. Les impédances et les composants sont indiqués sous leur forme complexe.



Ce système peut être représenté par l'équation de matrice complexe

$$\begin{bmatrix} 10 + 200i & -200i \\ -200i & (200 - 30)i \end{bmatrix} \begin{bmatrix} I_1 \\ I_2 \end{bmatrix} = \begin{bmatrix} 5 \\ 0 \end{bmatrix}$$

ou

$$\mathbf{A} \mathbf{X} = \mathbf{B}.$$

La taille de mémoire de stockage doit être réglée à au moins 26 registres (SIZE 026) pour loger deux matrices complexes.

Frappez

ALPHA R ALPHA
4.004 XEQ MATDIM

XEQ CMEDIT

10 R/S 200 R/S
0 R/S 200 CHS R/S
0 R/S 200 CHS R/S
0 R/S 170 R/S

ALPHA R17 ALPHA
4,002 XEQ MATDIM

XEQ CMEDIT

5 R/S 0 R/S
0 R/S 0 R/S

Affichage

4,0040

RE.1:1= ?

RE.1:2= ?

RE.2:1= ?

RE.2:2= ?

-170,0000

4,0020

RE.1:1= ?

RE.2:1= ?
0,0000

Dimensionne la matrice de coefficient complexe $\mathbf{R0}$ à 4×4 pour ses 2 lignes et 2 colonnes. Elle nécessite 17 registres.

Editeur de matrice complexe.

Charge les parties réelle et imaginaire des éléments en $\mathbf{R0}$, la matrice coefficient (A).

Dimensionne la matrice colonne $\mathbf{R17}$ à 4×2 pour 2 lignes complexes et 1 colonne complexe. Elle nécessite 9 registres.

Editeur de matrice complexe.

Charge les parties réelle et imaginaire des éléments dans $\mathbf{R17}$, la matrice colonne (B).

Frappez

ALPHA R, R17 ALPHA
XEQ MSYS

SF 08

ALPHA R17 ALPHA
XEQ CMEDIT
R/S
R/S
R/S

Affichage

0,0000

RE.1:1=0,0372
IM.1:1=0,1311
RE.2:1=0,0437
IM.2:1=0,1543

Calcule la matrice solution (X) et la charge en R17.

Règle l'éditeur en mode «examen».

Affiche les résultats complexes de I_1 et I_2 , qui se trouvent en R17. Si une imprimante est reliée, et si l'indicateur 08 est armé avant d'exécuter CMEDIT, tous les éléments seront imprimés automatiquement.

La solution est

$$\begin{bmatrix} I_1 \\ I_2 \end{bmatrix} = \begin{bmatrix} 0,0372 + 0,1311i \\ 0,0437 + 0,1543i \end{bmatrix}$$

Ce dernier exemple vous demande de résoudre un ensemble de deux équations simultanées avec deux variables inconnues. Ceci demande l'utilisation de MSYS.

Syphorien Duresson se trouve avoir les résultats de ventes d'épinards et de choux suivants, répartis sur trois semaines différentes. Il connaît le poids total de légumes vendus chaque semaine, le prix touché chaque semaine et le prix au kilo de chaque récolte. Trouvez les poids de choux et d'épinards vendus chaque semaine...

	1 ^{ère} semaine	2 ^{ème} semaine	3 ^{ème} semaine
Poids total (kg)	274	233	331
Valeur totale	120,32 F	112,96 F	151,36 F

Le prix du chou est de 0,24 F/kg et le prix des épinards, 0,86 F/kg.

La liste suivante d'équations linéaires décrit les deux inconnues (les prix des choux et des épinards) pour les trois semaines, où la première ligne de la matrice constante représente le poids de choux pour les trois semaines et la deuxième ligne, le poids d'épinards. Puisque la matrice constante n'est pas une matrice colonne, il vous faut utiliser MSYS et non la fonction SE dans le programme MATRX.

$$\begin{bmatrix} 1 & 1 \\ 0,24 & 0,86 \end{bmatrix} \begin{bmatrix} d_{11} & d_{12} & d_{13} \\ d_{21} & d_{22} & d_{23} \end{bmatrix} = \begin{bmatrix} 274 & 233 & 331 \\ 120,32 & 112,96 & 151,36 \end{bmatrix}$$

La taille mémoire doit être réglée à 12 registres minimum (**SIZE** 012) pour loger ces deux matrices réelles.

Frappes

ALPHA R ALPHA
2.002 **XEQ MATDIM**

ALPHA R5 ALPHA
2.003 **XEQ MATDIM**

CF 08

XEQ MEDIT

274 **R/S** 233 **R/S**
331 **R/S** 120.32 **R/S**
112.96 **R/S** 151.36 **R/S**

ALPHA R ALPHA
XEQ MEDIT
1 **R/S** 1 **R/S**
0.24 **R/S** 0.86 **R/S**
ALPHA R, R5 ALPHA

XEQ MSYS

SF 08

ALPHA R5 ALPHA
XEQ MEDIT
R/S
R/S
R/S
R/S
R/S
R/S

Affichage

2,0020

2,0030

1:1= ?

1:3= ?

2:2= ?

3,0010

1:1= ?

2:1= ?

3.0010

1:1= ?

2:1= ?

3.0010

1:1 = 186,0000

1:2 = 141,0000

1:3 = 215,0000

2:1 = 88,0000

2:2 = 92,0000

2:3 = 116,0000

3,0010

Dimensionne la matrice coefficient **R0** à 2×2 .

Dimensionne la matrice constante **R5** à 2×3 .

Règle l'éditeur pour permettre l'édition.

Appelle l'éditeur de matrice pour la matrice en cours, c.à.d. **R5**.

Charge **R5**, la matrice constante.

L'éditeur pour **R0**. Charge **R0**, la matrice coefficient.

Spécifie les matrices introduites (coefficient, constante). La solution ira dans **R5**.

Calcule la matrice solution.

Règle l'éditeur en mode «visualisation».

Affiche le résultat dans la matrice solution.

La solution est

	1 ^{ère} semaine	2 ^{ème} semaine	3 ^{ème} semaine
Choux (kg)	186	141	215
Epinards (kg)	88	92	116

Tableau alphabétique des fonctions

Sans autre directive, chaque fonction opère sur la matrice (ou les matrices) nommées dans le registre alphabétique. Lorsque le registre alphabétique est effacé, la fonction opère sur la matrice en cours.

Fonctions matricielles

Fonction	Description
AIP (p. 50)	Ajoute la partie intégrale de x au registre alphabétique.
C<>C (p. 46)	Echange les colonnes k et l .
CMAXAB (p. 48)	Donne la valeur absolue minimum dans la k ^{ème} colonne.
CMEDIT (p. 34)	Invoque l'éditeur de matrices complexes.
CNRM (p. 48)	Donne la norme colonne.
CSUM (p. 49)	Trouve les sommes de colonnes et les place dans une matrice ligne.
DIM? (p. 33)	Donne la dimension $mmm.nnn$.
FNRM (p. 48)	Donne la norme de Frobenius.
I+ (p. 40)	Augmente la partie «ligne» du pointeur.
I- (p. 40)	Diminue la partie «ligne» du pointeur.
J+ (p. 40)	Augmente la partie «colonne» du pointeur.
J- (p. 40)	Diminue la partie «colonne» du pointeur.
M*M (p. 44)	Multiplication vraie (non scalaire) de deux matrices.
MAT+ (p. 43)	Ajoute scalaires ou matrices à une matrice.
MAT- (p. 43)	Soustrait scalaires ou matrices d'une matrice.
MAT* (p. 44)	Multiplie un scalaire ou une matrice par une matrice élément par élément.
MAT/ (p. 44)	Divise une matrice par un scalaire ou une matrice élément par élément.
MATDIM (p. 31)	Dimensionne une matrice à $mmm.nnn$.
MAX (p. 48)	Donne l'élément maximum.

Fonctions matricielles (suite)

Fonction	Description
MAXAB (p. 48)	Donne la valeur maximum absolue d'un élément.
MDET (p. 45)	Donne le déterminant.
MEDIT (p. 34)	Invoque l'éditeur de matrices réelles.
MIN (p. 48)	Donne l'élément minimum.
MINV (p. 45)	Inverse la matrice en place.
MMOVE (p. 46)	Copie la matrice source ou la sous-matrice dans la matrice destination.
MNAME? (p. 37)	Donne le nom de la matrice en cours dans le registre alphabétique.
MP (p. 50)	Aj. une sollicitation <i>rrr:ccc</i> = au reg. alpha.
MR (p. 40)	Rappelle l'élément en cours.
MRC+ (p. 41)	Rappelle séquentiellement les éléments par colonne.
MRC- (p. 41)	Rappelle séquentiellement les éléments en ordre récurrent par colonne.
MRIJ (p. 40)	Rappelle le pointeur <i>iii.jjj</i> de la matrice en cours.
MRIJA (p. 40)	Rappelle le pointeur <i>iii.jjj</i> .
MRR+ (p. 41)	Rappelle les éléments séquentiels par ligne.
MRR- (p. 41)	Rappelle les éléments séquentiels de manière récurrente par ligne.
MS (p. 40)	Stocke l'élément en cours.
MSC+ (p. 41)	Stocke l'élément en cours par colonne.
MSIJ (p. 40)	Règle le pointeur de la matrice en cours à <i>iii.jjj</i> .
MSIJA (p. 40)	Règle le pointeur à <i>iii.jjj</i> .
MSR+ (p. 41)	Stocke l'élément courant par ligne.
MSWAP (p. 46)	Echange deux matrices ou sous-matrices.
MSYS (p. 45)	Résout un système d'équations simultanées.
PIV (p. 49)	Donne la valeur absolue maximum d'une colonne se trouvant sur ou sous la diagonale.
R<>R (p. 46)	Echange les lignes <i>k</i> et <i>l</i> .
R>R? (p. 49)	Teste élément par élément si la <i>k</i> est plus grande que <i>l</i> .
RMAXAB (p. 48)	Donne la valeur absolue maximum de la ligne <i>k</i> .
RNRM (p. 48)	Donne la norme ligne.
RSUM (p. 49)	Trouve la somme des lignes et la place dans la matrice colonne.
SUM (p. 49)	Donne la somme de tous les éléments.

Fonctions matricielles (Suite)

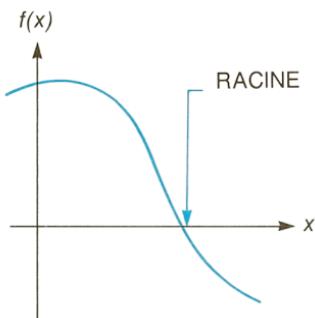
Fonction	Description
SUMAB (p. 49)	Donne la somme des valeurs absolues de tous les éléments.
TRNPS (p. 45)	Transpose la matrice en place.
YC+C (p. 49)	Multiplie chaque élément de la colonne k par la valeur y et ajoute le produit à l'élément de la colonne l , remplaçant ce dernier.

RESOLUTION D'UNE EQUATION

Le programme **SOLVE** permet de trouver les racines d'une équation de la forme

$$f(x) = 0,$$

où x représente une *racine réelle*.*

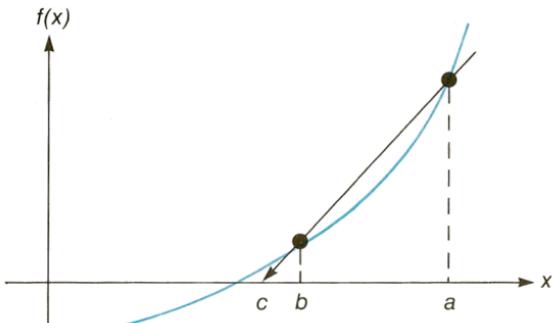


L'exécution du programme **SOLVE** (**SOLVE**) fait appel à une technique numérique perfectionnée permettant de trouver les racines réelles de nombreux types d'équations. Vous fournissez l'équation de la fonction (dans un programme), ainsi que deux estimations initiales, et **SOLVE** fait le reste.

Principe de fonctionnement

SOLVE utilise normalement la méthode des sécantes pour trouver itérativement et tester des valeurs de x comme racines potentielles. Il faut au programme entre plusieurs secondes et plusieurs minutes pour parvenir à un résultat.

* Remarquez que toute équation à une variable peut être exprimée sous cette forme. Par exemple, $f(x) = a$ équivaut à $f(x) - a = 0$, et $f(x) = g(x)$ équivaut à $f(x) - g(x) = 0$.



Si c n'est pas une racine mais si $f(c)$ est plus proche de zéro que $f(b)$, b remplace alors a , c remplace b , et le processus d'estimation se répète. Si le graphe de $f(x)$ est régulier et si les valeurs initiales de a et b sont proches d'une racine simple, cette méthode converge rapidement vers une racine.

Si la sécante calculée est presque horizontale, SOLVE modifie la méthode des sécantes pour garantir que $|c - b| \leq 100 |a - b|$ (particulièrement important car cela réduit la tendance de la méthode des sécantes à s'égarer lorsque l'erreur d'arrondi devient importante à proximité d'une racine).

Si SOLVE a déjà trouvé des valeurs a et b telles que $f(a)$ et $f(b)$ sont de signes opposés, il modifie la méthode des sécantes pour que c se trouve toujours à l'intérieur de l'intervalle dans lequel se produit le changement de signe. Ceci garantit que l'intervalle de recherche diminue à chaque itération, et qu'une racine est finalement trouvée.

Si cette méthode ne fournit pas une racine, SOLVE ajuste une parabole passant par les valeurs de la fonction aux points a , b , et c , et trouve la valeur d au maximum ou au minimum de la parabole. La recherche se poursuit ensuite selon la méthode des sécantes, d venant alors remplacer a .

Si trois ajustements successifs de paraboles ne fournissent pas de racine, ou si $d = b$, le calculateur affiche **NO** (non). Les registres X et Z contiennent alors respectivement b et $f(b)$, et le registre Y contient a ou c . A ce stade, vous pouvez : reprendre la recherche là où elle s'est arrêtée, reprendre la recherche en un autre point, décider que $f(b)$ est négligeable et que $x = b$ est une racine, transformer l'équation en une autre équation plus facile à résoudre, ou conclure qu'il n'existe pas de racine.

Utilisation

Lors du calcul des racines, la fonction SOLVE appelle et exécute de manière répétitive un sous-programme d'évaluation de $f(x)$ que *vous devez écrire*. Vous devez aussi fournir à SOLVE deux estimations initiales de x , définissant ainsi un *intervalle* permettant à la fonction de commencer sa recherche des racines.

Vous augmenterez grandement la rapidité et la précision du calcul en fournissant d'entrée des estimations réalistes. Si la variable x ne constitue une solution réaliste et significative que dans un intervalle limité, il est raisonnable de choisir vos deux estimations initiales dans cet intervalle (des racines négatives, par exemple, n'ont souvent pas de sens pour des problèmes réels, physiques).

- SOLVE doit pouvoir disposer de treize registres de programme inutilisés. Si un nombre suffisant de registres de programme n'est pas disponible, SOLVE ne pourra pas être exécutée et il en résultera l'erreur **NO ROOM** (pas assez d'espace). Exécutez **GTO** \square \square en mode Programme pour voir combien il reste de registres de programme disponibles.
- Avant d'exécuter SOLVE, vous devez avoir un programme (stocké en mémoire programme ou dans un module enfichable) qui évalue votre fonction $f(x)$ à zéro. Ce programme doit être nommé avec un *label global**. SOLVE appelle alors répétitivement votre programme pour calculer successivement des approximations de plus en plus précises de x . Votre programme peut tirer parti du fait que la fonction SOLVE remplit la pile opérationnelle du calculateur avec son approximation courante de x chaque fois qu'elle appelle votre programme.
- Vous introduisez ensuite vos deux estimations initiales de la racine, a et b , dans les registres X et Y.
- Enfin, vous introduisez le nom de votre programme (qui évalue la fonction) dans le registre alphabétique, puis vous exécutez **SOLVE**.

* Ce programme *ne doit pas* inclure les fonctions **PASN**, **PSIZE**, **AK**, toute fonction du lecteur de carte (HP 82104A), ou toute autre fonction modifiant la configuration de la mémoire du calculateur, les fonctions affectées aux touches ou les alarmes.

64 Résolution d'une équation

Lorsque le programme s'arrête et que le calculateur affiche un nombre, le contenu de la pile opérationnelle est :

Z = valeur de la fonction pour $x =$ racine (devrait être zéro).*

Y = approximation précédente de la racine (devrait être proche de la racine définitive).

X = la racine (la valeur qui est affichée).

Si la fonction que vous analysez est égale à zéro pour plus d'une valeur de x , **SOLVE** s'arrête lorsqu'elle trouve l'une quelconque de ces valeurs. Pour trouver d'autres valeurs, introduisez de nouvelles estimations initiales et exécutez à nouveau **SOLVE**.

Instructions d'utilisation de **SOLVE**

Instructions	Fappez :	Taille : 013 Affichage
1. Passez en mode Programme et compactez la mémoire avant d'introduire un nouveau programme (l'affichage vous indiquera le nombre de registres de programme disponibles). 2. Frappez un label global alphabétique qui servira de nom au programme décrivant $f(x)$ pour $f(x) = 0$. 3. Frappez les lignes du programme et mettez fin au programme avec une instruction RTN . 4. Vérifiez que la mémoire programme est assez grande pour permettre l'exécution de SOLVE ($nnn \geq 13$).* Quittez ensuite le mode Programme. 5. Placez le nom de votre programme (point 2 ci-dessus) dans le registre alphabétique. 6. Introduisez l'intervalle de recherche pour x :	PRGM GTO [] [] LBL label global : RTN GTO [] [] PRGM ALPHA label global ALPHA a ENTER ↴ b	00 REG nnn 01 LBL Tlabel 00 REG nnn a b

* Si le contenu du registre Z n'est pas zéro, le registre X ne contient pas la racine exacte. Les contenus de X et de Y sont alors des estimations proches de la racine, encadrant un changement de signe de la valeur de la fonction.

Instructions d'utilisation de SOLVE (suite)

Instructions	Frappez :	Affichage
7. Exécutez SOLVE L'exécution du programme peut prendre jusqu'à plusieurs minutes, puis la racine obtenue est affichée. Si aucune racine n'a été trouvée, le calculateur affiche NO (non).	SOLVE †	X
8. Pour chercher une autre racine, répétez les points 6 et 7 ci-dessus.		
* Si nnn n'est pas ≥ 13 , utilisez SIZE pour affecter plus de mémoire aux registres de programme, ou supprimez des programmes. Consultez à ce sujet le manuel d'utilisation du HP-41.		
† Pour exécuter un programme, appuyez sur XEQ ALPHA nom alphabétique du programme ALPHA ou redéfinissez une touche sur le clavier personnel (USER) en lui affectant cette fonction.		

Remarques

Une pression sur **R/S** met fin au programme **SOLVE**.

Exemples

Trouvez les racines de l'équation $f(x) = x^2 - 3x - 10 = 0$.

Commencez par écrire un programme appelé **TEST** pour définir la fonction. Puis, avant d'exécuter **SOLVE**, frappez le nom de ce programme dans le registre alphabétique (**ALPHA**) et introduisez vos estimations initiales de la racine.

En utilisant la méthode de Horner, vous pouvez ré-écrire $f(x)$ de manière qu'elle soit plus facilement programmable : $f(x) = (x - 3)x - 10$ (remarquez que vous pouvez aussi trouver cette racine par le calcul algébrique). Comme le programme **SOLVE** remplit la pile opérationnelle du calculateur avec l'approximation courante de x avant d'appeler **TEST**, ce dernier peut tirer x de la pile au cours de son exécution.

Frappez

FIX 4

PRGM **GTO** **•** **•**

LBL **ALPHA** **TEST**

ALPHA

3

Affichage

00 REG *nnn*

01 LBL^T**TEST**

02 3—

Valide le format d'affichage utilisé dans cet exemple.

Mode Programme ; prêt pour l'introduction d'un programme définissant $(x - 3)x - 10$.

Label global alphabétique «TEST».

66 Résolution d'une équation

Frappez

[-]
[x]
10
[-]
[RTN]

[GTO] [.] [.]

[PRGM]

[ALPHA] TEST
[ALPHA]

0 [ENTER↑] 10

[XEQ] [SOLVE]

Affichage

03 -
04 *
05 10_-
06 -
07 RTN

00 REG nnn

TEST_-

10_-

5,0000

$(x - 3)$
 $(x - 3)x$
 $(x - 3)x - 10$
Fin du programme
définissant $f(x)$.

Nombre de registres
de programme
disponibles (devrait
être ≥ 13).

Sortie du mode
Programme.

Place «TEST» (nom de
votre programme)
dans le registre alpha-
bétique (première
étape nécessaire à
l'exécution de SOLVE).

Saisie de zéro et dix
comme estimations
initiales. Vous êtes
maintenant prêt à exé-
cuter [SOLVE].

Exécute le programme
SOLVE ; trouve une
racine $x = 5,0000$ (en
12 secondes environ).

Assurez-vous que 5,0000 est bien une racine de $f(x) = 0$ en vérifiant le
registre Z. Vérifiez ensuite s'il existe une seconde racine (ce qui est
courant pour les équations du second degré) en introduisant de nouvelles
estimations initiales 0 et -10, pour trouver une éventuelle racine
négative.

Frappez

[R↓] [R↓]

0 [ENTER↑] 10 [CHS]

[XEQ] [SOLVE]

[R↓] [R↓]

Affichage

0,0000

-10_-

-2,0000

0,0000

Affiche le registre Y,
puis le registre Z.
Comme $f(5) = 0$, 5 est
une bonne racine.

Nouvelles estimations
initiales pour chercher
une éventuelle
seconde racine.

Seconde racine.

Cette racine est égale-
ment valable.

Voici maintenant un problème où les racines ne peuvent pas être trouvées par l'algèbre. Si le champion Chuck Fahr lance un marteau verticalement à une vitesse de 50 mètres/seconde, combien de temps faut-il au marteau pour retomber au sol ? Calculez t dans l'équation

$$h = 5000 (1 - e^{-t/20}) - 200t$$

Supposons que h est en mètres et t en secondes. Naturellement, nous ne sommes intéressés que par une racine t positive.

Comme précédemment, le programme définissant la fonction peut tirer parti du fait que la pile opérationnelle contient l'approximation courante de t au moment où vous appelez votre programme.

Frappez

PRGM GTO . .
LBL ALPHA HAUT
ALPHA

20 CHS

÷

e^x

CHS

1

+

5000

×

$\times\text{S}y$

200

×

−

RTN

GTO . .

PRGM

ALPHA HAUT ALPHA

5 [ENTER↑] 6

XEQ SOLVE

R↓ R↓

Affichage

00 REG nnn

01 LBL^THAUT

02 −20_

03 /

04 E↑X

05 CHS

06 1_

07 +

08 5000_

09 *

10 X<>Y

11 200_

12 *

13 −

14 RTN

00 REG nnn

Donne au programme le label global «HAUT».

$-t/20$

$-e^{-t/20}$

$1 - e^{-t/20}$

$5000 (1 - e^{-t/20})$

$200t$

L'équation est complète, et le programme est donc le suivant :
 $5000(1 - e^{-t/20}) - 200t$

Est-ce que $nnn \geq 13$?

Sortie du mode Programme.

Place le nom de votre programme dans le registre alphabétique.

Exemple d'estimations initiales pour t .

Racine $t = 9,2843$ s.

Montre que $h(9,2843) = 0$.

6_

9,2843

0,0000

Lorsqu'il n'existe pas de racine

Il est possible qu'une équation n'ait pas de racine réelle. Dans ce cas, le calculateur affiche **NO** (non) au lieu d'un résultat numérique. Cela se produirait par exemple si vous essayez de résoudre l'équation

$$|x| = -1$$

qui n'a pas de solution puisque la fonction de valeur absolue n'est jamais négative.

Trois types d'erreur peuvent arrêter l'exécution de **SOLVE** :

- Si des itérations successives dans la recherche de la racine donnent une valeur constante différente de zéro pour la fonction spécifiée, le calculateur affiche **NO** (non).
- Si de nombreuses approximations indiquent que la *grandeur* de la fonction semble présenter une valeur minimale différente de zéro dans l'intervalle étudié, le calculateur affiche **NO** (non).
- Si vous utilisez un argument incorrect pour une opération mathématique de votre programme, le calculateur affiche **DATA ERROR** (erreur de donnée).

Informations pour la programmation

Vous pouvez incorporer **SOLVE** dans un programme plus vaste que vous avez créé. Assurez-vous dans ce cas que le programme place les estimations initiales dans les registres X et Y juste avant l'exécution de **SOLVE**. N'oubliez pas non plus que **SOLVE** cherchera dans le registre alphabétique le nom du sous-programme qui calcule votre fonction.

Si l'exécution de **SOLVE** dans votre programme donne une racine, votre programme passera à l'exécution de la ligne suivante. Si *aucune* racine n'est trouvée, la ligne suivante sera sautée (c'est la règle qui régit la programmation du HP-41). Vous pouvez donc écrire votre programme de façon à tenir compte du cas où **MOVE** ne trouve pas de racine, par exemple en choisissant de nouvelles estimations initiales ou en modifiant un paramètre de la fonction.

L'utilisation de **SOLVE** dans un programme fait appel à l'un des six niveaux d'attente de retour de branchement dont le calculateur est doté, ce qui laisse cinq niveaux d'attente pour le sous-programme qui appelle **SOLVE**.

Remarquez que **SOLVE** ne peut pas être utilisé de manière récursive (en se rappelant lui-même). Si cela se produit, le programme s'arrête et affiche **RECURSION**. Vous pouvez utiliser **SOLVE** avec **INTEG**, le programme d'intégration.

Références

«Utilisation efficace de SOLVE», *Manuel des fonctions de haut niveau du HP-15C*, Hewlett-Packard Co., 1982.

Kahan, W.M., «Personal Calculator Has Key to Solve Any Equation $f(x)=0$ », *Hewlett-Packard Journal*, 30:12, December 1979.

RESOLUTION D'UNE EQUATION POLYNOMIALE

Le programme PLY permet de trouver les racines d'une équation polynomiale à coefficients réels de degré inférieur ou égal à 5, ou d'évaluer un polynôme de degré inférieur ou égal à 20.

L'équation polynomiale peut être représentée sous la forme :

$$a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0 = 0,$$

avec $n = 2, 3, 4$, ou 5 .

Il est également possible d'évaluer des polynômes pour des valeurs arbitraires de x . Ceci est utile pour tracer des polynômes et utiliser des corrélations de données basées sur des polynômes.

Lorsqu'il lance le programme, l'utilisateur doit spécifier le degré (n) du polynôme. Le programme demande ensuite les coefficients un par un a_n, \dots, a_1, a_0 . Il faut introduire un zéro pour les coefficients nuls. Ce sont les registres 00 à 05 qui servent à stocker les coefficients (dans le cas de l'évaluation d'un polynôme de degré inférieur ou égal à 20, les coefficients sont stockés dans les registres 00 à 20).

Equations utilisées par l'algorithme

Dans le calcul des racines, la première étape du programme consiste à diviser tous les coefficients par a_n pour obtenir une équation de la forme $x^n + a'_{n-1} x^{n-1} + \dots + a'_0 = 0$. Le diviseur est conservé dans le registre a_n en vue de l'évaluation du polynôme pour des valeurs arbitraires de x .

Les programmes pour les équations du troisième et du cinquième degré utilisent un algorithme itératif pour trouver une racine réelle de l'équation. Ces programmes requièrent que le terme constant a_0 ne soit pas nul pour ces équations (si $a_0 = 0$, zéro est alors une racine réelle, et on peut réduire l'équation d'un degré par mise en facteur de x). Dès qu'une racine est trouvée, une division polynomiale est effectuée pour réduire l'équation initiale à une équation du second ou quatrième degré.

Pour résoudre une équation du quatrième degré, il faut d'abord résoudre l'équation du troisième degré suivante :

$$y^3 + b_2 y^2 + b_1 y + b_0 = 0,$$

$$\text{avec } b_2 = -a_2$$

$$b_1 = a_3 a_1 - 4a_0$$

$$b_0 = a_0 (4a_2 - a_3^2) - a_1^2.$$

72 Résolution d'une équation polynomiale

Soit y_0 , la plus grande racine réelle de l'équation précédente.

L'équation du quatrième degré est alors ramenée aux deux équations du second degré suivantes :

$$x^2 + (A + C)x + (B + D) = 0$$

$$x^2 + (A - C)x + (B - D) = 0$$

avec $A = \frac{a_3}{2}$, $B = \frac{y_0}{2}$, $D = \sqrt{B^2 - a_0}$, $C = \sqrt{A^2 - a_2 + y_0}$

Les racines de l'équation du quatrième degré sont obtenues par résolution des deux équations du second degré.

Une équation du second degré $x^2 + a_1x + a_0 = 0$ admet pour racine

$$x_{1,2} = -\frac{a_1}{2} \pm \sqrt{\frac{a_1^2}{4} - a_0}.$$

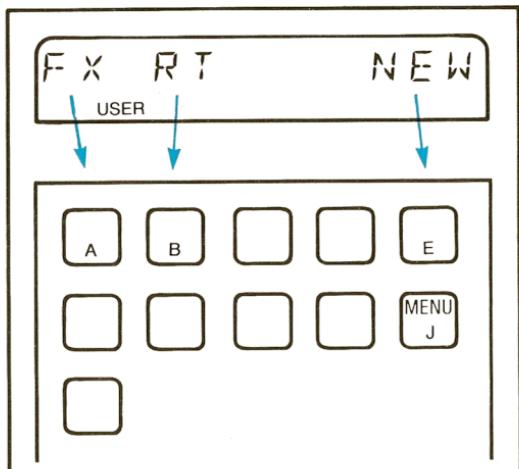
Si $D = a_1^2/4 - a_0 > 0$, les racines sont réelles ; si $D < 0$, les racines sont complexes, avec $u \pm iv = -(a_1/2) \pm i\sqrt{-D}$.

Une racine réelle est affichée sous la forme d'une nombre unique. En revanche, les racines complexes apparaissent toujours par paires de la forme $u \pm iv$, et elles sont identifiées à l'affichage.

Il faut s'attendre à des temps d'exécution assez longs pour les équations de degré 3, 4, ou 5, du fait que le programme utilise une méthode itérative une ou plusieurs fois.

Utilisation

Une fois que vous avez introduit vos variables, le menu affiché par ce programme montre quelles touches correspondent à quelles fonctions dans PLY. Vous pouvez appuyer sur **J** pour afficher à nouveau ce menu à n'importe quel moment. Cela ne perturbera pas le déroulement du programme.



FX = Evaluate le polynôme pour x

RT = Root
Calcule les racines

NEW = Permet de choisir un nouveau polynôme

Pour effacer le menu, appuyez sur \leftarrow . Ceci affiche le contenu du registre X, sans mettre fin au programme. Vous pouvez effectuer des calculs, puis continuer le programme en appuyant sur **R/S** (il n'est toutefois pas nécessaire d'effacer l'affichage issu de l'exécution du programme avant d'effectuer des calculs).

Instructions d'utilisation de PLY

Instructions	Frappez :	Taille : 023
<ol style="list-style-type: none"> 1. Lancez le programme PLY. 2. Frappez le degré du polynôme ($n = 2, 3, 4, 5$ pour trouver la racine ; jusqu'à 20 pour une évaluation seulement). 3. Introduisez le coefficient a_n du polynôme (il faut introduire 0 pour les coefficients nuls). Répétez cette opération jusqu'à ce que l'affichage demande a_0. 4. Introduisez le coefficient a_0. 5. Pour évaluer le polynôme pour x, utilisez FX. Vous pouvez répéter cette étape pour de nouvelles valeurs de x. 	<p>XEQ PLY *</p> <p>n R/S</p> <p>a_n R/S ⋮ a_1</p> <p>a_0 R/S</p> <p>x A (FX) R/S †</p>	<p>DEGREE=?</p> <p>an=?</p> <p>a(n - 1)=? ⋮ a0=?</p> <p>FX RT NEW</p> <p>F<X>=f(x) FX RT NEW</p>

Instructions d'utilisation de PLY (suite)

Instructions	Fappez :	Affichage
6. Pour trouver les racines du polynôme, utilisez RT puis R/S pour afficher les racines successives.	B (RT) R/S ⋮ R/S R/S R/S R/S R/S † E (NEW)	ROOT=racine 1 ROOT=racine 2 ⋮ U=valeur de u V=valeur de v U=valeur de u V=valeur de -v FX RT NEW DEGREE=?
7. Pour travailler sur un nouveau polynôme, choisissez NEW (E) and retournez à l'étape 2.		
* Pour exécuter un programme, frappez XEQ ALPHA nom alphabétique du programme ALPHA ou redéfinissez une touche du clavier personnel (USER) en lui affectant cette fonction.		
† Une pression sur cette touche n'est pas nécessaire si une imprimante est connectée à votre calculateur, car l'imprimante imprime automatiquement les résultats puis affiche le menu.		

Remarque : Ce programme peut calculer des racines incorrectes par suite de l'arrondi des résultats intermédiaires. Ceci n'affecte normalement que les racines *réelles*. Pour vérifier la racine calculée, ré-exécutez PLY pour évaluer un polynôme (étape 5). Introduisez la racine *x* que vous voulez vérifier. Si le résultat est un très petit nombre proche de zéro, la racine est correcte.

Remarques

Si vous armez l'indicateur binaire 06 (**SF** 06) juste avant l'étape 6, les valeurs obtenues lors de cette étape seront stockées telles qu'elles ont été trouvées dans les registres R_{24} et suivants, et dans l'ordre racines réelles d'abord, puis racines imaginaires (les racines réelles stockent une partie imaginaire nulle).

Ce programme affecte des labels locaux alphabétiques (cf. manuel d'utilisation du HP-41) aux touches **A**, **B**, **E**, et **J**. Ces affectations locales n'ont *pas la priorité* sur les affectations que vous avez éventuellement déjà attribuées aux touches du clavier personnel (USER), et le programme ne peut pas fonctionner dans ce cas. *Par conséquent, avant de lancer le programme, n'oubliez pas d'annuler toute redéfinition que vous auriez déjà affectée aux touches du clavier personnel qui sont également utilisées par ce programme*, et évitez de redéfinir ces touches à l'avenir.

Exemples

Trouvez les racines de $x^5 - x^4 - 101x^3 + 101x^2 + 100x - 100 = 0$.

Frappez**FIX** 4**XEQ** **SIZE** 023**XEQ** **PLY**

5 **R/S**
 1 **R/S**
 1 **CHS** **R/S**
 101 **CHS** **R/S**
 101 **R/S**
 100 **R/S**
 100 **CHS** **R/S**
B **(RT)**
R/S
R/S
R/S
R/S

Affichage

DEGREE=?
a5=?
a4=?
a3=?
a2=?
a1=?
a0=?
FX RT NEW
ROOT=10,0000
ROOT=1,0000
ROOT=1,0000
ROOT=-1,0000
ROOT=-10,0000

Valide le format d'affichage utilisé.

Optionnel—définit le nombre de registres de stockage nécessaires au programme (inutile si l'affectation mémoire est déjà $\text{SIZE} \geq 023$).Résolvez $4x^4 - 8x^3 - 13x^2 - 10x + 22 = 0$.**Frappez****J****E** **(NEW)**

4 **R/S**
 4 **R/S**
 8 **CHS** **R/S**
 13 **CHS** **R/S**
 10 **CHS** **R/S**
 22 **R/S**
B **(RT)**
R/S
R/S
R/S
R/S
R/S

Affichage**FX RT NEW****DEGREE=?**

a4=?
a3=?
a2=?
a1=?
a0=?
FX RT NEW
U=-1,0000
V=1,0000
U=-1,0000
V=-1,0000
ROOT=3,1180
ROOT=0,8820

Affiche le menu (optionnel).

Le calculateur demande un nouveau polynôme (après celui de l'exemple 1).

Affiche le menu.

Les racines 1 et 2 sont $-1,00 \pm 1,00i$.

Racine 3.

Racine 4.

76 Résolution d'une équation polynomiale

Evaluez le polynôme suivant pour $x = 2,5$ et $x = -5$.

$$f(x) = x^5 + 5x^4 - 3x^2 - 7x + 11$$

Frappez

J

E (NEW)

5 [R/S]
1 [R/S]
5 [R/S]
0 [R/S]
3 [CHS] [R/S]
7 [CHS] [R/S]
11 [R/S]
2.5 [A] (FX)
5 [CHS] [R/S]

Affichage

FX RT NEW

DEGREE=?

a5=?
a4=?
a3=?
a2=?
a1=?
a0=?
FX RT NEW
 $F<X> = 267,7188$
 $F<X> = -29,0000$

Affiche le menu (optionnel).

Le calculateur demande un nouveau polynôme.

Informations pour la programmation

Vous pouvez utiliser le sous-programme RTS dans vos propres programmes. Il permet de trouver les racines réelles et complexes d'un polynôme de degré inférieur ou égal à 5.

Taille mémoire minimale pour l'exécution de RTS : SIZE 023, à moins que l'indicateur binaire 6 ne soit armé, auquel cas le nombre de registres de stockage de données requis est $24 + 2 \times (\text{degré})$.

Indicateurs binaires utilisés : 00, 03, 05, 06, 21

Sous-programme : RTS

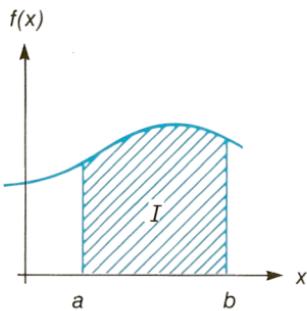
Registres initiaux	Registres finals	Indicateurs binaires à modifier
$R_{00} = a_0$	$R_{00} = a_0/a_5$	SF 00
$R_{01} = a_1$	$R_{01} = a_1/a_5$	CF 03
$R_{02} = a_2$	$R_{02} = a_2/a_5$	CF 05
$R_{03} = a_3$	$R_{03} = a_3/a_5$	SF 06 si vous voulez stocker les racines
$R_{04} = a_4$	$R_{04} = a_4/a_5$	CF 06 si vous ne voulez pas stocker les racines
$R_{05} = a_5$	$R_{05} = a_5$	SF 21 pour que le programme s'interrompe lorsque des résultats sont affichés
	$R_{06} \dots R_{21} =$ espace de travail du programme	CF 21 pour que le programme ne s'interrompe pas lorsque des résultats sont affichés
$R_{22} =$ degré de l'équation	$R_{22} =$ degré de l'équation	
$R_{23} =$ pointeur		
Si l'indicateur binaire 06 est armé :	$R_{24}, R_{25} =$ racine 1 $R_{26}, R_{27} =$ racine 2 $R_{28}, R_{29} =$ racine 3 $R_{30}, R_{31} =$ racine 4 $R_{32}, R_{33} =$ racine 5	

Commentaires. Pour utiliser RTS, introduisez les coefficients dans $R_{00}\dots R_{05}$, le degré dans R_{22} , armez l'indicateur binaire 06 pour stocker les racines, désarmez les indicateurs binaires 03 et 05, et armez l'indicateur binaire 00. Si les racines sont stockées, elles le sont avec leurs parties réelles et imaginaires ; une racine réelle a une partie imaginaire nulle.

INTEGRATION NUMERIQUE

Le programme INTEG calcule l'intégrale définie, I , d'une fonction $f(x)$ sur un intervalle d'intégration limité par a et b . Ceci s'exprime mathématiquement et graphiquement sous la forme

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



L'exécution du programme INTEG ([INTEG](#)) fait appel à une technique numérique élaborée de calcul de l'intégrale définie d'une fonction. Vous définissez l'équation de la fonction (dans un programme) et l'intervalle d'intégration, et INTEG fait le reste.

Principe de fonctionnement

L'algorithme de INTEG utilise une méthode de Romberg de cumul de la valeur d'une intégrale. L'algorithme évalue $f(x)$ pour plusieurs valeurs de x (appelés échantillons) entre les limites de l'intervalle d'intégration. Il faut au programme entre plusieurs secondes et plusieurs minutes pour obtenir un résultat.

Plusieurs améliorations rendent l'algorithme plus efficace. Par exemple, au lieu d'utiliser des échantillons uniformément répartis, ce qui peut induire une sorte de résonance et produire des résultats trompeurs lorsque la fonction à intégrer est périodique, INTEG utilise des échantillons répartis de manière non-uniforme. Autre amélioration : INTEG utilise une précision accrue (13 chiffres significatifs) pour le cumul des sommes internes, ce qui permet au calculateur de cumuler avec précision des milliers d'échantillons, si nécessaire.

Un calculateur utilisant l'intégration numérique ne peut presque jamais calculer une intégrale avec précision. Cependant, vous disposez d'un moyen pratique de spécifier quelle marge d'erreur est tolérable. Vous pouvez en effet choisir le format d'affichage en fonction du nombre de chiffres requis pour la précision de la fonction à intégrer $f(x)$. Le choix d'un format d'affichage **FIX** 2 indique au calculateur que, au-delà du deuxième chiffre après la virgule décimale, les chiffres ne sont plus significatifs, de sorte que le calculateur n'a pas besoin de perdre du temps à estimer l'intégrale avec une précision incertaine. Consultez le paragraphe «Précision de **INTEG**» ci-après.

Utilisation

Lors du calcul d'une intégrale, **INTEG** exécute itérativement un programme que vous écrivez pour décrire $f(x)$. Vous devez également fournir à **INTEG** deux limites pour x , définissant ainsi un intervalle d'intégration.

- **INTEG** a besoin de 32 registres de programme inutilisés pour fonctionner. S'il ne dispose pas de ces registres, il n'est pas exécuté et le message d'erreur **NO ROOM** (pas assez d'espace) est affiché. Exécutez **GTO**   en mode Programme pour vérifier combien de registres de programme sont encore disponibles.
- Avant d'exécuter **INTEG**, vous devez avoir un programme (stocké en mémoire Programme ou dans un module enfichable) qui définisse votre fonction $f(x)$. Vous devez nommer ce programme à l'aide d'un *label global**. Votre programme peut tirer parti du fait que **INTEG** remplit la pile opérationnelle du calculateur avec son approximation courante de x chaque fois qu'il appelle votre programme.
- Vous introduisez ensuite les deux limites, a et b , dans les registres X et Y respectivement.
- Enfin, vous frappez le nom de votre programme (celui qui définit la fonction) dans le registre alphabétique, puis exéutez **INTEG**.

Lorsque le programme s'arrête et affiche la valeur de l'intégrale, le contenu de la pile opérationnelle du calculateur est le suivant :

- T** = limite inférieure de l'intervalle d'intégration, a .
- Z** = limite supérieure de l'intervalle d'intégration, b .
- Y** = imprécision de l'approximation de l'intégrale.
- X** = approximation de l'intégrale (valeur affichée).

* Ce programme ne doit pas inclure les fonctions **PASN**, **PSIZE**, **AK**, toute fonction d'un lecteur de carte (HP 82104A), ou toute autre fonction modifiant la configuration de la mémoire du calculateur, les affectations des touches, ou les alarmes.

Instructions d'utilisation de INTEG

Instructions	Frappez :	Affichage
1. Passez en mode Programme et compactez la mémoire, pour préparer le calculateur à recevoir un nouveau programme.	PRGM GTO [] []	00 REG nnn
2. Frappez un label global alphabétique pour identifier le programme décrivant $f(x)$.	LBL <i>label global</i>	01 LBL^Tlabel
3. Frappez les lignes de votre programme et mettez fin au programme avec une instruction RTN	RTN	
4. Vérifiez qu'il reste suffisamment de mémoire Programme pour permettre l'exécution de INTEG ($nnn \geq 32$).* Quittez ensuite le mode Programme.	GTO [] [] PRGM	00 REG nnn
5. Frappez le nom de votre programme (défini au point 2) dans le registre alphabétique.	ALPHA <i>label global</i> ALPHA	
6. Introduisez les limites de l'intervalle initial de recherche de l'intégrale :	a ENTER ↑ b	a b
7. Choisissez le format d'affichage correspondant à la précision que vous recherchez pour le résultat.	FIX <i>n</i> ou SCI <i>n</i> ou ENG <i>n</i>	
8. Exécutez INTEG . Le programme fonctionne itérativement pour une durée pouvant atteindre plusieurs minutes, puis il affiche la valeur obtenue pour l'intégrale.	INTEG †	<i>intégrale</i>
9. Pour répéter ce calcul avec un degré de précision différent, choisissez un nouveau format d'affichage, faites défiler vers le bas les valeurs de la pile opérationnelle pour rappeler les limites inférieure et supérieure de l'intervalle, et ré-exécutez INTEG .	FIX <i>n</i> ou SCI <i>n</i> ou ENG <i>n</i> R ↓ R ↓ INTEG	b <i>intégrale</i>
* Si <i>nnn</i> n'est pas ≥ 32 , utilisez SIZE pour affecter plus de mémoire aux registres de programme, ou supprimez des programmes pour libérer des registres. Consultez le manuel d'utilisation du HP-41 à ce sujet.		
† Pour exécuter un programme, appuyez sur XEQ ALPHA <i>nom alphabétique du programme ALPHA</i> ou redéfinissez une touche sur le clavier personnel (USER) en lui affectant cette fonction.		

Remarques

Une pression sur **R/S** met fin au programme INTEG.

Exemple 1

La fonction de Bessel de premier type de degré 0 peut s'exprimer :

$$J_0(x) = 1/\pi \int_0^\pi \cos(x \sin \theta) d\theta.$$

Calculez

$$J_0(1) = 1/\pi \int_0^\pi \cos(\sin \theta) d\theta.$$

Commencez par écrire un programme décrivant la fonction à intégrer. Assurez-vous que le calculateur est en mode Radians pour calculer ces fonctions trigonométriques. Puis, avant d'exécuter **INTEG**, frappez le nom de votre programme dans le registre alphabétique et introduisez les limites de l'intervalle d'intégration. Une fois que vous avez obtenu l'intégrale, n'oubliez pas de la multiplier par $1/\pi$.

Frappez

FIX 4

PRGM **GTO** **•** **•**

LBL **ALPHA** **J01** **ALPHA**

SIN
COS
RTN

GTO **•** **•**

PRGM

ALPHA **J01**
ALPHA

0 **ENTER** **↑** **π**

XEQ **RAD**

Affichage

00 REG *nnn*

01 LBL **T** **J01**

02 SIN

03 COS

04 RTN

00 REG *nnn*

J01

3,1416

3,1416

Valide le format d'affichage utilisé.

Mode Programme ; calculateur prêt à recevoir un programme décrivant $\cos(\sin \theta)$.

Label global alphabétique «J01».

$\sin \theta$.

$\cos(\sin \theta)$.

Fin du programme définissant $f(x)$.

Nombre de registres de programme disponibles ; *nnn* ≥ 32 ?

Sortie du mode Programme.

Place «J01» (nom de votre programme) dans le registre alphabétique (1ère étape dans l'exécution de **INTEG**).

Saisie des limites de l'intervalle d'intégration : zéro et π .

Valide le mode Radians. Vous êtes maintenant prêt à exécuter **INTEG**.

Frappez**XEQ** **INTEG****Affichage****2,4040****π****÷****3,1416****0,7652**

Exécute INTEG et donne l'intégrale (en 25 s environ). N'oubliez pas de multiplier ce résultat par la constante extérieure à l'intégrale pour compléter l'équation.

 $J_0(1)$.

Précision de INTEG

Comme le programme ne peut pas calculer exactement la valeur d'une intégrale, il en donne une *approximation*. La précision de cette approximation dépend de la précision de la fonction à intégrer elle-même, telle qu'elle est calculée par votre programme* ; la précision de la fonction, quant à elle, est affectée par l'erreur d'arrondi du calculateur et par la précision des constantes empiriques.

Pour spécifier la précision voulue pour la fonction, choisissez le format d'affichage (**FIX** *n*, **SCI** *n*, ou **ENG** *n* pour notation «fixe», «scientifique» ou «ingénieur» respectivement) de manière que *n* ne soit pas plus grand que le nombre de chiffres après la virgule que vous considérez comme donnant assez de précision pour les valeurs de la fonction. Si vous choisissez *n* plus petit, le calculateur trouvera l'intégrale plus rapidement mais supposera également que la fonction est précise avec un nombre de chiffres inférieur ou égal à ceux affichés par le format d'affichage.†

INTEG fournit l'intégrale recherchée dans le registre X (registre de l'affichage), et *l'imprécision* de cette approximation dans le registre Y.‡ Pour voir la valeur de l'imprécision, appuyez sur **x_Y**.

Si l'imprécision est supérieure à celle que vous choisi de tolérer, vous pouvez la réduire en choisissant un format d'affichage avec plus de chiffres significatifs et en ré-exécutant INTEG.

* Les intégrales de fonctions présentant certaines caractéristiques (pics, rapides oscillations...) risquent d'être calculées avec imprécision, mais ces fonctions sont rares.

† **SCI** et **ENG** déterminent une imprécision de la fonction qui est *proportionnelle* à la grandeur de la fonction, alors que **FIX** détermine une imprécision qui est *indépendante* de la grandeur de la fonction.

‡ Aucun algorithme d'intégration numérique ne peut calculer la différence exacte entre l'approximation qu'il fournit et l'intégrale réelle. Toutefois, l'algorithme utilisé ici donne une estimation de la *limite supérieure* de cette différence, qui représente l'imprécision de l'approximation (dans le registre Y).

Pour exécuter à nouveau **INTEG** pour le même problème mais avec un format d'affichage différent, vous n'avez pas besoin d'introduire à nouveau les limites de l'intervalle d'intégration (*à condition* que vous n'ayez pas effectué d'autres calculs après avoir obtenu l'intégrale). Comme en fin de calcul, ces limites sont contenues dans les registres **T** et **Z** (comme indiqué au paragraphe «Utilisation»), il vous suffit d'appuyer deux fois sur **R↓** pour les rappeler, puis d'exécuter à nouveau **INTEG**.

Exemple 2

Ayant choisi le format d'affichage **SCI** 2, calculez l'intégrale de l'expression de l'exemple 1 pour $J_0(1)$. Vérifiez l'imprécision du résultat. Puis calculez un résultat ayant une précision de quatre chiffres après la virgule au lieu de deux, et vérifiez son imprécision (assurez-vous que le calculateur est toujours en mode Radians par la présence du témoin **RAD** sur l'affichage). Vous n'aurez à introduire à nouveau les limites de l'intervalle d'intégration que pour le *premier* calcul.

Frappez

SCI 2

0 **ENTER↑** π

XEQ **INTEG**

xsy

SCI 4

R↓ **R↓**

XEQ **INTEG**

xsy

Affichage

3,14 **00**

2,40 **00**

1,57 **-03**

1,5708 **-03**

3,1416 **00**

2,4039 **00**

1,5708 **-05**

Valide la notation scientifique ; précision de deux chiffres après la virgule.

Saisie des limites inférieure (0) et supérieure (π) de l'intervalle.

Valeur de l'intégrale, avec une précision de deux chiffres après la virgule.

Imprécision de l'intégrale.

Choisit une précision de quatre chiffres après la virgule.

Fait défiler la pile du calculateur vers le bas jusqu'à afficher la limite supérieure de l'intervalle.

Valeur de l'intégrale, avec une précision de quatre chiffres après la virgule.

Imprécision (beaucoup plus petite).

Informations pour la programmation

Vous pouvez inclure INTEG dans un programme plus vaste que vous créez vous-même. Assurez-vous que votre programme place des limites supérieure et inférieure dans les registres X et Y juste avant d'exécuter **INTEG**. N'oubliez pas non plus que INTEG cherchera dans le registre alphabétique le nom du programme qui décrit votre fonction.

L'exécution de INTEG fait appel à l'un des six niveaux de retour de branchements dont est doté le calculateur, ce qui laisse cinq niveaux de retour pour un programme appelant INTEG.

Remarquez que INTEG ne peut pas être utilisé de manière récursive (en se rappelant lui-même). Si cela se produit, le programme s'arrête et affiche **RECURSION**. Vous *pouvez* utiliser INTEG avec SOLVE. L'exécution d'un programme combinant INTEG et SOLVE nécessite 32 registres de programme disponibles.

Références

«Utiliser \int », dans le *Manuel des fonctions de haut niveau du HP-15C*, Hewlett-Packard Co., 1982.

Kahan, W.M., «Handheld Calculator Evaluates Integrals», *Hewlett-Packard Journal*, 31:8, August 1980.

EQUATIONS DIFFÉRENTIELLES

Le programme DIFEQ résout les équations différentielles du premier et du second ordre par la méthode du quatrième ordre de Runge-Kutta. Une équation du premier ordre est de la forme $y' = f(x, y)$, avec pour valeurs initiales x_0, y_0 ; une équation du second ordre est de la forme $y'' = f(x, y, y')$, avec pour valeurs initiales x_0, y_0, y'_0 .

Dans les deux cas, vous pouvez introduire la fonction $f(x)$ en mémoire programme sous n'importe quel label *global* non encore utilisé (six caractères au maximum), et l'on supposera que x et y se trouvent respectivement dans les registres X et Y ; y' sera dans le registre Z pour les équations de second ordre. Le programme DIFEQ utilise les registres 00 à 07. Les registres restants sont à votre disposition pour la définition de la fonction.

La solution est une solution numérique, qui calcule y_i pour $x_i = x_0 + ih$ ($i = 1, 2, 3, \dots$), où h est un incrément précisé par l'utilisateur. Vous pouvez modifier la valeur de h à tout moment pendant l'exécution du programme en stockant $h/2$ dans le registre 01. Ceci permet d'obtenir une solution de l'équation aussi proche que l'on veut d'un pôle ($y \rightarrow \pm\infty$).

Équations utilisées par l'algorithme

Premier ordre :

$$y_{i+1} = y_i + \frac{1}{6} (c_1 + 2c_2 + 2c_3 + c_4)$$

avec

$$c_1 = hf(x_i, y_i)$$

$$c_2 = hf\left(x_i + \frac{h}{2}, y_i + \frac{c_1}{2}\right)$$

$$c_3 = hf\left(x_i + \frac{h}{2}, y_i + \frac{c_2}{2}\right)$$

$$c_4 = hf(x_i + h, y_i + c_3)$$

88 Equations différentielles

Second ordre :

$$y_{i+1} = y_i + h \left[y'_i + \frac{1}{6} (k_1 + k_2 + k_3) \right]$$

$$y'_{i+1} = y'_i + \frac{1}{6} (k_1 + 2k_2 + 2k_3 + k_4)$$

avec

$$k_1 = hf(x_i, y_i, y'_i)$$

$$k_2 = hf \left(x_i + \frac{h}{2}, y_i + \frac{h}{2} y'_i + \frac{h}{8} k_1, y'_i + \frac{k_1}{2} \right)$$

$$k_3 = hf \left(x_i + \frac{h}{2}, y_i + \frac{h}{2} y'_i + \frac{h}{8} k_2, y'_i + \frac{k_2}{2} \right)$$

$$k_4 = hf \left(x_i + h, y_i + hy'_i + \frac{h}{2} k_3, y'_i + k_3 \right)$$

Utilisation

Pour la saisie des valeurs d'une équation du second ordre, vous devez introduire d'abord les valeurs de x_0 et de y_0 avant d'introduire celle de y'_0 . Toutes les valeurs doivent être introduites, même si elles sont nulles.

Notez qu'une valeur de h (l'incrément) trop grande peut produire des résultats incorrects.*

* Vous pouvez vérifier un résultat en procédant à reculons à partir du résultat vers la condition initiale en utilisant $-h$. Si vous ne retrouvez pas les valeurs initiales correctes, exécutez à nouveau DIFEQ avec une valeur plus petite de h .

Instructions d'utilisation de DIFEQ

Instructions	Frappez :	Taille : 008 Affichage
<ol style="list-style-type: none"> 1. Préparez le calculateur à recevoir la fonction $f(x, y, y')$. 2. Passez en mode Programme. 3. Chargez la fonction sous le label global alphabétique voulu. Ajoutez RTN. 4. Quittez le mode Programme. 5. Lancez le programme. 6. Frappez le <i>label de la fonction</i> (défini au point 3). 7. Frappez l'ordre de l'équation différentielle (1 ou 2). 8. Introduisez l'incrément (h). 9. Introduisez la valeur initiale de x. 10. Introduisez la valeur initiale de y. 11. Pour une équation de second ordre, introduisez la valeur initiale de y'. 12. Obtenez les valeurs successives de x et y. 	GTO • • PRGM LBL <i>label de la fonction</i> ⋮ RTN PRGM XEQ DIFEQ * <i>label de la fonction</i> R/S <i>ordre</i> R/S <i>h</i> R/S <i>x₀</i> R/S <i>y₀</i> R/S <i>y_{0'}</i> R/S R/S R/S R/S	NAME? ORDER=? STEP SIZE=? X0=? Y0=? <i>x₁</i> (équation de 1er ordre) <i>ou</i> <i>Y0.=?</i> (équation de 2ème ordre) <i>x₁</i> <i>y₁</i> <i>x₂</i> <i>y₂</i> <i>etc.</i>

* Pour exécuter un programme, frappez **XEQ** **ALPHA** *nom alphabétique du programme ALPHA* ou redéfinissez une touche du clavier personnel (USER) en lui affectant cette fonction.

Exemples

En utilisant **FX** comme label de fonction, résolvez numériquement l'équation différentielle de premier ordre

$$y' = \frac{\sin x + \tan^{-1}(y/x)}{y - \ln(\sqrt{x^2 + y^2})}$$

avec $x_0 = y_0 = 1$, et $h = 0,5$. Vous devez choisir le mode angulaire Radians, et trois registres de stockage supplémentaires sont nécessaires pour définir la fonction.

Frappez

FIX 4

XEQ **SIZE** 011

Affichage

Valide le format d'affichage utilisé.
Optionnel—définit le nombre de registres de stockage nécessaires pour ce programme (opération inutile si l'affectation de la mémoire est déjà $\text{SIZE} \geq 011$).

```

PRGM
GTO . .
LBL ALPHA FX ALPHA
XEQ RAD
STO 08
xsy
STO 09
xsy
R→P
LN
STO 10
R↓
RCL 08
SIN
+
RCL 09
RCL 10
-
÷
XEQ DEG
RTN
PRGM

```

Frappez

XEQ DIFEQ

FX R/S

1 R/S

.5 R/S

1 R/S

1 R/S

R/S

R/S

R/S

R/S

R/S

Affichage

NAME?

ORDER=?

STEP SIZE=?

X0=?

Y0=?

1,5000

 x_1

2,0553

 y_1

2,0000

 x_2

2,7780

 y_2

2,5000

 x_3

3,2781

 y_3

etc.

En utilisant DIF comme label de fonction, résolvez l'équation du second ordre

$$(1 - x^2)y'' + xy' = x,$$

où $x_0 = y_0 = y_0' = 0$ et $h = 0,1$.

Ré-écrivez l'équation sous la forme

$$y'' = \frac{x(1 - y')}{1 - x^2} = \frac{x(y' - 1)}{x^2 - 1} \quad x \neq 1$$

Frappez

PRGM

GTO • •

LBL ALPHA DIF ALPHA

STO 08

R↓ R↓

1 -

RCL 08

x

LASTx

x²

1 - ÷

RTN

PRGM

Affichage

92 Equations différentielles

Frappez

XEQ **DIFEQ**

DIF R/S

2 R/S

.1 R/S

0 R/S

0 R/S

0 R/S

Affichage

NAME?

ORDER=?

STEP SIZE=?

X0=?

Y0=?

Y0.=?

0,1000

x_1

0,0002

y_1

0,2000

x_2

0,0013

y_2

0,3000

x_3

0,0046

y_3

0,4000

x_4

0,0109

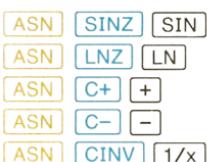
y_4

etc.

OPERATIONS SUR LES NOMBRES COMPLEXES

Cet ensemble de programmes permet d'effectuer des calculs en chaîne sur des nombres complexes de forme $z = x + iy$. Il effectue les quatre opérations arithmétiques ($+$, $-$, \times , \div), ainsi que la plupart des fonctions usuelles de variables complexes z et w ($|z|$, $1/z$, z^n , $z^{1/n}$, e^z , $\ln z$, $\sin z$, $\cos z$, $\tan z$, a^z , $\log_a z$, $z^{1/w}$, et z^w). Vous pouvez mélanger les fonctions et les opérations au cours d'un calcul, ce qui vous permet de calculer des expressions du type $z_3/(z_1 + z_2)$, $e^{z_1 z_2}$, $|z_1 + z_2| + |z_2 - z_3|$, etc., où z_1 , z_2 , et z_3 sont des nombres complexes de la forme $x + iy$.

Si vous voulez ré-utiliser plusieurs fois ces opérations, vous pouvez réaffecter chaque programme à des touches du clavier et créer une grille d'identification correspondante. Exemple de réaffectation :



Ces fonctions opèrent selon une logique qui est une variation de la pile opérationnelle du HP-41 («empilement» des quatre registres mémoire X, Y, Z et T stockant automatiquement les données saisies et résultats intermédiaires). Au lieu de contenir quatre nombres réels, la pile contient ici deux nombres complexes. Soit ξ le registre inférieur d'une pile complexe, et τ son registre supérieur. Ces registres sont analogues aux registres X et T de la pile opérationnelle de votre calculateur*. Un nombre complexe z est introduit dans le registre ξ par la frappe de la séquence z_y [ENTER↑] z_x . Lors de l'introduction d'un second nombre complexe w ([ENTER↑] w_y [ENTER↑] w_x), z est automatiquement transféré dans τ et w est placé dans ξ . Le contenu précédent de τ est perdu.

* Chaque registre de la pile complexe doit en fait contenir deux nombres réels—la partie réelle et la partie imaginaire du nombre complexe. Ainsi, deux registres de votre calculateur constituent un registre de la pile complexe. Dans la suite de ce chapitre, nous parlerons des deux registres contenant le nombre complexe comme s'ils ne constituaient qu'un seul registre.

Piles de registres mémoire

T	t
Z	z
Y	y
X	x

Pile du calculateur

τ	$\left. \begin{array}{l} iz_y \\ z_x \end{array} \right\} z$
ξ	$\left. \begin{array}{l} iw_y \\ w_x \end{array} \right\} w$

Pile complexe

Les fonctions sont calculées sur le seul registre ξ , et le résultat (sauf pour $|z|$, qui donne un nombre réel), est laissé en ξ . τ reste inchangé. Les opérations arithmétiques, elles, font intervenir les deux registres ξ et τ ; le résultat de l'opération est laissé dans ξ .

Ces programmes utilisent les registres 00 à 04.

Équations utilisées par l'algorithme

Soit

$$z_k = x_k + iy_k = r_k e^{i\theta_k}, k = 1, 2$$

$$z = x + iy = re^{i\theta}$$

Notons dans chaque cas le résultat sous la forme $u + iv$.

$$z_1 + z_2 = (x_1 + x_2) + i(y_1 + y_2)$$

$$z_1 - z_2 = (x_1 - x_2) + i(y_1 - y_2)$$

$$z_1 z_2 = r_1 r_2 e^{i(\theta_1 + \theta_2)}$$

$$z_1/z_2 = \frac{r_1}{r_2} e^{i(\theta_1 - \theta_2)}$$

$$|z| = \sqrt{x^2 + y^2}$$

$$1/z = \frac{x}{r^2} - i \frac{y}{r^2}$$

$$z^n = r^n e^{in\theta}$$

$$z^{1/n} = r^{1/n} e^{i\left(\frac{\theta}{n} + \frac{360k}{n}\right)}, k = 0, 1, \dots, n-1$$

(Toutes les n racines seront affichées pour les valeurs de k , dans l'ordre $k = 0, 1, \dots, n-1$.)

$$e^z = e^x (\cos y + i \sin y), \text{ où } y \text{ est en radians}$$

$$\ln z = \ln r + i\theta, \text{ où } z \neq 0$$

$$a^z = e^{z \ln a}, \text{ où } a > 0 \text{ et réel}$$

$$\log_a z = \frac{\ln z}{\ln a}, \text{ où } a > 0 \text{ et réel, et } z \neq 0$$

$z^w = e^{w \ln z}$, où $z \neq 0$, et où w est complexe

$z^{1/w} = e^{\ln z/w}$, où $z \neq 0$, w est complexe et $w \neq 0$

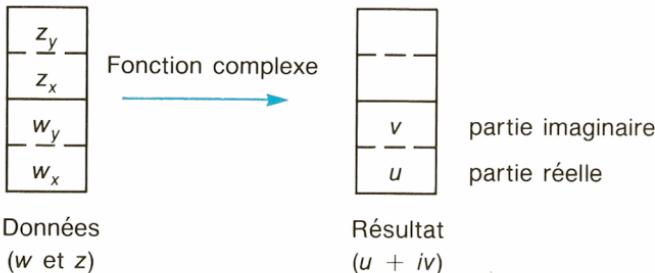
$\sin z = \sin x \cosh y + i \cos x \sinh y$, angles en radians

$\cos z = \cos x \cosh y - i \sin x \sinh y$, angles en radians

$$\tan z = \frac{\sin 2x + i \sinh 2y}{\cos 2x + \cosh 2y}, \text{ angles en radians}$$

Utilisation

Données et résultat types



Instructions d'utilisation des fonctions arithmétiques sur les complexes

Instructions	Frappez :	Affichage	Taille : 005
Arithmétique complexe <ol style="list-style-type: none"> 1. Introduisez le premier nombre complexe ($z_x + iz_y$). 2. Introduisez le second nombre complexe ($w_x + iw_y$). 3. Choisissez l'une des quatre opérations : 	z_y z_x w_y w_x	z_y z_x w_y w_x	

Instructions d'utilisation des fonctions arithmétiques sur les complexes (suite)

Instructions	Frappez :	Affichage
■ Addition	XEQ C+ R/S	U=valeur de u V=valeur de v
■ Soustraction	XEQ C- R/S	U=valeur de u V=valeur de v
■ Multiplication	XEQ Cx R/S	U=valeur de u V=valeur de v
■ Division	XEQ C÷ R/S	U=valeur de u V=valeur de v
4. Le résultat de l'opération reste dans la pile ; reprenez au point 2 pour d'autres opérations.		
Fonction complexe avec un nombre complexe		
1. Introduisez le nombre complexe $(z_x + iz_y)$.	z_y ENTER↑ z_x	z_y z_x
2. Choisissez l'une des opérations suivantes :		
■ SINZ (sin z)	XEQ SINZ R/S	U=valeur de u V=valeur de v
■ COSZ (cos z)	XEQ COSZ R/S	U=valeur de u V=valeur de v
■ TANZ (tan z)	XEQ TANZ R/S	U=valeur de u V=valeur de v
■ MAGZ (module, $ z $)	XEQ MAGZ	U=valeur de u V=valeur de v
■ CINV ($1/z$)	XEQ CINV R/S	R=module
■ e^z	XEQ e^z R/S	U=valeur de u V=valeur de v
■ LNZ ($\ln z$)	XEQ LNZ R/S	U=valeur de u V=valeur de v
■ Z^N (z^n , où n est un entier)	ENTER↑ n XEQ Z^N R/S	n U=valeur de u V=valeur de v
■ Z^{1/N} ($z^{1/n}$)	ENTER↑ n XEQ Z^{1/N} R/S	n U=valeur de u V=valeur de v
Notez que l'on obtient n racines $(u + iv)$.		

Instructions d'utilisation des fonctions arithmétiques sur les complexes (suite)

Instructions	Frappez :	Affichage
<ul style="list-style-type: none"> [a\uparrowZ] (a^z, où a est réel) [LOGZ] ($\log_a z$, où a est réel) 	ENTER \uparrow a XEQ [a \uparrow Z] R/S ENTER \uparrow a XEQ [LOGZ] R/S	a U=valeur de u V=valeur de v a U=valeur de u V=valeur de v
Fonctions complexes avec deux nombres complexes 1. Introduisez le premier nombre complexe ($z_x + iz_y$). 2. Introduisez le second nombre complexe ($w_x + iw_y$). 3. Choisissez l'une des opérations suivantes : <ul style="list-style-type: none"> [Z\uparrowW] (z^w) [Z\uparrow1/W] ($z^{1/w}$) 	z_y ENTER \uparrow z_x ENTER \uparrow w_y ENTER \uparrow w_x XEQ [Z \uparrow W] R/S XEQ [Z \uparrow 1/W] R/S	z_y z_x w_y w_x U=valeur de u V=valeur de v U=valeur de u V=valeur de v

Remarques

Lorsque l'indicateur binaire 04 est armé, vous pouvez accéder à toutes les opérations complexes individuellement sous la forme de sous-programmes dans vos programmes (car elles sont en fait des programmes). Les résultats complexes sont obtenus dans les registres X (partie réelle) et Y (partie imaginaire).

Exemples

Calculez l'expression

$$\frac{z_1}{z_2 + z_3},$$

où $z_1 = 23 + 13i$, $z_2 = -2 + i$, $z_3 = 4 - 3i$.

Suggestion: comme le programme ne peut mémoriser que deux nombres à la fois, effectuez le calcul sous la forme

$$z_1 \times [1/(z_2 + z_3)].$$

98 Opérations sur les nombres complexes

Frappez

FIX 4

XEQ **SIZE** 005

1 **ENTER**
2 **CHS** **ENTER**
3 **CHS** **ENTER** 4
XEQ **C+**

R/S

XEQ **CINV**
R/S
13 **ENTER** 23
XEQ **Cx**
R/S

Affichage

1,0000
-2,0000
4__
U=2,0000
V=-2,0000
U=0,2500
V=0,2500
23__
U=2,5000
V=9,0000

Valide le format d'affichage utilisé.
Optionnel—Choisit le nombre de registres de stockage nécessaire au programme. Cette opération n'est pas nécessaire si vous avez déjà affecté plus d'espace à la mémoire de stockage (SIZE ≥ 005).

Partie réelle ($z_2 + z_3$).

Partie imaginaire ($z_2 + z_3$).

$1/(z_2 + z_3)$

$z_1/(z_2 + z_3)$

Calculez les trois racines cubiques de 8.

Frappez

0 **ENTER**
8 **ENTER** 3
XEQ **Z^{1/N}**
R/S
R/S
R/S
R/S

Affichage

0,0000
3__
U=2,0000
V=0,0000
U=-1,0000
V=1,7321
U=-1,0000
V=-1,7321

Calculez $e^{z^{-2}}$, où $z = (1 + i)$.

Frappez

1 [ENTER↑]
 1 [ENTER↑] 2
 XEQ [Z↑N]
 R/S
 XEQ [CINV]
 R/S
 XEQ [e↑Z]
 R/S

Affichage

1,0000	z^2
2_	
U=0,0000	
V=2,0000	
U=0,0000	z^{-2}
V=-0,5000	
U=0,8776	
V=-0,4794	$e^{z^{-2}}$

Calculez $\sin(2 + 3i)$.

Frappez

3 [ENTER↑] 2
 XEQ [SINZ]
 R/S

Affichage

2_	
U=9,1545	
V=-4,1689	

OPERATIONS VECTORIELLES

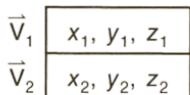
Le programme VC simule un «calculateur vectoriel», qui vient «se superposer» à votre calculateur standard. Il redéfinit les fonctions des deux rangées supérieures de touches pour les opérations ci-après : addition, soustraction, distance (norme de la différence), produit scalaire, produit vectoriel, angle entre vecteurs, norme, et vecteur unitaire. Vous pouvez aussi effectuer ces opérations comme des fonctions standard (vous n'êtes alors plus en mode Calculateur Vectoriel) que vous exécutez comme n'importe quelle autre fonction du HP-41 (non présente sur le clavier). Le nom alphabétique des fonctions figure à la rubrique «Résumé des opérations vectorielles».

Les opérations vectorielles s'effectuent sur des vecteurs à *trois dimensions* définis dans des *coordonnées rectangulaires*. Chaque vecteur a donc trois composantes, V_x , V_y , et V_z . Les vecteurs à deux dimensions doivent avoir z égal à zéro.

VC dispose d'un complément : le programme TR de transformation de coordonnées. Vous pouvez ainsi effectuer des opérations vectorielles et des transformations de coordonnées sur les mêmes données, car vous pouvez accéder au second programme à partir du premier et inversement. TR est traité au chapitre suivant, «Transformation de coordonnées».

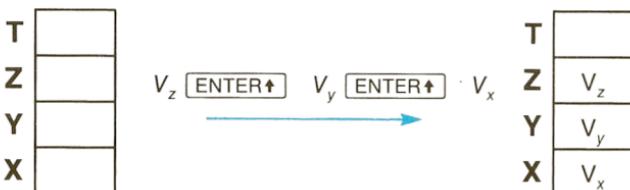
Principe de fonctionnement

La pile vectorielle

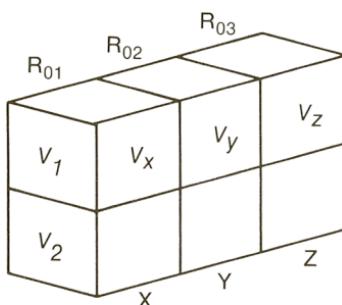


Le Calculateur Vectoriel (VC) crée une *pile vectorielle* fonctionnant de concert avec la pile opérationnelle standard (registres X, Y, Z et T). Lorsque vous introduisez les trois composantes d'un vecteur dans l'ordre V_z, V_y, V_x , elles occupent la pile opérationnelle comme suit :

La pile du calculateur



Comment les deux piles fonctionnent-elles l'une par rapport à l'autre ? Le niveau «inférieur» de la pile vectorielle (V_2) est stocké dans les registres X , Y , et Z de la pile opérationnelle du calculateur, alors que le niveau «supérieur» de la pile vectorielle (V_1) est stocké dans les registres de stockage des données R_{01} , R_{02} , et R_{03} . Vous pouvez vous représenter les registres comme partagés dans une pile tridimensionnelle du type ci-dessous :



La pile vectorielle comporte deux niveaux, de façon à contenir deux vecteurs. Notez toutefois que chaque niveau contient *trois* composantes : les composantes x , y et z de chaque vecteur.

Le diagramme de la page suivante vous montre ce qui se passe lors de l'introduction d'un vecteur et les déplacements qui en résultent à l'intérieur des piles, du point de vue de la pile vectorielle seule et du point de vue de la pile vectorielle et de la pile du calculateur conjointement.

Lorsque vous introduisez deux vecteurs (comme c'est le cas avant l'exécution d'une opération vectorielle type), le premier introduit devient V_1 et le second V_2 . Le fait de saisir le vecteur (fonction **VE**, ou pression sur **R/S** en mode Calculateur Vectoriel) copie le vecteur inférieur (V_2) dans le vecteur supérieur (V_1). Ensuite, lorsque vous introduisez le vecteur suivant, il vient remplacer la copie se trouvant dans le vecteur inférieur (V_2), laissant le premier vecteur dans V_1 et le second vecteur dans V_2 .

Déplacements dans la pile vectorielle

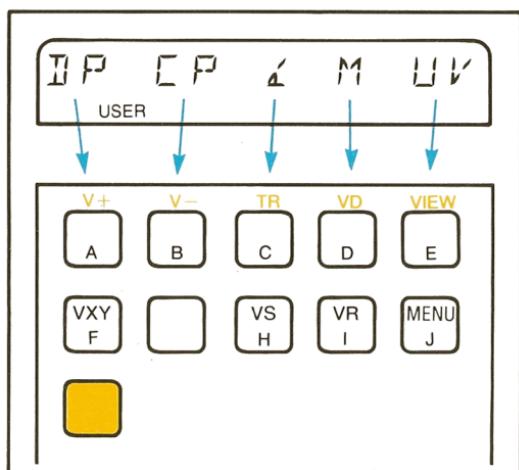
	1. Frappe des composantes du 1er vecteur :	2. Saisie du vecteur :	3. Frappe des composantes du 2ème vecteur :
Pile vectorielle	V_1 V_2 x_1, y_1, z_1	V_1 x_1, y_1, z_1 V_2 x_1, y_1, z_1	V_1 x_1, y_1, z_1 V_2 x_2, y_2, z_2
Frappe de :	V_z V_y V_x	(ou en mode Calculateur Vectoriel)	V_z V_y V_x
Piles vectorielles et du calculateur		$R_{01} R_{02} R_{03}$ V_1 V_2 X Y Z	$R_{01} R_{02} R_{03}$ V_1 V_2 X Y Z

Toutes les opérations sur deux vecteurs dont le résultat est un vecteur placent ce résultat à la fois dans V_1 et dans V_2 . Ceci facilite les calculs vectoriels en chaîne (éventuellement effectués par la suite). Le rappel d'un vecteur copie V_2 dans V_1 puis place le vecteur rappelé dans V_2 .

Utilisation

- Lorsque vous lancez VC (validant ainsi le mode Calculateur Vectoriel), la pile vectorielle *n'est pas* effacée, de sorte que vous pouvez continuer à travailler avec des vecteurs stockés précédemment.
- N'oubliez pas de donner trois dimensions pour chaque vecteur. S'il n'en a que deux, frappez un zéro pour V_z .
- Introduisez la description du vecteur sous la forme de *coordonnées rectangulaires*. Si vous avez des coordonnées polaires (grandeur et angle) pour un vecteur à deux dimensions, convertissez-les à l'aide de la fonction (*conversion polaire-rectangulaire*).
- Pour les opérations faisant intervenir des angles, les unités seront celle du mode angulaire en cours (degrés, radians ou grades).
- La fonction VIEW (voir) () est très utile pour voir quels sont les composantes de V_2 dans la pile.

- V_1 fait référence au vecteur «supérieur» (celui se trouvant dans les registres R_{01} , R_{02} , and R_{03}). V_2 fait référence au vecteur «inférieur» (celui qui se trouve dans X, Y, et Z).



IP = «Dot Product»
Produit scalaire

CP = «Cross Product»
Produit vectoriel

\angle = angle entre 2 vecteurs

M = «Magnitude»
Norme, ou longueur

UV = Unit Vector
Vecteur unitaire

VIEW = Voir les composantes de
V₂

Les autres abréviations du menu figurent dans le tableau p. 107 à 109.

Ce menu vous montre quelles touches correspondent à quelles fonctions dans VC. Appuyez sur **J** pour réafficher ce menu à tout moment.

Pour effacer le menu à n'importe quel moment, appuyez sur **⬅**. Ceci affiche le contenu du registre X mais ne met pas fin au programme. Vous pouvez effectuer des calculs, puis rappeler le menu en appuyant sur **J** (il n'est toutefois *pas nécessaire* que vous effaciez l'affichage avant effectuer des calculs).

Le calculateur vectoriel autorise deux méthodes pour l'introduction d'un vecteur dans la pile vectorielle. La première, la fonction de saisie de vecteur (**VE**), est identique à la touche **ENTER**. La seconde, la touche **R/S**, est une méthode de saisie encore plus rapide. Lorsque vous introduisez les composantes d'un vecteur à partir du clavier et que le menu était la dernière chose affichée avant que vous n'introduisiez les trois composantes du vecteur, une pression sur **R/S** a la même fonction qu'une pression sur **VE**.

Le tableau ci-après montre les séquences de touches à actionner pour exécuter des opérations vectorielles en mode Calculateur Vectoriel (sous programme VC). Pour une définition de chaque opération, consultez le «Résumé des opérations vectorielles» qui fait suite au tableau.

Instructions d'utilisation de VC

Instructions	Frappez :	Taille : 004	Affichage
1. Lancez le programme du Calculateur Vectoriel, VC.	XEQ VC *		DP CP Δ M UV
2. Introduisez les trois composantes de votre premier vecteur (V_1). Séparez deux vecteurs successifs par une «saisie de vecteur» après le premier jeu de coordonnées en exécutant VE ou—uniquement si le menu était la dernière chose affichée avant la saisie de la première composante du vecteur—en appuyant sur R/S .	Z₁ ENTER y₁ ENTER x₁ R/S	Z₁ y₁ x₁	DP CP Δ M UV
3. Introduisez le second vecteur (V_2). <i>N'appuyez pas sur R/S.</i>	Z₂ ENTER y₂ ENTER x₂	Z₂ y₂ x₂	
4. Affichez le menu principal (optionnel).	J		DP CP Δ M UV
5. Exécutez une opération vectorielle :			
■ Produit scalaire, $V_1 \cdot V_2$	A (DP)		DOT = <i>résultat</i>
■ Produit vectoriel, $V_1 \times V_2$	B (CP) R/S † R/S †		X = <i>x du produit</i> Y = <i>y du produit</i> Z = <i>z du produit</i>
■ Angle entre V_1 et V_2	C (Δ)		Δ = <i>résultat</i>
■ Norme (longueur) de V_2 (ceci place aussi le vecteur unitaire de V_2 dans Y, Z, T.)	D (M)		M = <i>résultat</i>
■ Vecteur unitaire de V_2 (ceci place aussi la norme dans le registre T).	E (UV) R/S † R/S †		X = <i>x du vect. unit.</i> Y = <i>y du vect. unit.</i> Z = <i>z du vect. unit.</i>
■ Addition vectorielle, $V_1 + V_2$	A R/S † R/S †		X = <i>x de la somme</i> Y = <i>y de la somme</i> Z = <i>z de la somme</i>
■ Soustraction vectorielle, $V_1 - V_2$	B R/S † R/S †		X = <i>x de la diff.</i> Y = <i>y de la diff.</i> Z = <i>z de la diff.</i>
■ Transformation de coordonnées—cf. le chapitre «Transformation de coordonnées» pour plus de détails. USER C rappelle le mode Calculateur Vectoriel.	C USER C		Z0,Y0,X0 ? DP CP Δ M UV
■ Distance entre V_1 et V_2	D		d = <i>résultat</i>

Instructions d'utilisation de VC (suite)

Instructions	Frappez :	Affichage
6. Restaurez le menu principal après ou entre les opérations (optionnel).	J (ou R/S)	DP CP Δ M UV
7. Pour voir les composantes (x, y, z) de V_2 (le vecteur se trouvant dans la pile) :	E R/S† R/S†	X=valeur de x Y=valeur de y Z=valeur de z
8. Pour permute V_1 et V_2 (les composantes de vecteur dans R_{01} , R_{02} , et R_{03} permuent avec celles des registres X, Y, et Z) :	F	DP CP Δ M UV
9. Pour stocker les composantes du vecteur V_2 sous la forme d'un registre vectoriel n dans R_{3n+1} , R_{3n+2} , et R_{3n+3} ($n \geq 0$) :	n H	DP CP Δ M UV
10. Pour rappeler le contenu (x, y, z) du registre vectoriel n dans V_2 (X, Y, et Z), poussant ainsi V_2 dans V_1 :	I R/S† R/S†	X=valeur de x Y=valeur de y Z=valeur de z

* Pour exécuter un programme, frappez XEQ ALPHA nom alphabétique du programme ALPHA ou redéfinissez une touche du clavier personnel (USER) en lui affectant cette fonction.

† Si une imprimante est connectée à votre calculateur, le menu principal est automatiquement affiché après l'impression des résultats.

Remarques

Vous pouvez supprimer l'affichage des résultats en mode Calculateur Vectoriel en armant l'indicateur binaire 04. Ceci vous permet d'effectuer des calculs successifs plus rapidement en vous évitant de passer par l'étape de l'affichage des résultats. Vous pouvez toujours voir les résultats lorsque vous le voulez en appuyant sur E.

Ce programme utilise des labels locaux alphabétiques (comme expliqué dans le manuel d'utilisation du HP-41) affectés aux touches A-F, H-J, et A-F. Ces affectations locales du programme *n'ont pas la priorité* sur les fonctions que vous avez éventuellement déjà attribuées à ces mêmes touches (le programme ne peut dans ce cas pas fonctionner). *Vous devez donc effacer toutes les affectations du clavier personnel (USER) s'appliquant à ces touches avant d'utiliser ce programme, et éviter de redéfinir ces touches à l'avenir.*

Résumé des opérations vectorielles

Vous pouvez accéder aux opérations vectorielles de deux manières :

- En utilisant le mode Calculateur Vectoriel et les touches qu'il redéfinit, comme expliqué ci-dessus.
- En exécutant directement une fonction vectorielle à l'aide de son nom alphabétique, comme n'importe quelle autre fonction du HP-41 non présente au clavier.
- V_1 fait référence au premier vecteur (ou vecteur «supérieur»), qui se trouve dans R_{01} , R_{02} , et R_{03} . V_2 fait référence au second vecteur (ou vecteur «inférieur»), se trouvant dans les registres X, Y, et Z.

Les opérations effectuent les mêmes calculs, indépendamment de la façon dont elles sont exécutées. Elles sont décrites dans le tableau ci-après.* Vous pouvez aussi les exécuter par leur nom alphabétique tout en étant en mode Calculateur Vectoriel, bien qu'il soit alors plus pratique d'utiliser les touches redéfinies par le programme.

Lorsque vous utilisez des opérations vectorielles sans être en mode Calculateur Vectoriel (c.à.d lorsque vous utilisez leurs noms alphabétiques), il est préférable que le clavier personnel **USER** ne soit pas validé. Ceci évite les conflits entre fonctions affectées aux touches du clavier personnel par le Calculateur Vectoriel et fonctions normales du clavier (comme **xy**).

Tableau des opérations vectorielles

Nom alphabétique de la fonction	Description
CROSS (produit vectoriel)	$V_1 \times V_2$. Le produit tridimensionnel est obtenu dans V_2 (registres X, Y, Z). Une copie va dans V_1 . R_0 n'est pas sauvegardé. En mode Calculateur Vectoriel, utilisez B (CP).
DOT (produit scalaire)	$V_1 \bullet V_2$. Le produit scalaire est obtenu dans le registre X. (V_2 est détruit ; V_1 reste inchangé.) En mode Calculateur Vectoriel, utilisez A (DP).
TR (transformation de coordonnées)	Appelle le programme de transformation de coordonnées, TR. Consultez le chapitre suivant. En mode Calculateur Vectoriel, utilisez BC .

* La fonction **VIEW**, qui permet de voir le vecteur, et le calcul de la norme ne sont disponibles qu'en mode Calculateur Vectoriel. Toutefois, la norme peut également être obtenue comme une partie du résultat du calcul du vecteur unitaire.

Tableau des opérations vectorielles (suite)

Nom alphabétique de la fonction	Description
UV (vecteur unitaire)	<p>Convertit V_2 (dans X, Y, Z) en son vecteur unitaire, et donne la norme du vecteur dans le registre T (V_1 reste inchangé).</p> <p>En mode Calculateur Vectoriel, utilisez E (UV).</p> <p>Remarque : le vecteur unitaire de (0,0,0) est (0,0,1) avec une norme égale à zéro.</p>
V+ (addition vectorielle)*	<p>$V_1 + V_2$. Donne la somme à la fois dans V_1 et dans V_2.</p> <p>En mode Calculateur Vectoriel, utilisez A.</p>
V- (soustraction vectorielle)*	<p>$V_1 - V_2$. Donne la différence à la fois dans V_1 et dans V_2.</p> <p>En mode Calculateur Vectoriel, utilisez B.</p>
V* (produit d'un vecteur par un scalaire)	<p>$V_2 * a$. Multiplie V_2 (dans Y, Z, T) par a (qui se trouve dans le registre X), et donne le résultat dans X, Y, et Z.</p>
VΔ (angle entre vecteurs)	<p>Donne l'angle dans le registre X. Cet angle est exprimé dans l'unité définie par le mode angulaire en cours. V_1 et V_2 ne sont pas sauvegardés ; le vecteur unitaire de V_2 aboutit dans V_1.</p> <p>En mode Calculateur Vectoriel, utilisez C (Δ).</p> <p>Remarque : le vecteur (0,0,0) est supposé avoir la même direction que (0,0,1).</p>
VD (distance)	<p>Fournit la distance entre V_1 et V_2 dans le registre X. Fournit aussi le vecteur différence ($V_1 - V_2$) dans V_1. V_2 n'est pas sauvegardé.</p> <p>En mode Calculateur Vectoriel, utilisez D.</p>
VE (saisie d'un vecteur)	<p>Fonction analogue à une pression sur ENTER↑. Sert à séparer la saisie de deux vecteurs successifs (V_1, puis V_2) avant l'exécution d'une opération (la saisie d'un vecteur copie le premier vecteur de X, Y, Z vers R_{01}, R_{02}, R_{03}).</p> <p>En mode Calculateur Vectoriel, vous pouvez aussi appuyer sur R/S, mais seulement si le menu était la dernière chose affichée.</p>
VR (rappel d'un vecteur)	<p>Avec n ($n > 0$)† dans le registre X, réalise une copie de V_2 dans V_1, puis rappelle un vecteur tridimensionnel du registre vectoriel n (registres de stockage R_{3n+1}, R_{3n+2}, et R_{3n+3}) dans V_2 (X, Y, et Z). Fonction analogue à une pression sur RCL (le V_2 précédent «monte» dans V_1, où il remplace V_1).</p> <p>En mode Calculateur Vectoriel, utilisez I.</p>

Tableau des opérations vectorielles (suite)

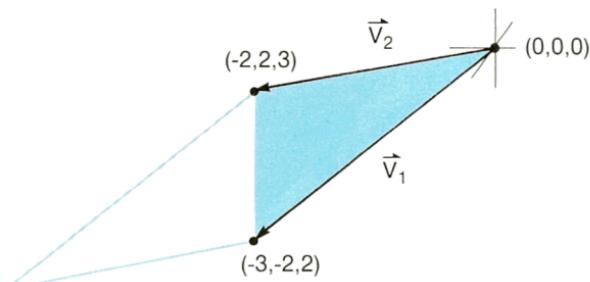
nom alphabétique de la fonction	Description
VS (stockage d'un vecteur)	Avec n ($n > 0$)† dans le registre X, copie et stocke V_2 (se trouvant dans Y, Z, et T) sous la forme de registre vectoriel dans les registres de stockage R_{3n+1} , R_{3n+2} , et R_{3n+3} . Fonction analogue à une pression sur STO (V_2 reste inchangé). En mode Calculateur Vectoriel, utilisez H .
VXY (permutation de vecteurs)	V_1 permute avec V_2 . Les valeurs x_1 , y_1 , et z_1 passent des registres R_{01} , R_{02} , et R_{03} aux registres X, Y, et Z, et les valeurs x_2 , y_2 , et z_2 passent des registres X, Y, et Z aux registres R_{01} , R_{02} , et R_{03} . En mode Calculateur Vectoriel, utilisez F (xsy).

* N'oubliez pas que + et – sont des caractères alphabétiques secondaires (il faut appuyer d'abord sur la touche jaune pour pouvoir les générer).

† Si $n = 0$, **VR** et **VS** copient tous deux V_2 dans V_1 , comme **VE**. Ne prenez pas $n < 5$ si vous avez l'intention de stocker des vecteurs en vue de leur utilisation avec le programme TR (**■ C**).

Exemples

Calculez la surface d'un triangle déterminé par les vecteurs $V_1 = (-3, -2, 2)$ et $V_2 = (-2, 2, 3)$. Rappelez-vous que la surface du parallélogramme déterminé par V_1 et V_2 est égale à la norme de $V_1 \times V_2$.



Frappez**FIX** 4**[XEQ]** **[SIZE]** 004**[XEQ]** **[VC]**

2 **ENTER**
 2 **CHS** **ENTER**
 3 **CHS** **R/S**

3 **ENTER**
 2 **ENTER**
 2 **CHS**
J

[B] **(CP)**
R/S
R/S

J or **R/S****[D]** **(M)**2 **÷****Affichage****DP CP** \triangle **M UV**

2,0000
-2,0000
DP CP \triangle **M UV**

3,0000
2,0000
-2
DP CP \triangle **M UV**

X = -10,0000
Y = 5,0000
Z = -10,0000

DP CP \triangle **M UV****M = 15,0000****7,5000**

Définit le format d'affichage utilisé.

Optionnel—définit le nombre de registres de stockage nécessaires. Etape superflue si vous avez déjà une affectation mémoire SIZE ≥ 004 .

Lance le Calculateur Vectoriel (vous pouvez aussi utiliser les opérations directement en frappant leurs noms alphabétiques).

Frappez la valeur de z_1 , de y_1 , puis de x_1 , et terminez la séquence en saisissant le vecteur.

Frappez la valeur de z_2 , de y_2 , puis de x_2 ,

Rappel du menu de VC (optionnel).

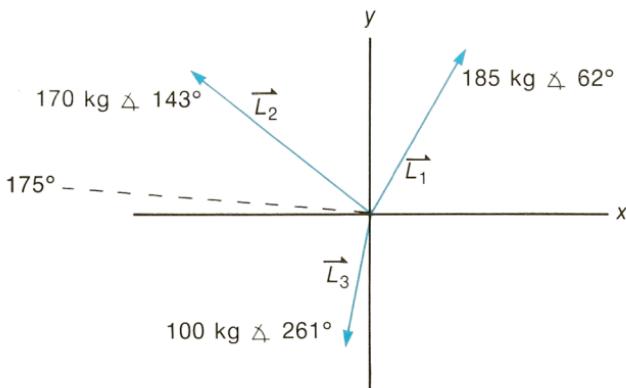
Le résultat est $(-10, 5, -10)$.

Prêt à trouver la norme.

Norme (longueur), égale à la surface du parallélogramme.

Ceci est la surface du triangle, qui est la moitié de celle du parallélogramme.

Trouvez la résultante des trois forces suivantes dans la direction donnée (175 degrés). Utilisez la somme des trois forces et le produit scalaire. Vous devez d'abord convertir les coordonnées polaires en coordonnées rectangulaires. N'oubliez pas de définir z comme égal à zéro.



Stockez les coordonnées polaires de L_3 et du vecteur unitaire à 175° pour pouvoir les ré-utiliser (nous referons les calculs avec une valeur double du module de L_3). Les résultats seront stockés dans les registres vectoriels 1 et 2. Nous exécutons ici les opérations vectorielles avec leurs noms alphabétiques, mais vous pouvez aussi utiliser le Calculateur Vectoriel (VC). Vérifiez que le clavier personnel (USER) n'est pas validé.

Frappez

XEQ **SIZE** 010

XEQ **DEG** *

0 **ENTER**↑

62 **ENTER**↑

185 **P→R**

Affichage

0,0000

62,0000

86,8522

Optionnel—définit le nombre de registres de stockage nécessaires (y compris pour stocker les vecteurs).
Etape superflue si l'affectation mémoire est déjà suffisante ($\text{SIZE} \geq 010$).

Mode angulaire en degrés.

Frappe d'un zéro pour la composante z (pour préparer les opérations vectorielles *après* conversion des coordonnées).

Pour convertir L_1 en coordonnées rectangulaires, introduisez d'abord θ , puis r .

Abscisse x de L_1 .

* Si le témoin (USER) du clavier personnel est affiché, appuyez sur **USER** pour le faire disparaître.

Frappez**[x₂y]** ***[x₂y]** ***[XEQ** **VE**0 **[ENTER↑**
143 **[ENTER↑**
170 **[P→R]****[XEQ** **V+**0 **[ENTER↑**
261 **[ENTER↑**
100 **[P→R]****[XEQ** **VS****[XEQ** **V+**0 **[ENTER↑**
175 **[ENTER↑**
1 **[P→R]****[XEQ** **VS****Affichage****163,3453****86,8522****86,8522****0,0000**
143,0000
–135,7680**–48,9158****0,0000**
261,0000
–15,6434**–15,6434****–64,5592****0,0000**
175,0000
–0,9962**–0,9962**Ordonnée y de L_1 . Cette étape est optionnelle—elle vous permet de voir y .Rétablit x dans X et y dans Y—nécessaire uniquement si vous les avez permutés (au cours de l'étape précédente).

Pas de menu ; affichage du résultat précédent.

Affichage de x_2 . L_2 est converti en coordonnées rectangulaires.Abscisse x de la résultante (à la fois dans V_1 et V_2).Abscisse x_3 . L_3 est converti en coordonnées rectangulaires.Stockage de L_3 dans le registre vectoriel 1 (dans R₄, R₅, R₆).Abscisse x de la résultante de ($L_1 + L_2 + L_3$) dans V_1 et V_2 .Abscisse x du vecteur unitaire à 175°.Stocke le vecteur unitaire à 175° dans le registre vectoriel 2 (dans R₇, R₈, R₉).

* Vous remarquerez que lorsque le témoin USER (clavier personnel) est affiché, vous ne pouvez pas utiliser **[x₂y]** en mode Calculateur Vectoriel pour permutez X et Y car cette touche est rédéfinie pour permutez V_1 et V_2 . Utilisez alors **[R↑]**.

Frappez

XEQ DOT

Affichage

78,8586

Le produit scalaire donne la norme de la projection de la résultante L sur la droite à 175°.

XEQ VXY

-64,5592

Place la résultante $(L_1 + L_2 + L_3)$ dans V_2 (X, Y, Z).

1 XEQ VR

-15,6434

Rappelle L_3 . Ajoute L_3 à la somme précédente (doublant ainsi la valeur de la norme de L_3).

XEQ V+

-80,2027

2 XEQ VR

-0,9962

Rappelle le vecteur unitaire à 175°.

XEQ DOT

85,8342

Calcule le nouveau produit scalaire pour la projection de la nouvelle résultante sur la droite à 175°.

Informations pour la programmation

Vous pouvez utiliser les sous-programmes suivants de VC dans vos propres programmes. Ce sont des opérations vectorielles à trois dimensions sur un ou deux vecteurs.

Taille mémoire minimale pour l'exécution : SIZE 004, sans compter les opérations de stockage et de rappel des vecteurs.

Sous-programmes

Nom du sous-programme	Registres initiaux	Registres finals
CROSS (produit vectoriel)	Registre X = V_{2x} Registre Y = V_{2y} Registre Z = V_{2z} R ₀₁ = V_{1x} R ₀₂ = V_{1y} R ₀₃ = V_{1z}	X = $(V_1 \times V_2)_x$ Y = $(V_1 \times V_2)_y$ Z = $(V_1 \times V_2)_z$ R ₀₀ = espace de travail R ₀₁ = $(V_1 \times V_2)_x$ R ₀₂ = $(V_1 \times V_2)_y$ R ₀₃ = $(V_1 \times V_2)_z$
DOT (produit scalaire)	Registre X = V_{2x} Registre Y = V_{2y} Registre Z = V_{2z} R ₀₁ = V_{1x} R ₀₂ = V_{1y} R ₀₃ = V_{1z}	X = $V_1 \cdot V_2$ R ₀₁ = V_{1x} R ₀₂ = V_{1y} R ₀₃ = V_{1z}

Sous-programmes (suite)

Nom du sous-programme	Registres initiaux	Registres finals
$V+$ (addition vectorielle)	Registre $X = V_{2x}$ Registre $Y = V_{2y}$ Registre $Z = V_{2z}$ $R_{01} = V_{1x}$ $R_{02} = V_{1y}$ $R_{03} = V_{1z}$	$X = V_{1x} + V_{2x}$ $Y = V_{1y} + V_{2y}$ $Z = V_{1z} + V_{2z}$ $R_{01} = V_{1x} + V_{2x}$ $R_{02} = V_{1y} + V_{2y}$ $R_{03} = V_{1z} + V_{2z}$
$V-$ (soustraction vectorielle)	Registre $X = V_{2x}$ Registre $Y = V_{2y}$ Registre $Z = V_{2z}$ $R_{01} = V_{1x}$ $R_{02} = V_{1y}$ $R_{03} = V_{1z}$	$X = V_{1x} - V_{2x}$ $Y = V_{1y} - V_{2y}$ $Z = V_{1z} - V_{2z}$ $R_{01} = V_{1x} - V_{2x}$ $R_{02} = V_{1y} - V_{2y}$ $R_{03} = V_{1z} - V_{2z}$
$V*$ (produit d'un vecteur par un scalaire)	$X = a$ $Y = V_x$ $Z = V_y$ $T = V_z$	$X = V_x * a$ $Y = V_y * a$ $Z = V_z * a$
$V\Delta$ (angle entre deux vecteurs)	Registre $X = V_{2x}$ Registre $Y = V_{2y}$ Registre $Z = V_{2z}$ $R_{01} = V_{1x}$ $R_{02} = V_{1y}$ $R_{03} = V_{1z}$	X = le plus petit angle entre V_1 et V_2 R_{01} = vecteur unitaire V_{2x} R_{02} = vecteur unitaire V_{2y} R_{03} = vecteur unitaire V_{2z}
VD (distance)	Registre $X = V_{2x}$ Registre $Y = V_{2y}$ Registre $Z = V_{2z}$ $R_{01} = V_{1x}$ $R_{02} = V_{1y}$ $R_{03} = V_{1z}$	X = distance entre V_1 et V_2 $R_{01} = V_{1x} - V_{2x}$ $R_{02} = V_{1y} - V_{2y}$ $R_{03} = V_{1z} - V_{2z}$
VE (saisie de vecteur)	Registre $X = V_x$ Registre $Y = V_y$ Registre $Z = V_z$	Registre $X = V_x$ Registre $Y = V_y$ Registre $Z = V_z$ $R_{01} = V_x$ $R_{02} = V_y$ $R_{03} = V_z$
VR (rappel de vecteur)	$X = n$ $Y = V_x$ $Z = V_y$ $T = V_z$	$X = R_{3n+1}$ $Y = R_{3n+2}$ $Z = R_{3n+3}$ $R_{01} = V_x$ $R_{02} = V_y$ $R_{03} = V_z$

Sous-programmes (suite)

Nom du sous-programme	Registres initiaux	Registres finals
VS (stocker un vecteur)	$X = n$ $Y = V_x$ $Z = V_y$ $T = V_z$	$X = V_x$ $Y = V_y$ $Z = V_z$ $R_{3n+1} = V_x$ $R_{3n+2} = V_y$ $R_{3n+3} = V_z$
VXY (permutation de vecteurs)	Registre $X = V_{2x}$ Registre $Y = V_{2y}$ Registre $Z = V_{2z}$ $R_{01} = V_{1x}$ $R_{02} = V_{1y}$ $R_{03} = V_{1z}$	Registre $X = V_{1x}$ Registre $Y = V_{1y}$ Registre $Z = V_{1z}$ $R_{01} = V_{2x}$ $R_{02} = V_{2y}$ $R_{03} = V_{2z}$
UV (vecteur unitaire)	$X = V_x$ $Y = V_y$ $Z = V_z$	$X = \text{vecteur unitaire } x$ $Y = \text{vecteur unitaire } y$ $Z = \text{vecteur unitaire } z$ $T = \text{norme}$

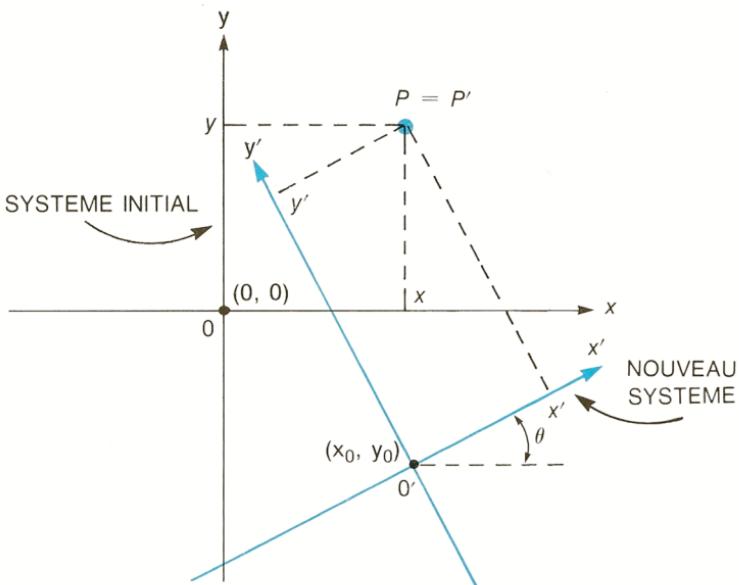
Commentaires. Les opérations vectorielles peuvent s'effectuer sur un ou deux vecteurs. L'un est stocké dans la pile opérationnelle du calculateur (registres X, Y et Z), et l'autre dans les registres R_{01} , R_{02} , et R_{03} . Lors des opérations sur deux vecteurs, V_1 va dans R_{01} – R_{03} et V_2 va dans la pile du calculateur. Les composantes du vecteur sont stockées dans l'ordre, à savoir V_x , V_y , et V_z dans X, Y, et Z ou dans R_{01} , R_{02} , et R_{03} , respectivement.

TRANSFORMATIONS DE COORDONNEES

Le programme TR effectue une translation de systèmes tridimensionnels, avec ou sans rotation. Il utilise des parties du programme VC pour les opérations vectorielles. Vous pouvez y accéder soit directement, soit depuis VC (cf. chapitre précédent, «Opérations vectorielles»).

Le programme vous demande les coordonnées de l'origine du *nouveau* système (x_0, y_0, z_0) , l'angle de rotation de ce système par rapport au système *initial*, et l'axe autour duquel la rotation s'effectue. Vous pouvez ensuite introduire des coordonnées (x, y, z) dans le système initial pour les convertir dans le nouveau système (x', y', z') , ou des coordonnées (x', y', z') dans le nouveau système pour les convertir dans le système initial (x, y, z) . Dans un espace à deux dimensions, introduisez zéro comme valeur de z_0 .

Rotation bidimensionnelle autour d'un axe $(0, 0, 1)$



Après avoir spécifié la nouvelle origine (x, y, z) , vous spécifiez l'angle de rotation. Dans un espace à trois dimensions, vous définissez aussi l'axe de rotation par un vecteur (a, b, c) de même direction, ayant une norme (longueur) quelconque non nulle.

Equations utilisées par l'algorithme

$$\vec{P}' = [(\vec{P} - \vec{T}) \cdot \vec{n}] \vec{n} (1 - \cos\theta) + (\vec{P} - \vec{T}) \cos\theta + [(\vec{P} - \vec{T}) \times \vec{n}] \sin\theta$$

$$\vec{P} = [(\vec{P}' \cdot \vec{n}) \vec{n} (1 - \cos\theta) + \vec{P}' \cos\theta + (\vec{P}' \times \vec{n}) \sin(-\theta)] + \vec{T}$$

avec

$\vec{P}' = \vec{OP}$, qui donne les coordonnées de P dans le nouveau système

$\vec{P} = \vec{OP}$, qui donne les coordonnées de P dans le système initial

\vec{T} = vecteur de translation ($\vec{OO'}$)

\vec{n} = vecteur unitaire de rotation (a, b, c)

θ = angle de rotation

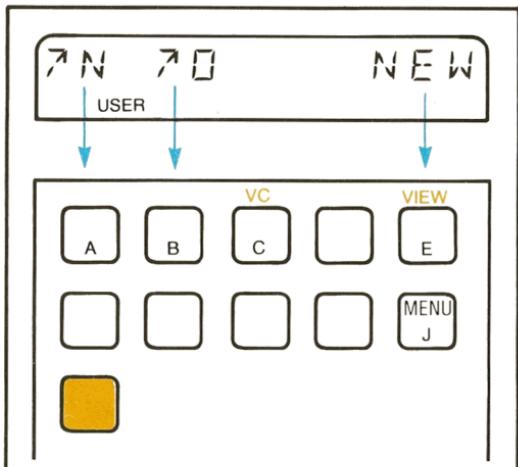
Les transformations bidimensionnelles sont traitées comme un cas particulier des transformations tridimensionnelles où (a, b, c) vaut (0, 0, 1).

Utilisation

Vous pouvez lancer TR soit directement (**XEQ** **TR**) soit à partir du Calculateur Vectoriel (**■ C**). Le Calculateur Vectoriel est présenté au chapitre «Opérations vectorielles».

Introduisez les coordonnées sous la forme de coordonnées rectangulaires et spécifiez les angles en fonction du mode angulaire en cours (degrés, radians ou grades).

- Dans les systèmes à deux dimensions, introduisez un zéro pour z.
- Pour une translation simple (sans rotation), introduisez un angle de rotation égal à zéro.
- Pour une rotation simple (sans translation), introduisez des zéros pour x_0 , y_0 , et z_0 .
- Le signe de l'angle de rotation est déterminé par la règle de la main droite et par la direction du vecteur de rotation. Dans les systèmes à deux dimensions, le sens de rotation est considéré comme positif dans le sens trigonométrique (inverse des aiguilles d'une montre).
- Vous pouvez entrer et sortir du mode Calculateur Vectoriel en appuyant sur **■ C** («C» pour Calculateur et transformations de Coordonnées). Vous pouvez ensuite effectuer les opérations sur les vecteurs se trouvant dans les registres de la pile et dans les registres de stockage (cf. «Remarques» pour des précisions sur les registres dans lesquels sont stockés les composantes des vecteurs).
- La fonction VIEW (voir) (**■ E**) est très utile pour vérifier ce que contient la pile.



$\rightarrow N$ = conversion vers le nouveau système

$\rightarrow O$ = conversion vers le système initial

$N \rightarrow W$ = Permet de créer un nouveau système transformé

VC = «Vector Calculator»

Vous fait passer en mode Calculateur Vectoriel

VIEW = Permet de visualiser les coordonnées du point se trouvant dans la pile.

Une fois que vous avez introduit vos variables, le menu ci-dessus vous montre quelles touches correspondent à quelles fonctions dans TR. Pour réafficher ce menu à tout moment, appuyez sur **J** si le témoin **USER** du clavier personnel est affiché (s'il ne l'est pas, appuyez sur la touche **USER** pour l'afficher). Ou bien, si le calculateur affiche des résultats, appuyez sur **R/S** jusqu'à ce que le menu apparaisse. L'affichage du menu ne perturbe en aucun cas le programme.

Pour effacer le menu, appuyez sur **⬅**. Ceci ramène sur l'affichage le contenu du registre X, sans mettre fin au programme. Vous pouvez donc effectuer des calculs, puis continuer le programme en appuyant sur **J** (il n'est toutefois pas nécessaire d'effacer l'affichage du programme avant d'effectuer les calculs).

Instructions d'utilisation de TR

Instructions	Frappez :	Taille : 017
1. Lancez le programme TR. Le menu indique l'emplacement des fonctions sur le clavier.	XEQ TR *	Z0,Y0,X0 ?
2. Introduisez l'origine du nouveau système.	Z0 ENTER ↑ Y0 ENTER ↑ X0 R/S	Z0 Y0 ROTΔ ?
3. Introduisez l'angle de rotation de la transformation :	θ R/S	c,b,a ?

Instructions d'utilisation de TR (suite)

Instructions	Frappez :	Affichage
4. Dans un système tridimensionnel : introduisez les composantes du vecteur de rotation. Dans un système bidimensionnel : appuyez simplement sur [R/S].	c [ENTER↑] b [ENTER↑] a [R/S]	c b tN tO NEW
5. Pour transformer les coordonnées d'un point du système initial en coordonnées dans le nouveau système (tN), introduisez les trois coordonnées de ce point et exécutez tN (dans un système bidimensionnel, introduisez z=0).	z [ENTER↑] y [ENTER↑] x [A] (tN) [R/S] † [R/S] † [R/S] †	z y X= x' Y= y' Z= z' tN tO NEW
6. Pour transformer les coordonnées d'un point dans le nouveau système en coordonnées dans le système initial (tO), introduisez les trois coordonnées de ce point et exécutez tO (dans les systèmes bidimensionnels, introduisez z=0).	z' [ENTER↑] y' [ENTER↑] x' [B] (tO) [R/S] † [R/S] † [R/S] †	z' y' X= x Y= y Z= z tN tO NEW
7. Pour voir les coordonnées du point stockées dans la pile :	[E] [R/S] † [R/S] † [R/S] †	X= coordonnée sur x Y= coordonnée sur y Z= coordonnée sur z tN tO NEW
8. Pour transformer un autre jeu de coordonnées, recommencez à partir du point 5 ou 6.		
9. Pour définir un nouveau système transformé, exécutez NEW puis revenez au point 2.	[E] (NEW)	Z0,Y0,X0 ?
10. Pour effectuer des opérations vectorielles, passez en mode Calculateur Vectoriel (VC). Vous avez alors accès à toutes les fonctions décrites au chapitre «Opérations vectorielles».	[C] (Le clavier personnel (USER) doit être validé)	DP CP Δ M UV
11. Pour revenir au programme TR depuis VC :	[C]	Z0,Y0,X0 ?
12. Pour transformer un vecteur résultat V_2 obtenu dans VC, ignorez les messages de sollicitation initiaux etappelez le menu principal (en supposant qu'un système transformé a déjà été défini) :	[USER] [J]	tN tO NEW

* Pour exécuter un programme, appuyez sur [XEQ] ALPHA nom alphabétique du programme ALPHA ou redéfinissez une touche du clavier personnel (USER) en lui affectant cette fonction.

† Séquence inutile si une imprimante est connectée au calculateur, car l'imprimante imprime automatiquement les résultats, à la suite de quoi le menu principal est affiché.

Remarques

Ce programme affecte des labels locaux alphabétiques (comme expliqué dans le manuel d'utilisation du HP-41) aux touches **A**, **B**, **E**, **C**, et **J**. Ces affectations locales du programme *n'ont pas la priorité* sur les fonctions que vous avez éventuellement déjà affectées à ces mêmes touches, et le programme ne peut dans ce cas pas être exécuté. *Vous devez donc effacer toutes les affectations du clavier personnel (USER) s'appliquant à ces touches avant d'utiliser ce programme*, et éviter de redéfinir ces touches à l'avenir.

Toutefois, ces labels locaux alphabétiques ne sont actifs que lorsque le témoin **USER** est affiché. Ceci vous permet d'utiliser les fonctions arithmétiques des deux rangées supérieures de touches lorsque le témoin **USER** n'est pas affiché (tant que **USER** est affiché, les touches susmentionnées sont redéfinies et n'exécutent donc plus leur fonction normale).

Stockage des données. Le vecteur ou point que vous voulez transformer est stocké dans R_{04} , R_{05} , R_{06} , qui constituent le registre de stockage vectoriel 1 (les données proviennent initialement des registres X, Y et Z). Le vecteur de rotation est stocké dans R_{07} , R_{08} , R_{09} , qui constituent le registre de stockage vectoriel 2. L'origine du nouveau système est stockée dans R_{10} , R_{11} , R_{12} , qui constituent le registre de stockage vectoriel 3. L'angle de rotation est stocké dans R_{16} , tandis que R_{13} , R_{14} , et R_{15} servent d'espace de travail au programme.

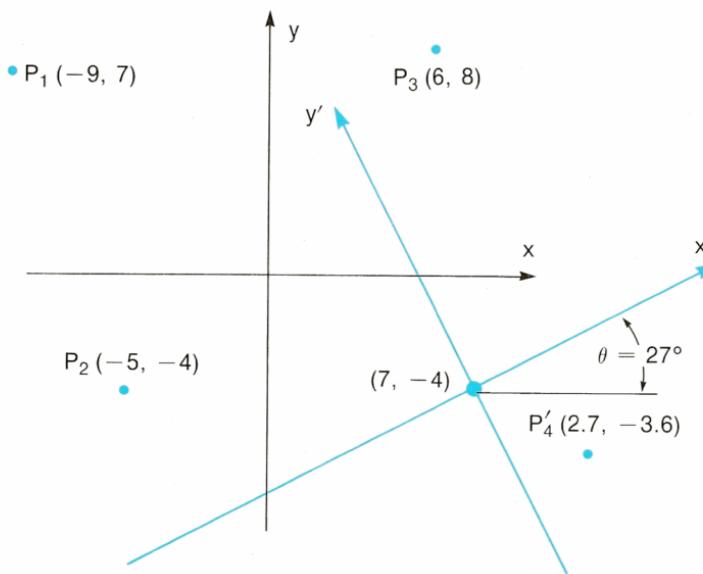
Si vous exécutez des opérations de stockage de vecteurs (**VS**, **VR**, et le Calculateur Vectoriel) avec **TR**, n'oubliez pas que **TR** utilise R_0 - R_{16} lorsqu'il est initialisé (**XEQ** **TR**). Cela signifie que vous ne devez pas stocker de vecteurs dans les registres vectoriels 1 à 5 (si vous prévoyez d'utiliser **TR** dans vos opérations vectorielles).

Indicateurs binaires. L'indicateur binaire 01 sert à indiquer si la transformation s'effectue vers le nouveau système ou vers le système initial. Lorsque cet indicateur binaire est armé, la transformation s'effectue vers le nouveau système.

L'indicateur binaire 05 est armé lorsque le système subit une rotation.

Exemples

Soit les systèmes de coordonnées (x, y) et (x', y') indiqués ci-dessous.



Trouvez les coordonnées des points P_1 , P_2 , et P_3 dans le système (x', y') . Trouvez les coordonnées du point P_4' dans le système (x, y) .

Frappez

FIX 4

XEQ **SIZE** 017

XEQ **TR**

0 **ENTER** \uparrow

4 **CHS** **ENTER** \uparrow
7 **R/S**

Affichage

Z0,Y0,X0 ?	
0,0000	
-4,0000 ROT\triangle?	

Valide le format d'affichage utilisé.

Optionnel—définit le nombre de registres de stockage nécessaires. Étape superflue si l'affectation mémoire est déjà suffisante (**SIZE** \geq 017).

Demande l'introduction de z_0 , y_0 , et x_0 du nouveau système.

Introduisez zéro pour z_0 .

Demande l'introduction de l'angle de rotation.

Frappez

27 [R/S]

[R/S]

0 [ENTER↑] 7 [ENTER↑]
9 [CHS] [A] (↑N)

[R/S]

[R/S]

[R/S]

0 [ENTER↑] 4 [CHS]
[ENTER↑]
5 [CHS] [A] (↑N)

[R/S]

[R/S]

[R/S]

0 [ENTER↑] 8 [ENTER↑]
6 [A] (↑N)

[R/S]

[R/S]

[R/S]

0 [ENTER↑] 3.6 [CHS]
[ENTER↑]

2.7 [B] (↑O)

[R/S]

[R/S]

Affichage

c,b,a ?

↑N ↑O NEW

7,0000
X = -9,2622

Y = 17,0649

Z = 0,0000

↑N ↑O NEW

-4,0000
X = -10,6921

Y = 5,4479

Z = 0,0000

↑N ↑O NEW

8,0000
X = 4,5569

Y = 11,1461

Z = 0,0000

↑N ↑O NEW

-3,6000
X = 11,0401

Y = -5,9818

Z = 0,0000

Demande
l'introduction du
vecteur de rotation.
Sautéz cette étape
dans les systèmes
bidimensionnels.Demande
l'introduction de P_1 . x_1' y_1' z_1' Prêt à recevoir P_2 .
Etape optionnelle—
elle sert à afficher le
menu principal. x_2' de P_2 , y_2' z_2'

Réaffiche le menu.

 x_3' de P_3 . y_3' z_3'

Réaffiche le menu.

 x_4 de P_4' . y_4 z_4

Un système tridimensionnel est traduit en (2,45, 4,00, 4,25). Après la translation, il subit une rotation de 62,5 degrés autour d'un axe (0, -1, -1). Dans le système initial, un point avait pour coordonnées (3,9, 2,1, 7,0). Quelles sont les coordonnées de ce point dans le nouveau système ?

Frappez

[J]

[E] (NEW)

4.25 [ENTER↑] 4 [ENTER↑]
 2.45 [R/S]
 62.5 [R/S]
 1 [CHS] [ENTER↑]
 1 [CHS] [ENTER↑]
 0 [R/S]
 7 [ENTER↑] 2.1 [ENTER↑]
 3.9 [A] (↑N)
 [R/S]
 [R/S]

Affichage

↑N ↑O NEW

Z0,Y0,X0 ?

4,0000

ROT△?

c,b,a ?

-1,0000

-1,0000

↑N ↑O NEW

2,1000

X=3,5861

Y=0,2609

Z=0,5891

Rappelle le menu (si le témoin **USER** est affiché).

Demande les coordonnées du nouveau système.

Prêt à recevoir P .

 x' y' z'

Dans le nouveau système, un point a pour coordonnées (1, 1, 1). Quelles sont ses coordonnées dans le système initial ?

Frappez

[R/S]

1 [ENTER↑] 1 [ENTER↑]
 1 [B] (↑O)
 [R/S]
 [R/S]

Affichage

↑N ↑O NEW

1,0000

X=2,9117

Y=4,3728

Z=5,8772

Rappelle le menu principal (optionnel).

 x y z

Informations pour la programmation

Vous pouvez utiliser CT comme sous-programme dans vos propres programmes. Il effectue des transformations de coordonnées (rotations et translations) dans trois dimensions. Il prend les valeurs x , y , et z dans la pile du calculateur (registres X, Y, et Z) et donne les valeurs correspondantes dans un autre système, ou les convertit en valeurs correspondantes dans le système initial.

Mémoire minimale pour l'exécution de CT : SIZE 017.

Indicateurs binaires utilisés : 01, 05.

Sous-programme : CT

Registres initiaux	Registres finals	Indicateurs binaires à modifier
Registre X = coordonnée sur x	Registre X = coordonnée transformée sur x	SF 01 pour obtenir les coordonnées dans le nouveau système
Registre Y = coordonnée sur y	Registre Y = coordonnée transformée sur y	CF 01 pour obtenir les coordonnées dans le système initial
Registre Z = coordonnée sur z	Registre Z = coordonnée transformée sur z $R_{00} = (1 - \cos\theta)(n \cdot P)$ $R_{01} = \text{contenu du registre } X$ $R_{02} = \text{contenu du registre } Y$ $R_{03} = \text{contenu du registre } Z$ $R_{04} = P_x \text{ (ou } P_x - T_x \text{ si l'indicateur binaire 01 est armé)}$ $R_{05} = P_y \text{ (ou } P_y - T_y \text{ si l'indicateur binaire 01 est armé)}$ $R_{06} = P_z \text{ (ou } P_z - T_z \text{ si l'indicateur binaire 01 est armé)}$	SF 05 pour effectuer une rotation des coordonnées CF 05 pour ne pas effectuer de rotation des coordonnées
$R_{07} = a \quad (n_x, \text{ le vecteur unitaire de rotation})$ $R_{08} = b \quad (n_y)$ $R_{09} = c \quad (n_z)$ $R_{10} = T_x, \text{ le vecteur de translation}$ $R_{11} = T_y$ $R_{12} = T_z$ $R_{16} = \text{angle de rotation}$	$R_{07} = a \quad (n_x, \text{ le vecteur unitaire de rotation})$ $R_{08} = b \quad (n_y)$ $R_{09} = c \quad (n_z)$ $R_{10} = T_x, \text{ le vecteur de translation}$ $R_{11} = T_y$ $R_{12} = T_z$ $R_{16} = \text{angle de rotation}$	

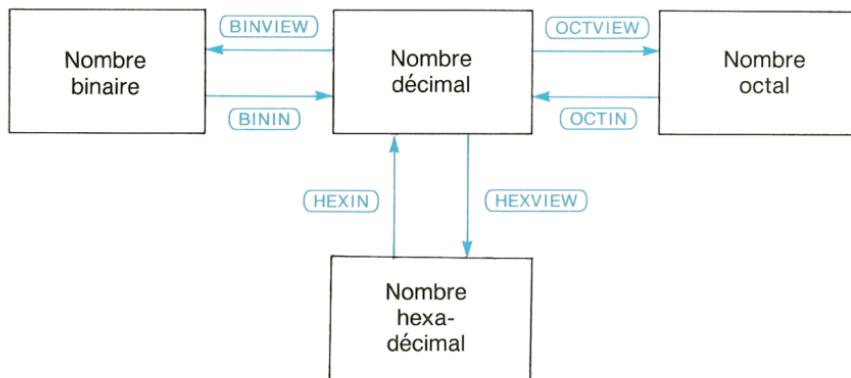
Commentaires. Pour utiliser CT, chargez le vecteur de translation (T), le vecteur unitaire de rotation (N), et l'angle de rotation, armez l'indicateur binaire 01 pour convertir vers le nouveau système ou désarmez-le pour convertir vers le système initial. Armez l'indicateur binaire 05 pour effectuer une rotation sur le vecteur (P). Le résultat est donné dans les registres X, Y, Z et R_{01} , R_{02} , R_{03} .

CHANGEMENTS DE BASE NUMÉRIQUE ET LOGIQUE BOOLEENNE

Ce module comprend plusieurs fonctions pour le calcul et la manipulation de nombres binaires, octaux et hexadécimaux. Vous disposez de six fonctions de conversion entre bases numériques, de quatres fonctions booléennes, et de deux fonctions de manipulation des bits. Toutes ces fonctions utilisent une longueur de mot de 32 bits.

Fonctions de changement de base numérique

Il existe six fonctions pour convertir des nombres en leurs équivalents décimaux, binaires, octaux, et hexadécimaux :



Données autorisées

- Les nombres binaires introduits pour `BININ` doivent être composés de 0 et de 1 ; dix chiffres au maximum.
- Les nombres décimaux introduits pour `BINVIEW` doivent être des entiers compris entre 0 et 1 023. Les non-entiers sont tronqués. La valeur absolue est utilisée.
- Les nombres octaux introduits pour `OCTIN` doivent comprendre des chiffres entre 0 et 7 ; dix chiffres au maximum.

- Les nombres décimaux introduits pour **OCTVIEW** doivent être des entiers entre 0 et 1 073 741 823. Les non-entiers sont tronqués. La valeur absolue est utilisée.
- Les nombres hexadécimaux introduits pour **HEXIN** doivent comprendre des chiffres entre 0 et 9 et des «lettres» entre A et F ; huit caractères au maximum.
- Les nombres décimaux introduits pour **HEXVIEW** doivent être des entiers compris 0 et 4 294 967 295. Les non-entiers sont tronqués. La valeur absolue est utilisée.

Utilisation

- Les fonctions dont le nom se termine par «VIEW» convertissent *l'affichage* de la valeur (décimale) se trouvant dans le registre X (la pile continue toutefois à contenir la valeur décimale). Appuyez sur **←** pour afficher à nouveau le registre X.
- Dans les nombres non-décimaux, les groupes de caractères sont séparés par des virgules pour faciliter la lecture. C'est le format **FIX** en cours qui détermine le nombre de chiffres affichés entre les virgules (ces virgules n'apparaissent que si l'indicateur binaire 29 est armé, ce qui est la condition par défaut).
- Les fonctions dont le nom se termine par «IN» sont des fonctions *préfixes* : vous exécutez *d'abord* la fonction, *puis* vous introduisez votre valeur. Appuyez sur **ENTER** pour voir le résultat.
- Pour annuler une fonction en «IN», appuyez sur **ALPHA ALPHA**.
- Lorsqu'une fonction en «IN» est exécutée dans un programme, elle interrompt l'exécution de ce programme.

Fonctions de changement de base numérique

Fonction	Description
BININ (binaire vers décimal)	Convertit une entrée binaire en valeur décimale, dans le registre X. 1. Exécutez BININ . L'affichage indique _ B . 2. Introduisez un nombre binaire. 3. Appuyez sur ENTER↑ pour obtenir le résultat.
BINVIEW (décimal vers binaire)	Affiche temporairement l'équivalent binaire de la valeur décimale se trouvant dans le registre X. 1. Introduisez la valeur décimale à convertir. 2. Exécutez BINVIEW . 3. L'affichage donne <i>résultat B</i> . 4. Appuyez sur ← pour voir à nouveau le contenu du registre X.

Fonctions de changement de base numérique (suite)

Fonction	Description
OCTIN (octal vers décimal)	Convertit une entrée octale en valeur décimale, dans le registre X. <ol style="list-style-type: none"> Exécutez OCTIN. L'affichage indique _ O. Introduisez un nombre octal. Appuyez sur ENTER↑ pour obtenir le résultat.
OCTVIEW (décimal vers octal)	Affiche temporairement l'équivalent octal de la valeur décimale se trouvant dans le registre X. <ol style="list-style-type: none"> Introduisez la valeur décimale à convertir. Exécutez OCTVIEW. L'affichage donne résultat O. Appuyez sur ← pour voir à nouveau le contenu du registre X.
HEXIN (hexadécimal vers décimal)	Convertit une entrée hexadécimale en une valeur décimale, dans le registre X. <ol style="list-style-type: none"> Exécutez HEXIN. L'affichage indique _ H. Introduisez un nombre hexadécimal. Appuyez sur ENTER↑ pour obtenir le résultat.
HEXVIEW (décimal vers hexadécimal)	Affiche temporairement l'équivalent hexadécimal d'une valeur décimale se trouvant dans le registre X. <ol style="list-style-type: none"> Introduisez la valeur décimale à convertir. Exécutez HEXVIEW. L'affichage donne résultat H. Appuyez sur ← pour voir à nouveau le contenu du registre X.

Fonctions booléennes

Ce groupe de fonctions couvre la logique booléenne, la vérification et la permutation circulaire des bits.

Données autorisées

Ces fonctions opèrent sur les nombres décimaux compris entre 0 et 4 294 967 295 (32 bits, entiers sans signe). Les non-entiers sont tronqués. En cas de valeur négative, c'est la valeur absolue qui est utilisée.

Utilisation

Le résultat d'une opération booléenne est placé dans le registre X. La valeur se trouvant initialement dans le registre X est stockée dans le registre LAST X *sauf* dans le cas de **BIT?**, qui n'affecte ni LAST X ni la pile. Toutes les autres fonctions à deux paramètres font descendre la pile.

Fonctions booléenne

Fonction	Description
AND	Calcule le ET logique de x et y.
OR	Calcule le OU inclusif logique de x et y.
XOR	Calcule le OU exclusif logique de x et y.
NOT	Calcule le NON logique de x.
BIT? (vérification de bit)	Prend la représentation binaire du nombre stocké dans le registre Y et vérifie le bit dont le numéro est donné dans le registre X. Si le bit vaut un, le calculateur affiche YES ; si le bit vaut zéro, le calculateur affiche NO . Dans un programme, BIT? est une fonction conditionnelle obéissant à la règle «exécution si la condition est vraie» : un bit à un entraîne l'exécution du pas de programme suivant, alors qu'avec un bit à zéro, le pas de programme suivant est sauté.
ROTXY (permutation circulaire de Y d'un nombre de bits spécifiés par X)	Prend la représentation binaire du nombre se trouvant dans le registre Y et lui fait subir une permutation circulaire en décalant les bits vers la droite du nombre de bits spécifiés dans le registre X. Une permutation circulaire de $(32 - x)$ bits vers la droite équivaut à une permutation circulaire de x bits vers la gauche.

Exemples

Quels sont les équivalents binaires, octaux et hexadécimaux de 65_{10} ? Choisissez **FIX 4** pour avoir un séparateur (virgule) tous les quatre chiffres.

Frappez

FIX 4

65

XEQ **BINVIEW**
XEQ **OCTVIEW**
XEQ **HEXVIEW**

Affichage

65__
100,0001 B
101 O
41 H

Valide le format d'affichage utilisé ici.

Binaire.

Octal.

Hexadécimal.

Quel est le résultat, en base huit (octal), de l'opération suivante : permutation circulaire de $FA407_{16}$ de six bits vers la droite, addition de 100100_2 au résultat, puis calcul du ET logique (AND) de la somme obtenue et de 25_{10} ?

Frappez

[XEQ **HEXIN**

FA407

[ENTER]

6

[XEQ **ROTXY**

[XEQ **BININ**

100100

[+]

25

[XEQ **AND**

[XEQ **OCTVIEW**

Affichage

_ H

FA407_ H

1.025.031,000

6_

469.778.064,0

_ B

10,0100_ B

469.778.100,0

25_

16,0000

20 O

Equivalent décimal de $FA407_{16}$.

Permutation circulaire de la valeur de six bits vers la droite.

Ajoute la valeur binaire introduite à la valeur précédente.

ET logique de 25 et du résultat précédent.

Résultat octal.

AJUSTEMENT DE COURBES

Le programme CFIT collecte des données statistiques (x_i, y_i) et ajuste à ces données celle des quatre courbes ci-dessous que vous choisissez ou celle de meilleur ajustement. Cette dernière est considérée comme celle ayant le coefficient de détermination, r^2 , le plus élevé pour les données.

- Droite (régression linéaire), $y = a + bx$
- Courbe exponentielle, $y = ae^{bx}$ (avec $a > 0$)
- Courbe logarithmique, $y = a + b(\ln x)$
- Courbe de puissance, $y = ax^b$ (avec $a > 0$)

Le programme permet de calculer les valeurs de a , b , r^2 , et de \hat{y} , (estimation de la valeur de y) pour un x donné.

Équations utilisées par l'algorithme

Les coefficients de régression a et b sont obtenus par résolution des équations linéaires suivantes, où n est le nombre total de couples de données.

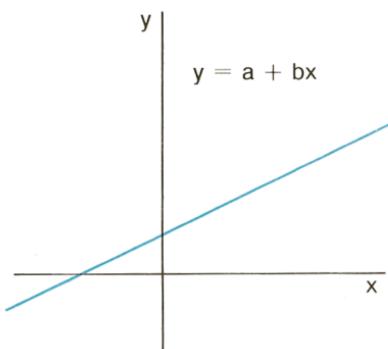
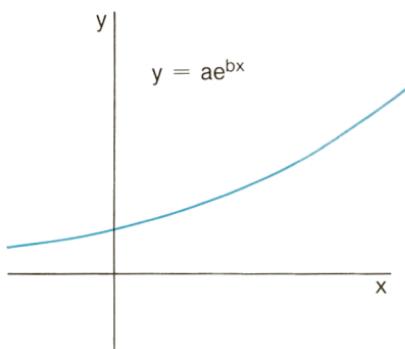
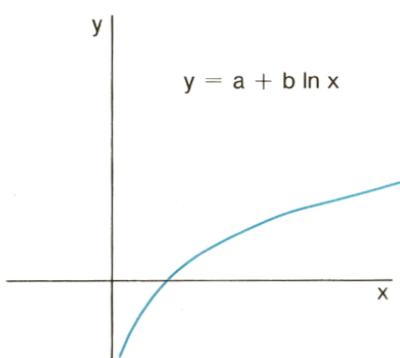
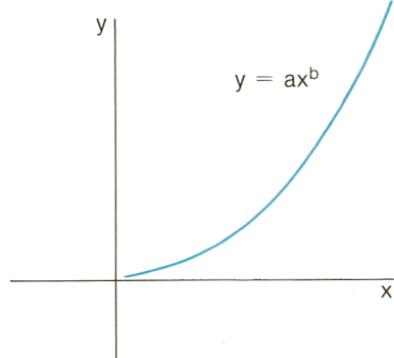
$$\begin{aligned} A_n + b \sum X_i &= \sum Y_i \\ A \sum X_i + b \sum (X_i)^2 &= \sum (Y_i X_i) \end{aligned}$$

Définitions des variables de régression

Régression	A	X_i	Y_i
Linéaire	a	x_i	y_i
Exponentielle	$\ln a$	x_i	$\ln y_i$
Logarithmique	a	$\ln x_i$	y_i
Puissance	$\ln a$	$\ln x_i$	$\ln y_i$

Le coefficient de détermination est

$$r^2 = \frac{A \sum Y_i + b \sum (X_i Y_i) - \frac{1}{n} (\sum Y_i)^2}{\sum (Y_i)^2 - \frac{1}{n} (\sum Y_i)^2}$$

Régression linaire**Courbe exponentielle****Courbe logarithmique****Courbe de puissance**

Données autorisées

Le programme CFIT évalue les données introduites par la méthodes des moindres carrés, en utilisant soit l'équation initiale (droite et courbe logarithmique), soit les équations transformées (courbes exponentielle et de puissance).

Toutes les valeurs (x_i, y_i) doivent être positives et non nulles, sinon il en résulte un message d'erreur **DATA ERROR** (erreur de donnée).

Plus la différence entre les valeurs de x et de y est petite, plus la précision des coefficients de régression diminue.

Remarquez également que les résultats peuvent être imprécis si une variable est beaucoup plus grande que l'autre ou varie beaucoup plus rapidement (ceci se produit lorsque le calculateur doit maintenir une précision de plus de dix chiffres significatifs, ce qui lui est impossible). Si les

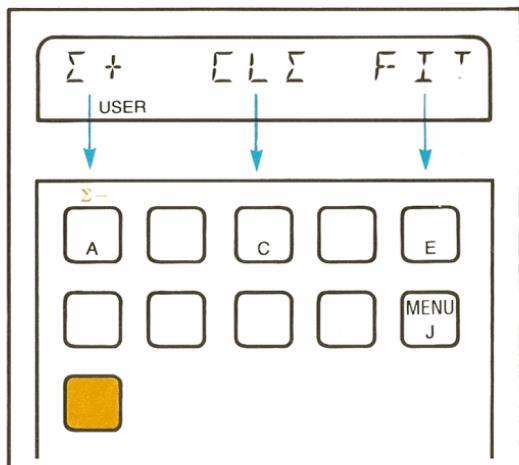
valeurs de vos données sont de ce type, vous devez utiliser des méthodes de pondération pour garantir la précision des résultats. Vous trouverez une description des méthodes de pondération dans de nombreux textes statistiques.

Un message **DATA ERROR** sera affiché si vous essayez d'ajuster une courbe contenant seulement une seule observation, ou si vous utilisez des valeurs négatives ou nulles.

Utilisation

- Le programme CFIT commence en affichant son menu principal, $\Sigma +$ **CLΣ FIT**. Ce menu sert à l'introduction de vos données statistiques : $\Sigma +$ pour la saisie (y d'abord, puis x), $\Sigma -$ pour supprimer, et **CLΣ** pour effacer les anciennes données statistiques. **FIT** affiche le menu d'ajustement de courbes.
- Le menu d'ajustement de courbes, **L EX LOG P B**, vous offre un choix de courbes à ajuster à vos données : **L** pour Linéaire, **EX** pour Exponentielle, **LOG** pour Logarithmique, **P** pour Puissance, et **B** pour la courbe de meilleur ajustement (Best fit). Le meilleur ajustement donne la courbe qui s'ajuste le mieux à vos données.
- Une fois que vous avez choisi la courbe à ajuster, des pressions successives sur **R/S** affichent les paramètres de la régression. Une pression sur **J** réaffiche le menu principal.

Menu principal



$\Sigma +$ = Saisie d'un couple de données

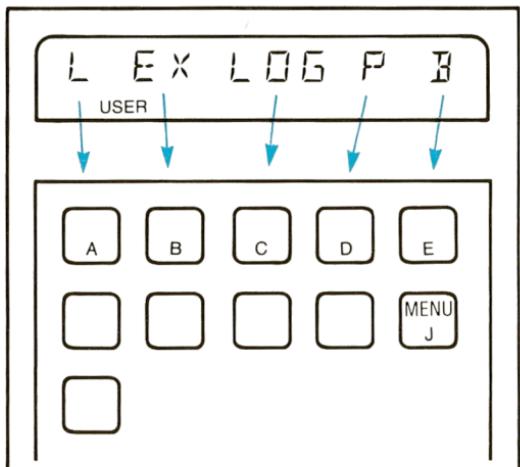
$\Sigma -$ = Effacement des anciennes données

FIT = Affichage du menu d'ajustement

$\Sigma -$ = Suppression d'un couple erroné

Ce menu vous montre quelles touches correspondent à quelles fonctions dans CFIT. Appuyez sur **J** pour réafficher ce menu à tout moment. Ceci ne perturbera en aucun cas le programme.

Menu d'ajustement de courbes



L = «Linear»
 Droite
 $E X$ = Courbe exponentielle
 $\Box \Box G$ = Courbe logarithmique
 P = Courbe de puissance
 B = «Best Fit»
 Courbe de meilleur
 ajustement

Pour effacer le menu à n'importe quel moment, appuyez sur \leftarrow . Ceci affiche le contenu du registre X, sans toutefois mettre fin au programme. Vous pouvez alors effectuer des calculs, puis rappeler le menu principal en appuyant sur J (vous n'avez cependant *pas besoin* d'effacer l'affichage issu du programme avant d'effectuer vos calculs).

Instructions d'utilisation de CFIT

Instructions	Frappez :	Taille : 018
1. Lancez le programme CFIT. Les éléments du menu indiquent l'emplacement des fonctions sur la rangée supérieure de touches.	$\text{XEQ } \text{CFIT}^*$	$\Sigma+ \text{ CL}\Sigma \text{ FIT}$
2. Efface les anciennes données statistiques (ceci n'est pas nécessaire si vous venez juste d'exécuter CFIT , qui efface automatiquement les anciennes données).	$\text{C } (\text{CL}\Sigma)$	$\Sigma+ \text{ CL}\Sigma \text{ FIT}$
3. Introduisez vos couples de données. Répétez l'opération pour chaque couple.	$y \text{ ENTER} \uparrow$ $x \text{ A } (\Sigma+) \dagger$	y $\Sigma+ \text{ CL}\Sigma \text{ FIT}$
4. Pour voir combien de couples de données (observations) vous avez déjà introduits, effacez l'affichage (optionnel).	\leftarrow J	n $\Sigma+ \text{ CL}\Sigma \text{ FIT}$

Instructions d'utilisation de CFIT (suite)

Instructions	Frappez :	Affichage
5. Pour corriger un couple de données, ré-introduisez ce couple afin de le supprimer. Introduisez alors le couple correct (point 3).	y_k ENTER ↑ x_k [A] (Σ-)	y_k Σ CLΣ FIT
6. Affichez le menu d'ajustement de courbes.	E (FIT)	L EX LOG P B
7. Choisissez la courbe que vous voulez ajuster.	A (L) B (EX) C (LOG) D (P) E (B)	LIN EXP LOG POW (le «meilleur ajustement»)
8. Cherchez les valeurs de a , de b , de r^2 .	R/S ‡ R/S ‡ R/S ‡	a = résultat b = résultat $Rt2$ = résultat
9. Trouvez l'estimation, \hat{y} . Répétez autant de fois que nécessaire.	R/S ‡ X R/S R/S ‡	X=? Y=résultat X=?
10. Pour recommencer (rappelez le menu principal) :	J	Σ CLΣ FIT

* Pour exécuter un programme, frappez **XEQ** **ALPHA** nom alphabétique du programme **ALPHA** ou redéfinissez une touche du clavier personnel (USER) en lui affectant cette fonction.

† Si une imprimante est connectée au calculateur, cette étape peut vous fournir une copie sur papier des valeurs que vous venez d'introduire. Consultez le manuel d'utilisation de l'imprimante.

‡ Cette séquence est inutile si une imprimante est connectée au calculateur, car l'imprimante imprime automatiquement les résultats puis affiche le menu.

Remarques

Ce programme affecte des *labels locaux alphabétiques* (cf. manuel d'utilisation du HP-41) aux touches **A**-**E**, **[A]**, et **J**. Ces affectations locales du programme *n'ont pas la priorité* sur les fonctions que vous pouvez avoir déjà affectées à ces mêmes touches sur le clavier personnel (USER), auquel cas le programme ne peut pas fonctionner. *Vous devez donc effacer toutes les fonctions que vous avez affectées à ces touches sur le clavier personnel avant d'utiliser ce programme*, et éviter de redéfinir ces touches à l'avenir.

CFIT modifie l'emplacement des registres statistiques. Pour accéder à des données dans les registres statistiques *après avoir utilisé ce programme*, vous devez repositionner ces registres à un emplacement connu avec la fonction **SREG** (cf. manuel d'utilisation du HP-41), même si vous voulez seulement les remettre à leur emplacement par défaut, dans R₁₁–R₁₆. Pour accéder à des données statistiques stockées *par ce programme*, consultez les «Informations pour la programmation» en fin de chapitre.

Exemples

Ajustez une droite à l'ensemble d'observations ci-dessous et calculez \hat{y} pour $x = 37$ et $x = 35$.

x	40,5	38,6	37,9	36,2	35,1	34,6
y	104,5	102	100	97,5	95,5	94

Frappez

FIX 4

XEQ **SIZE** 018

XEQ **CFIT**

104.5 **ENTER** 40.5

A $(\Sigma +)$

102 **ENTER** 38.6

A $(\Sigma +)$

100 **ENTER** 37.9

A $(\Sigma +)$

97.5 **ENTER** 36.2

A $(\Sigma +)$

95.5 **ENTER** 35.2

A $(\Sigma +)$

95.5 **ENTER** 35.2

A $(\Sigma -)$

Affichage

$\Sigma + \text{ CLΣ FIT}$

40,5 $\underline{}$
 $\Sigma + \text{ CLΣ FIT}$

38,6 $\underline{}$
 $\Sigma + \text{ CLΣ FIT}$

37,9 $\underline{}$
 $\Sigma + \text{ CLΣ FIT}$

36,2 $\underline{}$
 $\Sigma + \text{ CLΣ FIT}$

35,2 $\underline{}$
 $\Sigma + \text{ CLΣ FIT}$

35,2 $\underline{}$
 $\Sigma + \text{ CLΣ FIT}$

Valide le format d'affichage utilisé ici.

Optionnel—définit le nombre de registres de stockage nécessaires. Superflu si l'affectation mémoire est déjà suffisante ($\text{SIZE} \geq 018$).

Lancer le programme efface aussi les anciennes données statistiques.

Saisie du 1er couple de données, en commençant par y .
2ème couple.

etc.

Erreur dans la saisie de x !

Supprimez le couple erroné.

Frappez

95.5 [ENTER↑] 35.1

[A] ($\Sigma +$)

94 [ENTER↑] 34.6

[A] ($\Sigma +$)

[E] (FIT)

[A] (L)

[R/S]

[R/S]

[R/S]

[R/S]

37 [R/S]

[R/S]

35 [R/S]

[R/S]

[R/S]

[R/S]

Introduisez l'ensemble d'observations suivantes et trouvez la courbe qui s'y ajuste le mieux. Calculez ensuite \hat{y} pour $x = 1,5$ et $x = 2$.

x	0,72	1,31	1,95	2,58	3,14
y	2,16	1,61	1,16	0,85	0,5

Frappez

[R/S]

[C] (CLΣ)

2.16 [ENTER↑] 0.72

[A] ($\Sigma +$)

1.61 [ENTER↑] 1.31

[A] ($\Sigma +$)

1.16 [ENTER↑] 1.95

[A] ($\Sigma +$)**Affichage**35,1_ $\Sigma +$ CLΣ FIT34,6_ $\Sigma +$ CLΣ FIT

L EX LOG P B

LIN

a=33,5271

b= 1,7601

R↑2=0,9909

X=?

Y=98,6526

X=?

Y=95,1323

X=?

L EX LOG P B

 $\Sigma +$ CLΣ FIT

Introduisez le couple correct.

Menu d'ajustement de courbes.

Choix de la droite.

Demande la valeur de x pour laquelle vous voudriez estimer y .
 \hat{y} .

Réaffiche le menu d'ajustement. Vous pouvez ajuster une autre courbe.

Affiche le menu principal ; vous pouvez commencer un nouveau problème.

Affichage $\Sigma +$ CLΣ FIT $\Sigma +$ CLΣ FIT0,72_ $\Sigma +$ CLΣ FIT1,31_ $\Sigma +$ CLΣ FIT1,95_ $\Sigma +$ CLΣ FIT2,58_ $\Sigma +$ CLΣ FIT

Affichez le menu principal.

Efface les données du premier exemple.

Introduisez le premier couple de données.

Frappez

0.85 [ENTER↑] 2.58

[A] (Σ+)

0.5 [ENTER↑] 3.14

[A] (Σ+)

[E] (FIT)

[E] (B)

[R/S]

[R/S]

[R/S]

[R/S]

1.5 [R/S]

[R/S]

2 [R/S]

[R/S]

[R/S]

Affichage

2,58_

Σ+ CLΣ FIT

3,14_

Σ+ CLΣ FIT

L EX LOG P B

LOG

a = 1,8515

b = -1,1021

R↑2 = 0,9893

X = ?

Y = 1,4046

X = ?

Y = 1,0875

X = ?

L EX LOG P B

La courbe qui s'ajuste le mieux est une courbe logarithmique.

Informations pour la programmation

Vous pouvez utiliser les sous-programmes AΣ, DΣ, FIT, et BFIT dans vos propres programmes.

- AΣ prend le couple de données se trouvant dans les registres X et Y et l'ajoute à un jeu de registres statistiques permettant de réaliser une évaluation statistique des données introduites (sommation, moyenne, écart-type).
- DΣ supprime du jeu de registres statistiques le couple de données se trouvant dans les registres X et Y.
- FIT ajuste une courbe de type 1 à 4 aux données statistiques stockées par le programme CFIT ou par les sous-programmes AΣ et DΣ.
- BFIT trouve la courbe (parmi les types de courbe 1 à 4) qui s'ajuste le mieux aux données statistiques stockées par le programme CFIT ou par les sous-programmes AΣ et DΣ.

Taille mémoire minimale pour l'exécution du programme : SIZE 018

Indicateurs binaires utilisés : BFIT et FIT utilisent 01, 02, 03, 04, 06, 07.

AΣ et DΣ n'utilisent aucun indicateur binaire.

Sous-programmes AΣ et DΣ

Registres initiaux	Registres finals	Indicateurs binaires à modifier
Registre Y : valeur de y		
Registre X : valeur de x		
$R_{04} = 0$	$R_{04} = \Sigma(y \ln x)$	
$R_{05} = 0$	$R_{05} = \Sigma(x \ln y)$	
$R_{06} = 0$	$R_{06} = \Sigma y$	
$R_{07} = 0$	$R_{07} = \Sigma y^2$	
$R_{08} = 0$	$R_{08} = \Sigma x$	
$R_{09} = 0$	$R_{09} = \Sigma x^2$	
$R_{10} = 0$	$R_{10} = \Sigma(xy)$	
$R_{11} = 0$	$R_{11} = n$	
$R_{12} = 0$	$R_{12} = \Sigma(\ln y)$	
$R_{13} = 0$	$R_{13} = \Sigma(\ln y)^2$	
$R_{14} = 0$	$R_{14} = \Sigma(\ln x)$	
$R_{15} = 0$	$R_{15} = \Sigma(\ln x)^2$	
$R_{16} = 0$	$R_{16} = \Sigma(\ln x)(\ln y)$	
$R_{17} = 0$	$R_{17} = n$, et temporairement Σy	

Sous-programme FIT

Registres initiaux	Registres finals	Indicateurs binaires à modifier
Registre X = 1 = linéaire 2 = exponentielle 3 = logarithmique 4 = puissance		CF 01
	$R_{00} = 1, 2, 3, \text{ ou } 4$	CF 02
	$R_{01} = a$	CF 03
	$R_{02} = b$	CF 04
	$R_{03} = r^2$	
$R_{04}-R_{17}$: tous les registres statistiques sont les mêmes que ci-dessus dans AΣ.		

Commentaires. Après avoir chargé les registres statistiques en utilisant $A\Sigma$ et $D\Sigma$, placez dans le registre X le numéro de la courbe à ajuster (1, 2, 3, 4) et exécutez **FIT**. Ce programme arme l'indicateur binaire 07 et arme un indicateur binaire (01–04) correspondant au type de la courbe. Il stocke a , b , et r^2 dans R_{01} , R_{02} , et R_{03} .

Sous-programme **BFIT**

Registres initiaux	Registres finals	Indicateurs binaires à modifier
	$R_{00} = 1, 2, 3, \text{ ou } 4$	CF 01
	$R_{01} = a$	CF 02
	$R_{02} = b$	CF 03
	$R_{03} = r^2$	CF 04
$R_{04}\text{--}R_{17}$: tous les registres statistiques sont les mêmes que ci-dessus dans $A\Sigma$ et FIT .		

Commentaires. Après avoir chargé les registres statistiques en utilisant $A\Sigma$ et $D\Sigma$, exéutez **BFIT**, qui trouvera le meilleur ajustement d'une courbe linéaire, exponentielle, logarithmique ou de puissance. **BFIT** arme l'indicateur binaire 01 (droite), 02 (courbe exponentielle), 03 (courbe logarithmique), ou 04 (courbe de puissance), stocke le numéro correspondant à la courbe dans R_{00} , et stocke a , b , et r^2 dans R_{01} , R_{02} , et R_{03} .

VALEUR DE L'ARGENT DANS LE TEMPS

Le programme TVM résoud des problèmes combinant temps, argent, et taux d'intérêts—les fonctions d'intérêts composés. Les variables suivantes peuvent être des données à introduire ou des résultats.

- N* Nombre de périodes de composition ou de paiements (pour un prêt sur 30 ans avec paiements mensuels, $N = 12 \times 30 = 360$).
- I* Taux d'intérêt périodique en pourcentage.
- PV* Valeur actuelle (peut aussi être un flux—transfert d'argent—initial ou la valeur escomptée d'une série de flux futurs). Le transfert se produit toujours au début de la première période.
- PMT* Paiement périodique.
- FV* Valeur future (ceci peut aussi être un flux final ou la valeur composée d'une série de flux). Pour ce flux, le transfert se produit toujours à la fin de la *Nième* période.

Vous pouvez spécifier que les paiements doivent avoir lieu *en fin* de période de composition (mode Fin de Période—End Mode) ou *en début* de période (mode Début de Période—Begin Mode). Si le paiement s'effectue en début de période, l'indicateur binaire 00 est armé. Les paiements en fin de période sont courants dans les prêts hypothécaires, et ceux en début de période dans les opérations de crédit-bail.

Remarque sur les taux d'intérêt

Dans toutes les transactions financières (placements, épargne liquide, emprunts), les taux d'intérêt annoncés sont toujours annuels. Les modalités de calcul des intérêts sont très variées et en général basées sur des périodes de composition non annuelles. Ces périodes sont semestrielles, trimestrielles, mensuelles ou par quinzaine.

Pour plus de précisions sur la définition et les méthodes de calcul des différents taux, consultez l'annexe A, «Les taux d'intérêt».

Equation utilisée par l'algorithme

$$0 = PV + (1 + ip) PMT \left[\frac{1 - (1 + i)^{-N}}{i} \right] + FV (1 + i)^{-N}$$

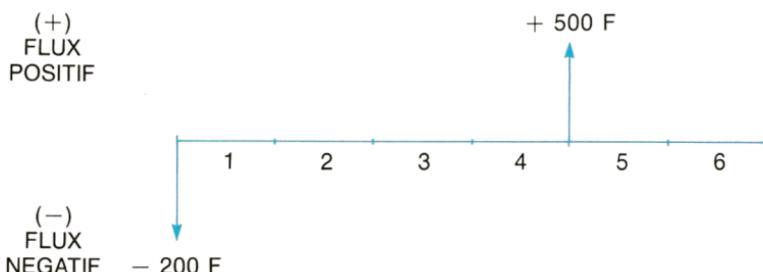
où i est le taux d'intérêt périodique *sous forme fractionnaire* ($i = I/100$),
 $p = 1$ en mode Début de Période ou 0 en mode Fin de Période.

Valeurs autorisées

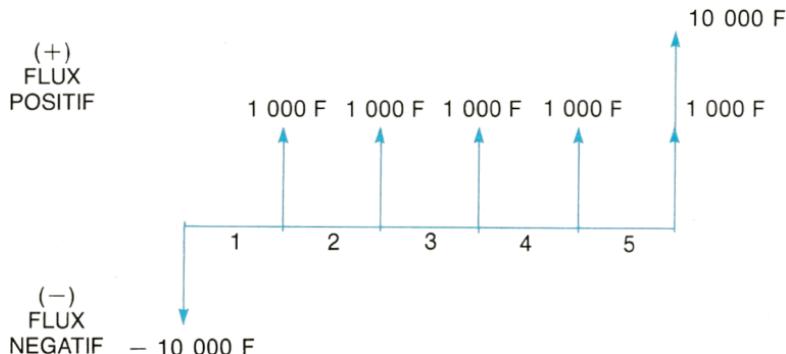
Utilisez un *diagramme des flux* pour déterminer vos flux (transferts d'argent) et s'ils doivent être spécifiés comme *positifs* ou *négatifs*.

Le diagramme des flux est une ligne représentant le temps et divisée en périodes de composition. Les flux sont représentés par des flèches verticales : une flèche vers le haut est *positive* et représente de l'argent *reçu*, et une flèche vers le bas est *négative* et représente de l'argent *déboursé*.

Par exemple, la ligne ci-dessous est divisée en six périodes et présente un flux initial négatif de 200 F et un flux positif de 500 F à la fin de la quatrième période (puisque il n'y a pas de paiements échelonnés, vous pouvez être ici indifféremment en mode Début de Période ou Fin de Période lorsque vous calculez *PV* ou *FV*).



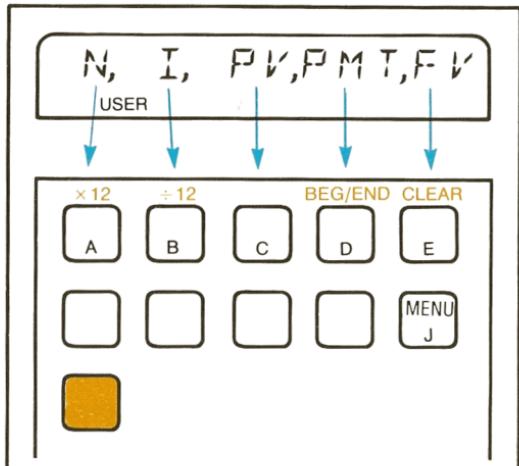
La ligne ci-dessous est divisée en cinq périodes et présente un flux initial négatif de 10 000 F et un flux positif de 1 000 F à la fin de chaque période, avec un flux positif supplémentaire de 10 000 F à la fin de la cinquième période.



Utilisation

- Le programme TVM calcule l'une quelconque des variables N , I , PV , PMT , ou FV lorsque les trois ou quatre autres sont fournies (vous devez cependant fournir soit N soit I). L'ordre dans lequel vous introduisez les données est sans importance.*
- Vous devez effacer les données financières (■ E) avant de commencer des calculs complètement nouveaux, sinon les données précédentes qui n'ont pas été remplacées seront utilisées dans les calculs. Les données financières sont aussi effacées si vous exécutez à nouveau le programme.
- N'oubliez pas de spécifier les flux positifs (flèche vers le haut) comme des valeurs positives et les flux négatifs (flèche vers le bas) comme des valeurs négatives. Les résultats sont également des valeurs positives ou négatives.
- Vérifiez que le mode de paiement correspond à celui que vous désirez. Si vous voyez le témoin de l'indicateur binaire 00 (un petit 0 en-dessous de la ligne d'affichage), cela indique que vous êtes en mode Début de Période. Sinon, vous êtes en mode Fin de Période. Pour passer d'un mode à l'autre, appuyez sur ■ D ; l'affichage montre alors le mode que vous venez de choisir : **BEGIN MODE** (Début de Période) ou **END MODE** (Fin de période). Le mode par défaut est le mode Fin de Période (indicateur binaire 00 désarmé).
- N'oubliez pas que le taux d'intérêt doit correspondre aux nombre de périodes de composition (un taux *annuel* n'est approprié que si le nombre de périodes de composition est aussi égal au nombre d'années).
- Vous voudrez peut-être choisir un format d'affichage avec deux chiffres après la virgule (■ FIX 2).

* Si vous utilisez seulement quatre variables, la cinquième doit être égale à zéro. Toutes les variables sont mises à zéro lorsque vous exécutez TVM pour la première fois ou lorsque vous effacez les données financières (■ E) ; dans ces deux cas, vous n'avez donc pas besoin de frapper un zéro pour la cinquième variable.



N = Nombre de périodes de composition

I = Taux d'intérêt périodique

PV = «Present Value» Valeur actuelle

PMT = Paiement périodique

FV = «Future Value» Valeur future

BEG/END = «Begin/End» Choisis le mode de paiement (BEG = Début, END = Fin de période)

CLEAR = Efface les anciennes données

Ce menu vous montre quelle touche correspond à quelle fonction dans TVM. Appuyez sur **J** pour rappeler ce menu à tout moment ; cela ne perturbera en aucun cas le programme.

Pour effacer le menu, appuyez sur **⬅**. Le contenu du registre X est alors affiché, mais cela ne met pas fin au programme. Vous pouvez effectuer des calculs, puis rappeler le menu principal en appuyant sur **J** (vous n'avez toutefois *pas besoin* d'effacer l'affichage issu du programme ni de rappeler le menu avant d'effectuer vos calculs).

Instructions d'utilisation de TVM

Instructions	Frappez :	Affichage
1. Lancez le programme TVM. Le menu affiché indique l'emplacement des fonctions N , I , PV , PMT , et FV sur les deux rangées supérieures de touches.	XEQ TVM *	N, I, PV,PMT,FV
2. Vérifiez le mode de paiement (la présence du témoin 0 signifie que vous êtes en mode Début de Période—Begin Mode ; son absence que vous êtes en mode Fin de Période—End Mode). Changez de mode si nécessaire.	END MODE ou BEGIN MODE D (fait passer d'un mode à l'autre) R/S ou J	N, I, PV,PMT,FV
3a. Introduisez le nombre de périodes de composition, N , sauf si N est ce que vous recherchez. L'étape 3b est un raccourci pour calculer le nombre de mois dans un nombre d'années donné.	N A (N)	N= N†

Instructions d'utilisation de TVM (suite)

Instructions	Frappez :	Affichage
3b. Alternative à 3a : si vous travaillez avec des paiements mensuels ou avec des périodes mensuelles de composition sur un nombre connu d'années, cette étape calcule et saisit automatiquement N (sous la forme $12 \times$ nb d'années). Introduisez le nombre d'années, n .	n	$N = 12 \times n$
4a. Introduisez le taux d'intérêt périodique, I , sauf si I est ce que vous recherchez. L'étape 4b vous permet de calculer directement un taux d'intérêt mensuel à partir d'un taux d'intérêt annuel <i>nominal</i> donné (méthode utilisée aux Etats-Unis).	I	$I = I$
4b. Alternative à 4a : si vous travaillez avec des périodes de composition mensuelles et taux d'intérêt annuel <i>NOMINAL</i> , ceci calcule et saisit automatiquement I (sous la forme d'un taux annuel en pourcentage $\div 12$). Frappez le taux annuel <i>NOMINAL</i> en pourcentage, TAN .	TAN	$I = TAN \div 12$
5. Introduisez la valeur actuelle, PV , sauf si PV est ce que vous recherchez ou si elle n'est pas une variable applicable à votre exemple.	PV	$PV = \text{valeur fournie}$
6. Introduisez le montant du paiement, PMT , sauf si PMT est ce que vous cherchez ou n'est pas applicable à votre exemple.	PMT	$PMT = \text{valeur fournie}$
7. Introduisez la valeur future, FV , sauf si FV est ce que vous recherchez ou n'est pas applicable à votre exemple.	FV	$FV = \text{valeur fournie}$
8. Trouvez maintenant la variable restante en appuyant sur la touche correspondante.	ou ou ou ou 	$N = \text{résultat}$ ou $I = \text{résultat}$ ou $PV = \text{résultat}$ ou $PMT = \text{résultat}$ ou $FV = \text{résultat}$
9. Pour ré-afficher la valeur d'une variable à n'importe quel moment :	à	valeur
10. Pour rappeler le menu principal (N, I, PV, PMT, FV) à tout moment (sans affecter les valeurs introduites ou les calculs) :		N, I, PV, PMT, FV
11. Effacez les anciennes données financières avant de commencer un nouveau problème.		N, I, PV, PMT, FV

* Pour exécuter un programme, appuyez sur **[XEQ]** **[ALPHA]** nom alphabétique du programme **[ALPHA]** ou redéfinissez une touche du clavier personnel (USER) en lui affectant cette fonction.

† Si une imprimante est connectée au calculateur, l'affichage revient automatiquement au menu principal (**N, I, PV, PMT, FV**) une fois que la dernière valeur introduite a été imprimée.

Remarques

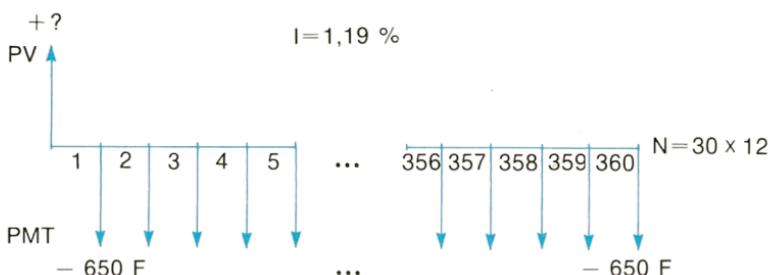
Ce programme affecte des labels locaux alphabétiques (comme expliqué dans le manuel d'utilisation du HP-41) aux touches **A**–**E** (utilisées seules et avec la touche préfixe jaune **Y**, sauf **Y****C**), et à la touche **J**. Ces affectations locales du programme *n'ont pas la priorité* sur les fonctions que vous avez éventuellement déjà affectées à ces mêmes touches, auquel cas le programme ne peut pas fonctionner. *Vous devez donc effacer toutes les affectations du clavier personnel (USER) s'appliquant à ces touches avant d'utiliser le programme*, et éviter de redéfinir ces touches à l'avenir.

Les touches correspondant aux variables financières ne stockent une valeur que si vous l'introduisez à partir du clavier. Si, par exemple, vous rappelez une valeur stockée dans un registre puis appuyez sur une touche correspondant à une des variables, le programme calcule cette variable au lieu de stocker la valeur rappelée. Pour stocker une valeur qui a été placée dans le registre X sans être frappée au clavier, appuyez sur **STO** avant d'appuyer sur la touche de la variable correspondante.

Exemples

Un emprunteur peut effectuer des paiements mensuels de 650 F pour un prêt hypothécaire sur 30 ans, à un taux actuarial annuel de 15,25 %. Combien peut-il emprunter ? Le premier paiement a lieu un mois après la date où l'argent lui a été prêté (mode Fin de Période). Vous trouverez en annexe A la formule permettant de calculer l'intérêt actuarial périodique (mensuel) ainsi que la méthode à utiliser avec votre calculateur pour ce calcul. L'intérêt actuarial mensuel est ici 1,19 %.

Diagramme des flux, exemple 1



Frappez**FIX** 2**XEQ** **SIZE** 010**XEQ** **TVM**30 **A****R/S**1.19 **B****R/S**650 **CHS** **D** **(PMT)****R/S****C** **(PV)****Affichage****N, I, PV,PMT,FV****N=360.00****N, I, PV,PMT,FV****I=1,19****N, I, PV,PMT,FV****PMT= -650,00****N, I, PV,PMT,FV****PV=53.849,47**

Valide le format d'affichage utilisé ici. Optionnel—définit le nombre de registres de stockage requis par le programme. Superflu si l'affectation mémoire est déjà $\text{SIZE} \geq 010$.

Lance le programme et efface les anciennes valeurs financières. Mode Fin de Période automatiquement validé.

Nombre total de périodes.

Rappelle le menu (optionnel).

Taux d'intérêt actuel mensuel.

Rappelle le menu (optionnel).

Paiement mensuel.

Rappelle le menu (optionnel).

Montant maximal de l'emprunt.

Si l'emprunt nécessaire n'est que de 53 500 F, quel est le paiement mensuel ? (Changez *PV*, laissez les autres variables, et cherchez *PMT*.)

Frappez**R/S**53500 **C** **(PV)****R/S****D** **(PMT)****Affichage****N, I, PV,PMT,FV****PV=53.500,00****N, I, PV,PMT,FV****PMT= -645,78**

Rappelle le menu (optionnel).

Montant de l'emprunt requis.

Rappelle le menu (optionnel).

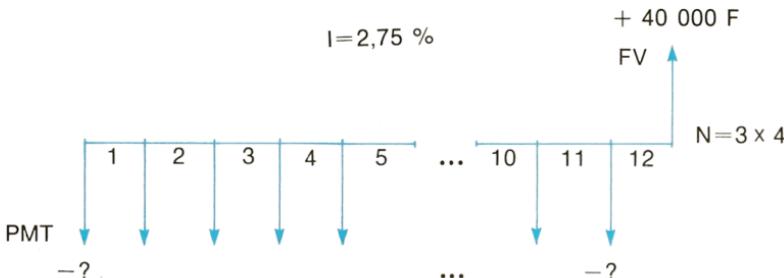
Paiement mensuel.

Remarquez que lorsque vous appuyez sur l'une des touches **A**–**E** après avoir introduit une valeur au clavier, le calculateur stocke cette valeur dans la variable choisie. Toutefois, lorsque vous appuyez sur l'une des touches **A**–**E** sans avoir introduit une valeur au clavier auparavant, le calculateur calcule une valeur pour la variable choisie.

150 Valeur de l'argent dans le temps

Combien d'argent doit être versé chaque trimestre sur un compte d'épargne pour accumuler 40 000 F en 3 ans ? Le compte d'épargne rapporte 11,46 % (taux annuel actuarial), soit 2,75 % (taux actuarial) par trimestre, et les dépôts commencent immédiatement. (Cf. Annexe A pour savoir comment convertir le taux d'intérêt annuel en taux trimestriel.)

Diagramme des flux, exemple 2



Frappez

E

Affichage

N, I, PV,PMT,FV

Efface les données financières (si vous êtes dans TVM).

D

BEGIN MODE

Mode Début de Période (le témoin 0 devrait apparaître).

R/S

N, I, PV,PMT,FV

Nombre total de dépôts.

3 [ENTER] 4 [X]

12,00

Taux actuarial trimestriel.

A (N)

N=12,00

Objectif.

2.75 [B] (I)

2,75

Dépôt trimestriel requis.

40000 [E] (FV)

I=2,75

FV=40.000,00

[D] (PMT)

PMT= -27.822,24

Informations pour la programmation

Vous pouvez utiliser les sous-programmes suivants dans vos propres programmes. Ils calculent le nombre de périodes, le taux d'intérêt, la valeur actuelle, le paiement ou la valeur future lorsque les quatre autres paramètres sont fournis.

Taille mémoire minimale requise : SIZE 010.

Indicateurs binaires utilisés : 00, 25.

Sous-programmes	Registres initiaux	Registres finals	Indicateurs binaires à modifier
N (nombre de périodes)	$R_{02} = I$ $R_{03} = PV$ $R_{04} = PMT$ $R_{05} = FV$	Registre X = N $R_{01} = N$ $R_{02} = I$ $R_{03} = PV$ $R_{04} = PMT$ $R_{05} = FV$	SF 00 pour Début Période CF 00 pour Fin Période
*I (intérêt)	$R_{01} = N$ $R_{03} = PV$ $R_{04} = PMT$ $R_{05} = FV$	X = I $R_{01} = N$ $R_{02} = I$ $R_{03} = PV$ $R_{04} = PMT$ $R_{05} = FV$	SF 00 pour Début Période CF 00 pour Fin Période
PV (valeur actuelle)	$R_{01} = N$ $R_{02} = I$ $R_{04} = PMT$ $R_{05} = FV$	X = PV $R_{01} = N$ $R_{02} = I$ $R_{03} = PV$ $R_{04} = PMT$ $R_{05} = FV$	SF 00 pour Début Période CF 00 pour Fin Période
PMT (montant paiement)	$R_{01} = N$ $R_{02} = I$ $R_{03} = PV$ $R_{05} = FV$	X = PMT $R_{01} = N$ $R_{02} = I$ $R_{03} = PV$ $R_{04} = PMT$ $R_{05} = FV$	SF 00 pour Début Période CF 00 pour Fin Période
FV (valeur future)	$R_{01} = N$ $R_{02} = I$ $R_{03} = PV$ $R_{04} = PMT$	X = FV $R_{01} = N$ $R_{02} = I$ $R_{03} = PV$ $R_{04} = PMT$ $R_{05} = FV$	SF 00 pour Début Période CF 00 pour Fin Période

Commentaires. Pour utiliser ces sous-programmes, chargez les quatre registres appropriés, armez (ou désarmez) l'indicateur binaire 00 pour choisir le mode Début de Période (ou Fin de Période), puis exécutez le sous-programme voulu. Il fournit la valeur recherchée dans le registre X et la stocke dans le registre correspondant.

ANNEXE A : LES TAUX D'INTERET

Dans toutes les transactions financières que vous effectuez (placements, épargne liquide, emprunts), les taux d'intérêt annoncés sont toujours annuels. Les modalités de calcul des intérêts sont très variées et en général basées sur des périodes de composition non annuelles. Ces périodes sont semestrielles, trimestrielles, mensuelles ou par quinzaine (ce qui est différent d'une période semi-mensuelle).

Avant d'aborder les méthodes de calcul, donnons quelques définitions et explications sur les formules des mathématiques financières.

Notion d'équivalence de plusieurs sommes

Pour comparer entre elles plusieurs sommes d'argent versées à des périodes différentes et placées (ou empruntées) avec des modalités de calcul différentes, il faut estimer leur valeur à une même époque.

$$FV = PV \left(1 + \frac{T}{100}\right)^n$$

Cette formule signifie que **FV** francs de l'époque n sont équivalents à **PV** francs d'aujourd'hui placés à $t\%$ pendant n périodes. Ceci est la formule fondamentale des intérêts composés.

Dans tout calcul financier, il est nécessaire si l'on veut connaître le rapport (ou le coût) réel de l'opération de passer par cette formule.

Exemple : 12 % l'an ne sont pas équivalents à 1 % par mois en composition mensuelle. 100 F placés un an à 12 % l'an deviennent 112 F.

Mais que deviennent 100 F placés 12 mois à 1 % par mois ?

Frappez



100  

12 

1



Affichage

N, I, PV,PMT,FV

N, I, PV,PMT,FV

-100,00

12,00

1,00

112,68

Lance TVM.

Efface toute ancienne donnée financière.

Valeur future.

Les deux valeurs trouvées sont différentes, donc 12 % l'an et 1 % par mois ne sont pas équivalents. Pour avoir un taux équivalent, il faut que les 100 F placés en composition annuelle et en composition mensuelle donnent la même valeur finale.

Quels sont les différents taux que l'on retrouve le plus souvent nommés ?

Taux périodique

- C'est le taux correspondant à la périodicité des opérations (semestre, trimestre, mois...).
- Il peut être, selon les organismes et selon la nature de l'opération, nominal ou actuel (équivalent).
- C'est le taux qui est utilisé dans votre module (i).

Taux nominal (ou proportionnel)

- Il peut être périodique : c'est le taux obtenu en divisant le taux annuel par la périodicité.
- Il peut être annuel : c'est le taux obtenu en multipliant le taux périodique par la périodicité.
- Ce taux est inexact au sens des intérêts composés.
- En prenant en compte dans les calculs les frais et diverses commissions, c'est le taux effectif global figurant sur les propositions de prêts d'organismes financiers.

Taux équivalent (ou actuel)

- C'est le taux obtenu par l'application de la formule fondamentale des intérêts composés.
- C'est le taux qui, seul, mesure le vrai coût de l'opération financière.

Mis à part certains placements assortis d'une obligation légale de faire figurer le taux actuel annuel de rendement, dans les autres cas la nature du taux annoncé (nominal ou actuel) n'est jamais précisée. Il faut donc pouvoir passer rapidement de l'un à l'autre.

Formules de conversion des taux entre eux

1. Du taux actuel annuel (t_{aa}) → au taux actuel périodique (i_{ap}).

$$i_{ap} = (1 + t_{aa})^{\frac{1}{n}} - 1$$

2. Du taux actuel annuel (t_{aa}) → au taux nominal périodique (i_{Np}).

$$i_{Np} = \frac{t_{aa}}{n}$$

3. Du taux actuel périodique (i_p) → au taux nominal annuel (t_{NA}).

$$t_{NA} = i_p \times n$$

4. Du taux actuel périodique (i_p) → au taux actuel annuel (t_{aa}).

$$t_{aa} = (1 + i_p)^n - 1$$

n = périodicité

i = taux périodique sous la forme d'une fraction ($i = I/100$)

t = taux annuel sous la forme d'une fraction ($t = T/100$)

Comment effectuer ces conversions sur votre HP-41

1. Quel est le taux actuel mensuel équivalent au taux actuel annuel de 12 % ?

Pour cela, utilisons les touches du programme TVM.

Frappez

12 [n]
100 [CHS] [PV]
112 [FV]
[i]

Affichage

12,00
- 100,00
112,00
0,95

Nombre de périodes.

Taux actuel
mensuel.

Quel est le taux trimestriel équivalent au taux annuel de 17 % ?

Frappez

4 [n]
100 [CHS] [PV]
117 [FV]
[i]
[f] 4
[f] 2

Affichage

4,00
- 100,00
117,00
4,00
4,0031
4,00

Taux trimestriel
actuel
Retour en notation
standard.

2. Quel est le taux semestriel proportionnel au taux annuel de 18 % ?

Frappez18 **ENTER**2 **÷****Affichage**

18,00

9,00

Taux semestriel proportionnel.

Quel est le taux trimestriel proportionnel aux taux annuel de 17 % ?

Frappez17 **ENTER**4 **÷****Affichage**

17,00

4,25

Taux trimestriel proportionnel.

(écart non négligeable avec le taux trimestriel équivalent).

3. Quel est le taux annuel proportionnel aux taux trimestriel de 4 % ?

Frappez4 **ENTER**4 **×****Affichage**

4,00

16,00

Taux annuel proportionnel.

Quel est le taux annuel proportionnel aux taux mensuel de 0,95 % ?

Frappez0,95 **ENTER**12 **×****Affichage**

0,95

11,40

Taux annuel proportionnel.

4. Quel est le taux annuel équivalent au taux mensuel de 1 % ?

Frappez12 **n**1 **i**100 **CHS** **PV****FV**100 **–****Affichage**

12,00

1,00

– 100,00

112,68

12,68

Périodicité.

Taux mensuel.

Valeur actuelle.

Valeur future.

Taux actuel annuel.

(Résultat bien loin des 12 % annuel nominaux !)

Quel est le taux annuel équivalent au taux trimestriel de 4,25 % ?

Frappez4 **n**4,25 **i**100 **CHS** **PV****FV**100 **–****Affichage**

4,00

4,25

– 100,00

118,11

18,11

Périodicité.

Taux trimestriel.

Valeur actuelle.

Valeur future.

Taux actuel annuel.

Compte tenu des différences obtenues selon les modalités de calcul, il apparaît clairement qu'il n'est pas indifférent de connaître ces dernières lors du choix d'un placement par rapport à un autre.

Exemple : Un organisme vous propose le choix entre deux placements, l'un à 15 % l'an, l'autre à 3,6 % par trimestre à composition trimestrielle. Pour lequel allez-vous opter ?

A quoi correspond 3,6 % par trimestre en taux actuariel annuel ?

Frappez

 **E**

4 **n**

3,6 **i**

100 **CHS** **PV**

FV

100 **-**

Affichage

N, I, PV,PMT,FV

4,00

3,60

- 100,00

115,20

15,20

Efface toute ancienne donnée financière.

Périodicité.

Taux trimestriel.

Taux actuariel annuel.

La deuxième solution offre un meilleur rendement.

INDEX DES PROGRAMMES

CFIT	133
Fonctions sur les complexes	93
DIFEQ	87
INTEG	79
Fonctions logiques	129, 130
Fonctions matricielles	30, 58
MATRX	19
Changements de base numérique	127, 128
PLY	71
SOLVE	61
TR	117
TVM	143
VC	101



Portable Computer Division
1000 N.E. Circle Blvd., Corvallis, OR 97330, U.S.A.

Siège pour l'Europe
150 Route du Nant-d'avril
B.P. CH-1217 Meyrin 2
Genève, Suisse

HP-United Kingdom
(Pinewood)
GB-Nine Mile Ride, Wokingham
Berkshire RG11 3LL

N° de référence
00041-90561
Français - French

Printed in Singapore 8/85
Imprimé à Singapour

Scan Copyright ©
The Museum of HP Calculators
www.hpmuseum.org

Original content used with permission.

Thank you for supporting the Museum of HP
Calculators by purchasing this Scan!

Please do not make copies of this scan or
make it available on file sharing services.