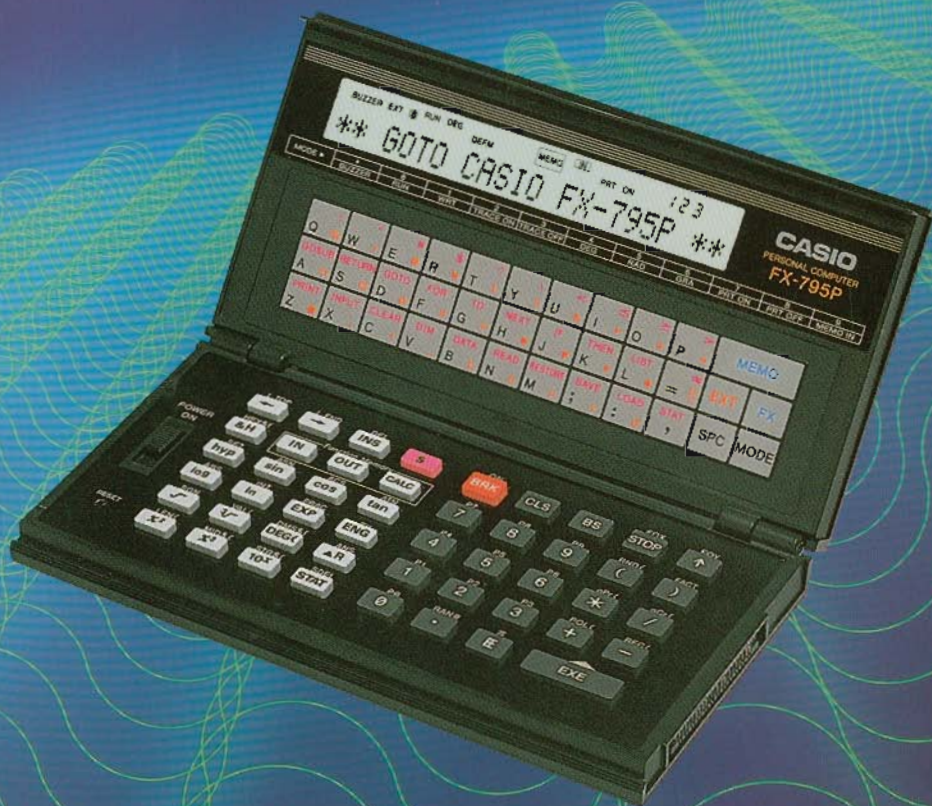


ORDINATEUR PERSONNEL

FX-795P

MANUEL DE L'UTILISATEUR



CASIO®

ORDINATEUR PERSONNEL
FX-795P
MANUEL DE L'UTILISATEUR

Ce manuel est sujet à modifications sans avis préalable.
Toute reproduction en tout ou partie de ce manuel est formellement interdite. Toute utilisation de ce manuel à des fins autres que personnelles, sans l'autorisation de CASIO, est interdite aux termes de la loi sur les publications.
CASIO Computer Co., Ltd. ne peut être tenue pour responsable des dommages ou pertes pouvant résulter de l'utilisation de ce manuel.

CASIO®

Avant-propos

Ce manuel contient une explication simple des diverses fonctions de cet ordinateur. Cet appareil se caractérise d'abord par l'abondance de ses fonctions scientifiques comprenant même le calcul des opérations statistiques. Les nombreuses fonctions numériques et statistiques de base de cet ordinateur vous permettront de réaliser sans problème des calculs scientifiques et technologiques comme des calculs de statistiques à l'aide d'opérations simples. La programmation en langage BASIC permet en plus de procéder à des calculs plus complexes.

La fonction "MEMORY" (mémoire) est la deuxième caractéristique de cet ordinateur. Grâce à cette fonction, il est possible de mettre des expressions numériques en mémoire, expressions qui, après entrée des valeurs numériques nécessaires, permettront d'obtenir les résultats des calculs. Cette fonction permet de procéder facilement à des calculs numériques sans l'aide d'un programme spécial en BASIC.

La troisième caractéristique est la fonction "DATA BANK" (banque de données) qui permet d'utiliser l'ordinateur comme un "bloc-notes électronique". Contrairement au bloc-notes dont la mise en ordre ou le repérage des données individuelles demande un travail considérable, cet ordinateur effectue le même travail de manière fort simple.

Enfin, il convient encore d'ajouter comme quatrième caractéristique de cet ordinateur, ses utilitaires incorporés: cinq programmes de calculs mathématiques qui pour trouver la solution à un problème donné vous demandent simplement d'entrer les valeurs ou expressions numériques nécessaires pour le calcul.

Un tel ordinateur s'adresse donc à tous, aussi bien à ceux qui vont découvrir le BASIC qu'à ceux qui le connaissent déjà et veulent en utiliser toutes les possibilités.

Notre souhait est que ce manuel vous permette d'utiliser cet ordinateur de manière efficace pendant de nombreuses années.

Avant utilisation

Cet ordinateur, fruit de la technologie électronique CASIO de haut niveau, a été soumis à des opérations d'essai rigoureuses et à un contrôle sévère de qualité. Assurez une longue durée de vie à votre ordinateur en observant les précautions suivantes.

Précautions d'utilisation

- Cet ordinateur étant composé de pièces électronique de précision, ne pas le démonter. Ne pas, non plus, le soumettre à des chocs, éviter de le jeter ou de laisser tomber, et ne pas l'exposer à des changements brusques de température. De plus, ne pas le ranger en un lieu où température, humidité et empoussièrement sont élevés. Si l'ordinateur est utilisé à basse température l'affichage est parfois lent, voire inexistant. Toutefois, dès que la température revient à la normale, l'ordinateur revient à un fonctionnement normal.
- Prendre soin à ne pas soumettre l'ordinateur à des forces de flexion. Ne pas le transporter, par exemple, dans la poche d'un vêtement.
- Ne pas raccorder d'unités périphériques autre que le FA-5, le FP-12S, le FP-40 ou tout autre modèle expressément spécifié au connecteur prévu à cet effet.
 - * Pour le branchement au FP-12S utiliser le câble pour imprimante SB-2 et pour le raccordement au FP-40, le module d'interface SB-42.
- Si l'affichage faiblit pendant que la sonnerie retentit, le bon fonctionnement de l'ordinateur n'est pas pour autant en cause. Par contre, si l'affichage est vraiment faible, changer les piles aussi rapidement que possible.
- Changer les piles tous les deux ans même si l'ordinateur n'a pas été utilisé pendant ce laps de temps. Ne pas laisser des piles épuisées dans l'appareil sous peine de problème possible en cas de fuite des piles.
- Toujours laisser en place le capuchon de la partie connecteur. Ne l'enlever que pour raccorder des périphériques au connecteur.
- Si une forte charge d'électricité statique est appliquée à l'ordinateur, le contenu de la mémoire risque éventuellement de changer ou bien le clavier ne répond plus. Pour décharger l'électricité statique accumulée dans le corps, toucher une pièce de métal comme par exemple un bouton de porte. Si un tel problème se produisait, enlever les piles puis les remettre en place.
- Avant de raccorder un périphérique, toujours mettre l'ordinateur hors tension.

- Pour le nettoyage de l'ordinateur ne pas utiliser de liquides volatiles du type benzène ou diluant. Essuyer l'ordinateur avec un chiffon doux et sec ou imbibé d'une solution composée d'un détergent neutre.
- Ne pas couper l'alimentation électrique pendant l'exécution d'un programme ou pendant une opération.
- L'ordinateur étant composé de pièces électroniques de précision, éviter de le faire tomber pendant qu'un programme est en cours d'exécution sous peine d'interruption du programme ou de modification du contenu de la mémoire.
- En cas de mal fonctionnement de l'ordinateur, consulter le magasin vendeur de l'appareil ou le revendeur le plus proche.
- Avant de vous adresser à un service après-vente, relisez ce manuel, vérifiez l'alimentation électrique, les erreurs de logique possibles dans le programme etc.

Sommaire

CHAPITRE 1 Guide général

1-1	Nom des composants	2
1-2	Fonctions des composants	3
1-3	Alimentation électrique	12

CHAPITRE 2 Opérations manuelles

2-1	Première utilisation de l'ordinateur	16
2-2	Quatre opérations arithmétiques pour commencer	21
2-3	Notes à propos des calculs	23
2-4	Calculs de fonctions	24
2-5	Calculs de statistiques	35

CHAPITRE 3 Utilisation de la fonction mémoire

3-1	Calculs avec une même équation	44
3-2	Utilisation pour la préparation de tableaux	48

CHAPITRE 4 Utilitaires incorporés

4-1	Menu des utilitaires	52
4-2	Calcul matriciel	53
4-3	Calcul de nombres complexes	67
4-4	Résolution numérique d'équations	74
4-5	Intégration numérique	78
4-6	Opérations binaires, décimales et hexadécimales	81

CHAPITRE 5 Programmation en langage BASIC

5-1	Ecrire les programmes	90
5-2	Exécuter un programme	93
5-3	Variabiles	95
5-4	Méthode de calcul de la longueur d'un programme	103
5-5	Techniques utiles	104
5-6	Messages erreur et mise au point	107
5-7	Périphériques utiles	109
5-8	Utilisation d'un programme PB-100	118

CHAPITRE 6 Bibliothèque de programmes

6-1	Redisposition des données (Tri)	124
6-2	Jeu des courses de chevaux	127

CHAPITRE 7 Détail des commandes

7-1	Commandes manuelles	136
	NEW [ALL]	136
	RUN	137
	LIST	138
	PASS	139
	SAVE [ALL]	141
	LOAD [ALL]	142
	VERIFY	143
	CLEAR	144
7-2	Commandes de programme	145
	END	145
	STOP	145
	[LET]	146
	REM	147
	INPUT	148
	KEY\$	150
	PRINT	151
	CSR	152
	GOTO	153
	ON ~ GOTO	154
	IF ~ THEN	156
	FOR ~ TO ~ [STEP] NEXT	158
	GOSUB	159
	RETURN	161
	ON ~ GOSUB	161
	DATA	162
	READ	163
	RESTORE	165
	PUT	166
	GET	167

	BEEP	170
	DEFM	171
	DIM	173
	ERASE	175
	MODE	176
	STAT CLEAR	177
	STAT	177
	STAT LIST	178
	SET	179
7-3	Fonctions caractère	180
	LEN	180
	MID\$	181
	VAL	182
	STR\$	183
7-4	Fonctions numériques	184
	SIN, COS, TAN	184
	ASN, ACS, ATN	185
	HYP SIN, HYP COS, HYP TAN	186
	HYP ASN, HYP ACS, HYP ATN	187
	LOG, LN	188
	EXP	188
	SQR	189
	CUR	189
	ABS	190
	SGN	190
	INT	191
	FRAC	192
	RND	192
	REC	193
	POL	194
	FACT	195
	NPR	196
	NCR	197
7-5	Fonctions statistiques	198
	EOX	198
	EOY	199

Sommaire

7-6	Autres	200
	RAN#	200
	DEG	201
	DMS\$	202
	HEX\$	203
	&H	204
7-7	Commandes de DATA BANK (Banque de données)	206
	NEW#	206
	LIST#	206
	SAVE#	208
	LOAD#	209
	READ#	210
	RESTORE#	212
	WRITE#	215

CHAPITRE 8 Fonction DATA BANK utile

8-1	Pour spécifier le mode MEMO IN	218
8-2	Introduction des données	219
8-3	Affichage du contenu des données	222
8-4	Correction des données	223
8-5	Repérage (recherche) des données	225
8-6	Effacement des données	231
8-7	Addition et insertion de données	232
8-8	Recherche à l'aide d'un programme	235
8-9	Application aux calculs de tabulation	237
8-10	Combinaison avec la fonction mémoire	241

Appendice

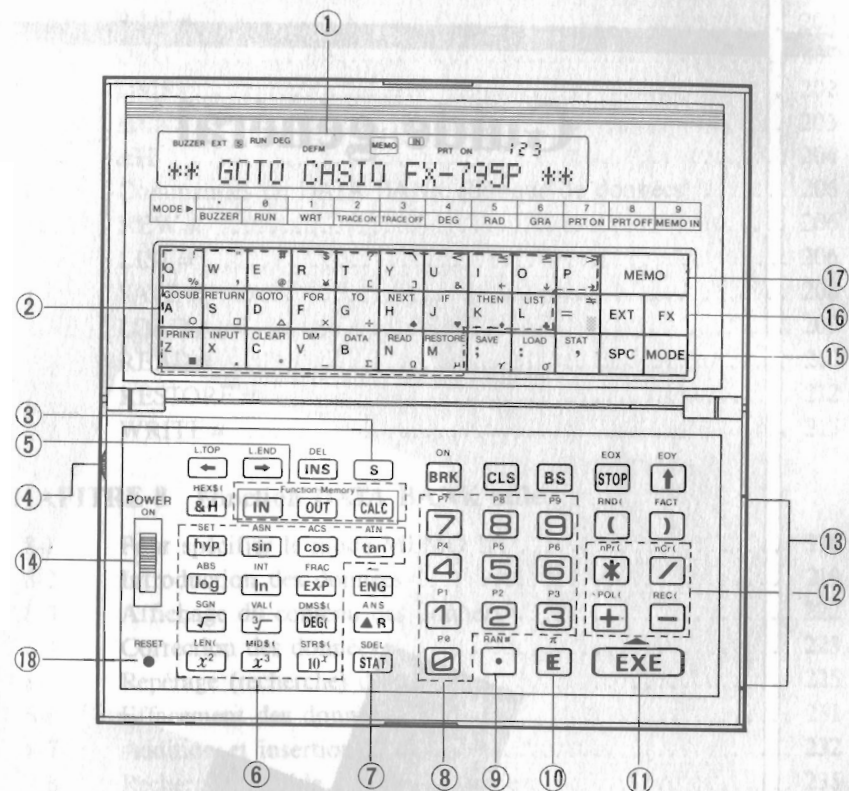
Table du code caractère	244
Calculs de fonctions	245
Messages d'erreur	248
Spécifications	251
Index	254

1

Guide général

Que vous soyez déjà familiarisé ou non avec les ordinateurs, nous vous conseillons de lire attentivement ce manuel dans un premier temps. Plus vite vous connaîtrez la configuration de cet ordinateur, plus vite vous serez à même d'en utiliser toutes les ressources.

1-1 Nom des composants



- | | |
|--|-----------------------------------|
| ① Visualisation | ⑩ Touche exposant |
| ② Touches alphabétiques | ⑪ Touche exécution |
| ③ Touche d'inversion | ⑫ Touches de commande des calculs |
| ④ Réglage visualisation | ⑬ Connecteur pour périphériques |
| ⑤ Touches fonction mémoire | ⑭ Interrupteur d'alimentation |
| ⑥ Touches fonctions | ⑮ Touche mode |
| ⑦ Touche d'entrée des données statistiques | ⑯ Touche FX |
| ⑧ Touches numériques | ⑰ Touche mémoire |
| ⑨ Touche virgule | ⑱ Bouton de réinitialisation |

1-2 Fonctions des composantes

- **Interrupteur d'alimentation**

Glisser l'interrupteur vers le haut pour mettre sous tension et vers le bas pour mettre hors tension.

- Touche d'inversion (Touche rouge **S**)

Appuyer sur cette touche pour une commande touche-unique (pour entrer plusieurs caractères à l'aide d'une seule touche) indiquée en rouge sur le panneau, pour l'affichage des symboles ou pour entrer une fonction. Quand cette touche est enclenchée une fois, le mode SHIFT-IN est sélectionné et le symbole “**S**” est affiché. Quand cette touche est de nouveau enclenchée ou si une autre touche est enclenchée, le mode SHIFT-IN est annulé et le symbole “**S**” disparaît de l'écran. (Dans ce manuel, on utilisera le symbole **SHIFT** pour distinguer cette touche de la touche alphabétique **S**.)

• Touches numériques



Touche virgule



Touches de commande de calcul



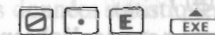
Touche exposant



Touches parentales



Interrupteur d'alimentation



Les dix touches $\boxed{\square}$ à $\boxed{0}$ permettent d'entrer des nombres. La touche $\boxed{.}$ est la virgule. Les touches de commande de calcul $\boxed{+}$, $\boxed{-}$, $\boxed{\times}$ et $\boxed{\div}$ sont spéciales. Ces touches servent à effectuer les quatre opérations arithmétiques (addition, soustraction, multiplication et division), mais les signes $\times$ et $\div$ des calculatrices électroniques ordinaires sont remplacés ici respectivement par $*$ et $/$. Ceci en raison de l'utilisation de ces signes en langage BASIC. Sur les calculatrices ordinaires, la touche $\boxed{=}$ permet d'obtenir un résultat. Sur cet ordinateur, on utilise la touche $\boxed{EXE}$ à cet effet. Par exemple, alors que sur une calculatrice ordinaire il faut appuyer sur les touches $12 \boxed{\times} 4 \boxed{\div} 3 \boxed{+} 7 \boxed{=} 5$ avec cet ordinateur on appuie sur les touches $12 \boxed{*} 4 \boxed{/} 3 \boxed{+} 7 \boxed{=}$ 5 $\boxed{EXE}$.

Quand elle est enclenchée directement, la touche $\boxed{E}$ sert de touche exposant. Appuyer sur cette touche avant d'entrer l'exposant (une certaine puissance 10). Pour entrer $1,23 \times 10^4$ par exemple, appuyer sur $1 \boxed{.} 2 \boxed{3} \boxed{E} 4$. Si l'exposant est négatif, appuyer sur la touche $\boxed{-}$ après la touche $\boxed{E}$.

Pour entrer $7,41 \times 10^{-9}$ par exemple, appuyer sur $\boxed{7} \boxed{.} \boxed{4} \boxed{1} \boxed{E} \boxed{-} \boxed{9}$.
 Si cette touche est enclenchée après la touche $\boxed{\pi}$ (π (quotient de la circonférence d'un cercle par son diamètre) sera affiché).
 Les touches $\boxed{(}$ et $\boxed{)}$ permettent d'entrer les parenthèses et la touche $\boxed{\wedge}$ permet les calculs de puissance (pour x^y , appuyer sur $x \boxed{\wedge} y$).

• Touches alphabétiques et touche espace

$\boxed{Q} \boxed{W} \boxed{E} \boxed{R} \boxed{T} \boxed{Y} \boxed{U} \boxed{I} \boxed{O} \boxed{P}$
 $\boxed{A} \boxed{S} \boxed{D} \boxed{F} \boxed{G} \boxed{H} \boxed{J} \boxed{K} \boxed{L} \boxed{=}$
 $\boxed{Z} \boxed{X} \boxed{C} \boxed{V} \boxed{B} \boxed{N} \boxed{M} \boxed{:} \boxed{.} \boxed{,} \boxed{SPC}$

Ces touches n'existent pas sur les calculatrices ordinaires. Les 26 lettres de l'alphabet, les touches symboles ($\boxed{=}$, $\boxed{:}$, $\boxed{.}$, $\boxed{,}$) et la touche espace ($\boxed{SPC}$) sont disposées comme sur une machine à écrire. Ces touches permettent de donner des ordres ou d'écrire des programmes. De plus, les touches alphabétiques de A à Z servent individuellement de "mémoire de variable" (où les valeurs numériques sont stockées).

• Touche égal ($\boxed{=}$)

Cette touche ne permet pas d'obtenir le résultat d'un calcul mais est utilisée pour un jugement conditionnel dans des instructions d'affectation (voir page 146) ou des instructions IF (voir page 156). Si cette touche est enclenchée après la touche $\boxed{\neq}$, le signe $\neq$ (différent de) est affiché.

• Touches fonction

Une des caractéristiques de cet ordinateur est qu'il permet d'entrer des fonctions à l'aide d'une seule touche comme sur les calculatrices scientifiques. Ceci est particulièrement pratique pour les calculs de fonctions, $\boxed{x^2}$ et $\boxed{x^3}$ représentant respectivement les touches "carré" et "cube". La touche $\boxed{10^x}$ permet le calcul des fonctions exponentielles (10^x) en base 10.

Enclenchées après la touche $\boxed{\text{SHIFT}}$, ces touches permettent le calcul de la fonction correspondante.

SET hyp	ASN sin	ACS cos	ATN tan
ABS log	INT log	FRAC EXP	ENG
SGN $\sqrt{}$	VAL $\sqrt[3]{}$	DMS DEG	
LEN x^2	MID x^3	STR 10^x	

• Touche hexadécimale ($\boxed{\text{HEX}}$)

Appuyer sur $\boxed{\text{HEX}}$ nombre hexadécimal $\boxed{\text{EXE}}$ pour convertir un nombre hexadécimal en un nombre décimal. Appuyer sur $\boxed{\text{SHIFT}} \boxed{\text{HEX}}$ nombre décimal $\boxed{\text{EXE}}$ pour convertir un nombre décimal en un nombre hexadécimal à 4 chiffres.

• Touche ingénierie ($\boxed{\text{ENG}}$)

Cette touche permet de convertir un résultat de calcul ou une valeur numérique affichée à l'aide la suite d'une instruction PRINT en affichage d'exposant. Si cette touche est enclenchée plusieurs fois de suite, l'exposant affiché diminue de 3 à chaque fois. L'exposant peut être augmenté de 3 en appuyant sur $\boxed{\text{SHIFT}} \boxed{\text{ENG}}$. (Multiples de 3 : $10^3 = k$ (kilo), $10^6 = M$ (méga), $10^9 = G$ (giga), $10^{-3} = m$ (milli), $10^{-6} = \mu$ (micro), $10^{-9} = n$ (nano), $10^{-12} = p$ (pico))

• Touche rappel/réponse ($\boxed{\text{ANS}}$)

Cette touche permet d'afficher les données entrées pour le calcul effectué immédiatement avant en utilisant $\boxed{\text{EXE}}$. Si on appuie sur la touche $\boxed{\text{SHIFT}}$ avant d'appuyer sur cette touche, c'est le résultat du dernier calcul qui est affiché.

• Touche entrée de données statistiques ($\boxed{\text{SDEL}}$)

Quand cette touche est enclenchée après les données statistiques, les données sont entrées dans une zone de mémoire exclusive. Si cette touche est enclenchée après la touche $\boxed{\text{SHIFT}}$, une fonction SDEL est effectuée pour annuler les données statistiques se trouvant dans la zone de mémoire. (Pour le détail voir page 35.)

• Touche fonction mémoire ($\boxed{\text{IN}} \boxed{\text{OUT}} \boxed{\text{CALC}}$)

Ces touches pour la fonction mémoire sont décrites en détail en Chapitre 3.

• Touches de déplacement du curseur ($\boxed{\text{TOP}} \boxed{\text{END}}$)

On utilise ces touches pour corriger des caractères affichés. Le curseur ("—" clignotant sur l'écran) peut être déplacé vers la droite et la gauche à l'aide de ces touches. Si la touche est maintenue enfoncée, le curseur se déplace en continu sur l'ensemble des caractères existants. Les touches $\boxed{\text{SHIFT}} \boxed{\text{TOP}}$ permettent d'amener le curseur à l'extrême gauche de la visualisation (au début de la ligne) — Fonction LINE TOP (première ligne). Les touches $\boxed{\text{SHIFT}} \boxed{\text{END}}$ permettent d'amener le curseur à droite du dernier caractère entré (fin de la ligne) — Fonction LINE END (fin de ligne).

• Touche insérer/effacer ()

Il s'agit d'une autre touche particulièrement pratique pour corriger des caractères affichés. Quand cette touche est enclenchée directement, les caractères suivant celui sous lequel se trouve le curseur sont déplacés vers la droite pour créer un espace dans lequel il sera possible d'insérer un nouveau caractère. Le curseur lui-même ne se déplace pas.

Si les touches   sont enclenchées, le caractère se trouvant sur le curseur est effacé et les caractères se trouvant à sa droite sont déplacés vers la gauche. La position du curseur reste inchangée.

Insertion et effacement en continu sont possibles en gardant respectivement appuyées les touches  et  .

• Touche d'interruption ()

Il s'agit d'une touche puissante permettant de suspendre diverses opérations (opération manuelle, exécution d'un programme, entrée/sortie avec une cassette, sortie vers une imprimante, sortie de la liste des programmes) et d'annuler des erreurs. Si cette touche est enclenchée alors que la visualisation est hors fonction sous l'action de ARRET AUTO ALIMENTATION (voir page 13), l'alimentation est de nouveau mise en circuit.

• Touche effacement visualisation ()

Cette touche permet d'effacer ce qui se trouve sur l'écran et d'amener le curseur à l'extrême gauche de l'écran d'affichage.

• Touche de retour en arrière ()

Cette touche permet d'effacer le caractère se trouvant immédiatement à gauche du curseur, les caractères se trouvant immédiatement à droite du caractère effacé étant alors déplacés vers la gauche. Contrairement à la touche , le curseur se déplace vers la gauche.

Il est possible d'effacer en continu les caractères placés à gauche du curseur en gardant cette touche appuyée.

• Touche arrêt ()

Si cette touche est enclenchée pendant l'exécution d'un programme, cette exécution est provisoirement suspendue. Si elle est enclenchée pendant le défilement de la visualisation, celui-ci est provisoirement suspendu. La touche  permet de reprendre l'exécution du programme.

• Touche mémoire ()

Appuyer sur cette touche pour utiliser la fonction DATA BANK. Pour le détail voir Chapitre 8.

• Touche FX ()

Appuyer sur cette touche pour utiliser les programmes utilitaires incorporés. Pour de plus amples détails, prière de se reporter au Chapitre 4.

• Touche mode ()

On utilise cette touche conjointement avec ,  ~  pour spécifier un mode ordinateur ou une unité d'angle.

-   Permet de mettre en et hors fonction la sonnerie entrée de touche. Quand la sonnerie est en fonction, le symbole "BUZZER" est affiché.
-   Le symbole "RUN" est affiché pour les calculs manuels et par programme. (Mode RUN)
-   Le symbole "WRT" est affiché pour l'écriture, le contrôle et l'édition de programmes. (Mode WRT)
-   Le symbole "TRACE ON" est affiché pour l'exécution de l'analyse. (Pour le détail voir page 108)
-   Quand le symbole "TRACE ON" est affiché, le mode analyse d'exécution est annulé et "TRACE ON" disparaît.
-   Le symbole "DEG" est affiché si l'unité d'angle spécifiée est le "degré".
-   Le symbole "RAD" est affiché si l'unité d'angle spécifiée est le "radian".

MOD [6] Le symbole "GRA" est affiché si l'unité d'angle spécifiée est le "grade".

MOD [7] Le symbole "PRT ON" est affiché et une sortie sur imprimante est possible si l'ordinateur est relié à une imprimante.

MOD [8] Quand le symbole "PRT ON" est affiché, le mode sortie sur imprimante est annulé et "PRT ON" disparaît.

MOD [9] Le symbole "MEMO IN" est affiché pour indiquer que le mode MEMO IN a été spécifié. Ce mode permet d'entrer les données mémo dans la DATA BANK. (Pour le détail, voir Chapitre 8)
Pour annuler ce mode, appuyer sur MOD [0].

• Touche extension (EXT)

Le mode direct est spécifié immédiatement après mise sous tension. Si une touche est enclenchée dans ce mode, le caractère correspondant à cette touche est entré. Toutefois, si la touche EXT est enclenchée, le mode extension (le symbole "EXT" est affiché) est spécifié et il est alors possible d'entrer des minuscules ou des symboles spéciaux. Le mode extension ne concerne que la moitié supérieure du clavier.

Les fonctions des touches individuelles dans chaque mode sont indiquées ci-après:

a) Mode direct (appuyer simplement sur la touche souhaitée.)

Q	W	E	R	T	Y	U	I	O	P	MEMO	
A	S	D	F	G	H	J	K	L	=	EXT	F X
Z	X	C	V	B	N	M	;	:	,	SPC	MODE

←	→	INS	S	BRK	CLS	BS	STOP	↑
&H	IN	OUT	CALC	7	8	9	(	)
hyp	sin	cos	tan	4	5	6	*	/
log	ln	EXP	ENG	1	2	3	+	-
√	√	DEG	ΔR	0	.	E	EXE	

b) Mode inversion (appuyer sur la touche souhaitée après [SHIFT])

!	"	#	\$	?	\	<	≤	≥	>	MEMO	
GOSUB	RETURN	GOTO	FOR	TO	NEXT	IF	THEN	LIST	≠	EXT	F X
PRINT	INPUT	CLEAR	DM	DATA	READ	RESTORE	SAVE	LOAD	STAT	SPC	MODE

LTOP	LEND	DEL	S	BRK	CLS	BS	EOX	EOY
HEX\$(	IN	OUT	CALC	P7	P8	P9	RND(	FACT
SET	ASN	ACS	ATN	P4	P5	P6	nPr(	nCr(
ABS	INT	FRAC		P1	P2	P3	POL(	REC(
SGN	VAL(	DM\$(	ANS	P0	RAN#	π		
LEN(	MID\$(	STR\$(	SDEL					

c) Mode direct en mode extension (appuyer sur la touche souhaitée pendant que "EXT" est en fonction.)

q	w	e	r	t	y	u	i	o	p	MEMO	
a	s	d	f	g	h	j	k	l	=	EXT	F X
z	x	c	v	b	n	m	;	:	,	SPC	MODE

d) Mode inversion en mode extension (appuyer sur la touche souhaitée après [SHIFT] pendant que "EXT" est en fonction.)

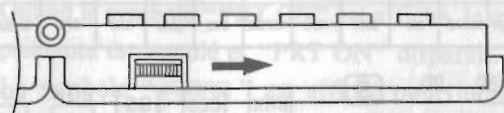
%	,	@	¥	£	¤	&	←	↓	→	MEMO	
○	□	△	×	÷	◆	♥	♦	♣	☼	EXT	F X
■	.	°	-	Σ	Ω	μ	Υ	σ	,	SPC	MODE

Noter que chaque touche a plusieurs fonctions.

Si la touche EXT est enclenchée une nouvelle fois alors que le symbole "EXT" est affiché (mode extension.), "EXT" disparaît et le mode extension est annulé.

• Réglage visualisation

Si la visualisation est sombre ou affaiblie, selon l'état des piles et l'angle de vision de l'écran, régler à l'aide de la commande se trouvant sur la gauche de l'ordinateur.

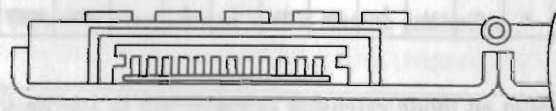


Pour augmenter la luminosité de l'écran, tourner la commande dans le sens indiqué par la flèche. Pour réduire la luminosité, tourner dans le sens contraire. Si la luminosité de l'écran reste faible, même quand la commande est placée en position de luminosité maximale, les piles sont probablement usées.

Si tel est le cas, changer les piles le plus rapidement possible. (Pour le changement des piles voir page 12.)

• Connecteur pour périphériques

Ce connecteur permet de raccorder les périphériques disponibles en option, à savoir l'imprimante FP-12S ou FP-40 ou encore l'interface cassette FA-5 qui permet d'utiliser un magnétophone.

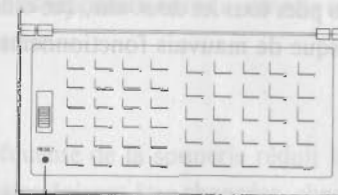


Pour le raccordement de l'imprimante FP-12S, utiliser le câble SB-2, fourni à part, qui est prévu spécialement à cet effet et pour le raccordement de l'imprimante FP-40, le module d'interface SB-42 qui est également fourni séparément.

Ne jamais insérer un câble ou une interface autre que SB-2, SB-42 ou FA-5 dans ce connecteur. Quand aucun périphérique n'est utilisé, recouvrir le connecteur de son capuchon.

• Commande RESET (réinitialisation)

Cette commande est placée sous l'interrupteur d'alimentation. Quand on appuie sur cette commande avec un objet pointu, l'ordinateur étant sous tension, celui-ci est réinitialisé dans l'état où ni spécification, ni entrée ne sont effectuées. On peut recourir à cette opération quand l'ordinateur est en état de verrouillage du fait d'une forte électricité statique. Attention car tous les programmes et toutes les données seront perdus.



Bouton de remise à zéro intégrale

Le système d'alimentation électrique de l'ordinateur comprend une alimentation électrique principale (deux piles au lithium CR2032) et une alimentation électrique auxiliaire de soutien pour la mémoire (une pile au lithium CR1220). Si la luminosité de l'écran reste faible même après réglage (voir page 10), changer les piles aussi vite que possible car celles-ci sont pratiquement vides.

Note:

Ne pas oublier de changer les piles tous les deux ans, que celles-ci aient été utilisées ou non, pour éviter tout risque de mauvais fonctionnement dû à des fuites de ces piles.

Changement des piles

- 1) Couper l'alimentation électrique et déposer le panneau arrière après avoir enlevé les trois vis.

- 2) Enlever les piles.

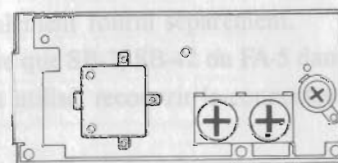
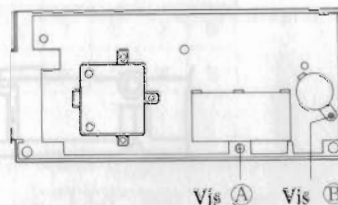
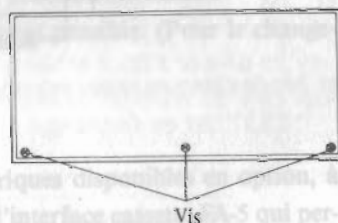
Piles principales:

Enlever le couvercle du logement des piles après avoir déposé la vis A.

Pile auxiliaire:

Enlever le couvercle du logement de la pile après avoir déposé la vis B.

- 3) Enlever les piles usées. (Celles-ci sortent d'elles-mêmes en retournant l'ensemble et en le tapant légèrement de la main.)
- 4) Bien essuyer les piles neuves à l'aide d'un chiffon sec avant de les mettre en place, côté $\oplus$ vers le haut.
- 5) Enfoncer les piles à l'aide du couvercle et glisser ce dernier en place pour fermer le logement des piles.
- 6) Reposer les vis sur le panneau arrière et brancher l'alimentation électrique.



• Pile auxiliaire

La pile auxiliaire permet d'entretenir la mémoire. Cette pile doit rester en place au moment où les piles principales sont changées pour éviter la disparition des programmes et des données.

Toujours avoir à l'esprit que si les piles principales et la pile auxiliaire sont enlevées en même temps, programmes et données seront effacés. Si toutes ces piles doivent être changées en même temps, appuyer sur le bouton de remise à zéro intégrale à l'aide d'un objet pointu après avoir mis en circuit l'alimentation électrique de l'unité centrale.

Notes:

1. Une utilisation fréquente de la sonnerie réduit la durée de vie des piles.
2. Si les piles principales doivent être changées, changer les deux piles en même temps.
3. Ne jamais jeter des piles au feu car elles risquent d'exploser.
4. Bien vérifier que les piles sont installées selon la polarité ($\oplus$, $\ominus$) prévue.

Conserver les piles hors de portée des enfants. En cas d'ingestion par accident, consulter immédiatement un médecin.

Arrêt auto alimentation

Cette fonction automatique d'économie d'énergie est prévue pour éviter toute consommation inutile d'énergie quand l'utilisateur oublie de couper l'alimentation électrique. L'alimentation électrique est coupée automatiquement au bout de 6 minutes après la fin d'une opération (sauf pendant l'exécution d'un programme) ou en état d'attente d'entrée par clavier à la suite de l'exécution d'une instruction INPUT ou PRINT.

Dans un tel cas, l'alimentation électrique peut être remise en circuit en plaçant l'interrupteur d'alimentation successivement sur arrêt et marche ou en appuyant sur la touche **ON/OFF**.

Note:

Même quand l'alimentation électrique est coupée, le contenu des variables, des programmes et de la BANQUE DES DONNEES est conservé mais les spécifications de mode ("WRT", "TRACE ON", "PRT ON" etc.) sont initialisés (immédiatement après remise en circuit de l'alimentation électrique).

Toutefois, les données et programmes sont effacés si toutes les piles doivent être changées en même temps, appuyez sur le bouton de remise à zéro. Les données et programmes sont effacés si toutes les piles doivent être changées en même temps, appuyez sur le bouton de remise à zéro.

Notes:

1. Une utilisation fréquente de la sonnerie retient la durée de vie des piles.
2. Si les piles principales doivent être changées, changer les deux piles en même temps.
3. Ne jamais jeter des piles au feu car elles risquent d'exploser.
4. Bien vérifier que les piles sont installées selon la polarité (+, -) prévue.

Conserver les piles hors de portée des enfants. En cas d'ingestion par accident, consulter immédiatement un médecin.

À l'arrêt, le sonnerie doit être coupée.

Arrêt auto alimentation

Cette fonction automatique d'économie d'énergie est prévue pour éviter toute consommation inutile d'énergie quand l'utilisateur oublie de couper l'alimentation électrique. L'alimentation électrique est coupée automatiquement au bout de 6 minutes après la fin d'une opération (sauf pendant l'exécution d'un programme) ou en état d'attente d'entrée par clavier à la suite de l'exécution d'une instruction INPUT ou PRINT.

Dans un tel cas, l'alimentation électrique peut être remise en circuit en plaçant l'interrupteur d'alimentation successivement sur arrêt et marche ou en appuyant sur la touche **ON**.

Si l'arrêt auto alimentation est activé, le sonnerie doit être coupée.

Après avoir branché l'alimentation électrique, appuyez sur la touche **ON**.

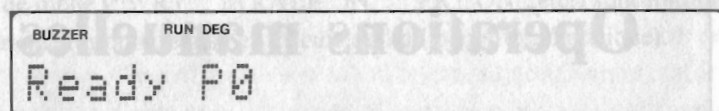
Opérations manuelles

2

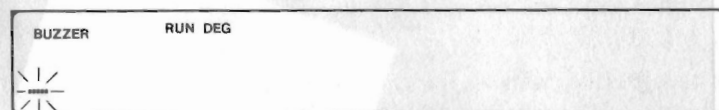
La meilleure façon de se familiariser avec l'ordinateur est de s'en servir. Même en cas d'erreur, la machine ne sera pas pour autant mise hors d'usage. "C'est en forgeant que l'on devient forgeron" dit le proverbe ... A vos claviers donc!

2-1 Première utilisation de l'ordinateur

Essayez votre ordinateur et notez comment il fonctionne. Tout d'abord, mettre l'interrupteur d'alimentation sur ON (marche) et l'affichage suivant apparaît sur l'écran.



Effacez cet affichage en appuyant sur la touche **CLS**. "Ready P0" disparaît. Puis le signe "—" va se mettre à clignoter à l'extrême gauche. Ce signe est le "curseur", il indique la position de départ pour l'écriture de caractères.



Ce curseur clignote lorsque l'ordinateur est en état dit "état d'attente d'entrée par clavier". En d'autres termes, le curseur clignote pour indiquer que l'ordinateur attend pour exécuter un calcul ou un ordre. Le curseur est en général représenté par le signe "—" clignotant, mais au fur et à mesure que des caractères sont écrits il peut parfois se transformer en un symbole "■" clignotant. Sur cet ordinateur, une ligne comprend 62 caractères au maximum. Le symbole "■" apparaît sous forme d'un signal de mise en garde quand le nombre de caractères écrits atteint plus de 55.

"BUZZER", "RUN" et "DEG" peuvent également s'inscrire en haut de l'écran. Ces indications sont des affichages de mode et indiquent l'état de l'ordinateur. "RUN" indique le mode RUN dans lequel il est possible d'effectuer des calculs manuels et d'exécuter un programme. "BUZZER" indique que la sonnerie est en fonction. Cette sonnerie retentit à chaque entrée effectuée par l'intermédiaire du clavier. "DEG" indique que l'unité d'angle est le degré. Les autres unités d'angle possibles sont le radian ("RAD" est indiqué), spécifiée par **MOD** **[S]** et le grade ("GRA" est indiqué), spécifiée par **MOD** **[G]**. Attention à ces unités d'angle pour la manipulation des fonctions trigonométriques, des fonctions trigonomé-

2-1 Première utilisation de l'ordinateur

triques inverses ou des transformations des coordonnées. Une fois qu'une unité d'angle a été spécifiée, elle reste effective même quand l'alimentation électrique est coupée.

Les autres modes affichés sont le mode d'écriture de programmes ("WRT" est indiqué) spécifié par **MOD** **[1]**, le mode analyse ("TRACE ON" est indiqué, voir page 108) spécifié par **MOD** **[2]**, le mode sortie sur imprimante ("PRT ON" est indiqué, voir page 117) spécifié par **MOD** **[7]**, le mode MEMO IN pour la fonction DATA BANK ("MEMO IN" est indiqué, voir page 218) spécifié par **MOD** **[S]** et le mode extension ("EXT" est indiqué) spécifié par **EXT**.

Vous apprendrez à utiliser ces modes au fur et à mesure de votre apprentissage de l'ordinateur.

A titre d'exercice, appuyez sur les touches correspondantes pour afficher ces modes. Si une confusion quelconque survient dans l'affichage des modes, couper l'alimentation électrique et la rebrancher de nouveau.

Commençons par un calcul simple.

Exemple:

$$123 + 456 = 579$$

Appuyer sur **CLS**.

Appuyer sur les touches correspondant à l'équation précédente.

$$123+456$$

Appuyer ensuite sur **EXE** et non sur **=** pour obtenir le résultat.

EXE

$$579$$

Le calcul est aussi simple qu'avec une calculatrice ordinaire, n'est-ce pas ? Effectuons maintenant un calcul avec multiplication et addition.

Exemple:

$$33 \times 5 + 16 = 181$$

Supposons que 34 ait été introduit par erreur au lieu de 33.

$$34 \times 5 + 16$$

$$34 \times 5 + 16$$

Vous avez noté l'erreur, mais nous allons la corriger. Appuyer sur la touche de déplacement du curseur ($\leftarrow$) et amener celui-ci sur le chiffre erroné.

$$\leftarrow \leftarrow \leftarrow \leftarrow \leftarrow \leftarrow$$

$$34 \times 5 + 16$$

Le curseur et 4 clignotent tour à tour.

Appuyer sur la bonne touche (3).

$$3$$

$$33 \times 5 + 16$$

Maintenant que l'opération a été formulée correctement, cherchons la réponse.

$$\text{EXE}$$

$$181$$

Exemple:

$$26 \times 7 + 23 = 205$$

Supposons que 32 ait été introduit par erreur au lieu de 23.

$$26 \times 7 + 32$$

$$26 \times 7 + 32$$

Dans ce cas, la touche BS est particulièrement pratique pour corriger.

$$\text{BS}$$

$$26 \times 7 + 3$$

$$\text{BS}$$

$$26 \times 7 +$$

Après avoir effacer l'erreur par une opération de retour en arrière, entrer les chiffres exacts et appuyer sur EXE .

$$23$$

$$26 \times 7 + 23$$

$$\text{EXE}$$

$$205$$

* Contrairement aux touches $\leftarrow$ et $\rightarrow$, la touche BS efface le caractère placé à gauche du curseur et déplace d'un espace vers la gauche tous les caractères se trouvant à droite. Attention en utilisant cette touche BS .

2-2 Quatre opérations arithmétiques pour commencer

Comme nous venons de le voir, il est facile de corriger une erreur à l'aide des touches de déplacement du curseur ou de la touche BS quand cette erreur est repérée pendant le processus d'introduction. Par contre, si l'erreur est notée après que la touche EXE ait été enclenchée, utiliser la fonction REPLAY (rappel) (touche R) ou reprendre le calcul à partir du début.

Ces touches sont disposées comme sur un clavier de machine à écrire (disposition ASCII).

Ecrivez d'abord en majuscules d'imprimerie.

Exemple: Entrer A, B, C, X, Y et Z.

Entrer tout d'abord A, B et C.

$$\text{A B C}$$

Entrer ensuite X, Y et Z.

$$\text{X Y Z}$$

Insérer maintenant un espace entre ABC et XYZ. Amener le curseur en position de X.

$$\leftarrow \leftarrow \leftarrow$$

$$\text{A B C X Y Z}$$

Créer un espace.

$$\text{INS}$$

$$\text{A B C } \text{X Y Z}$$

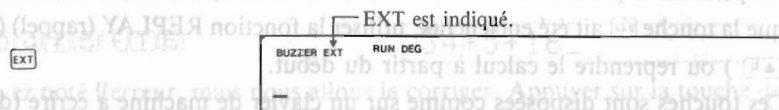
Pour insérer de cette manière un espace entre deux caractères, amener le curseur à l'endroit où l'espace doit être inséré et appuyer sur la touche INS . Pour insérer plusieurs espaces, garder cette touche appuyée.

En plus des chiffres et des lettres majuscules, cet ordinateur peut afficher des lettres minuscules et des caractères spéciaux. Pour l'affichage de ces caractères, on utilise le mode extension. Voir page 9.

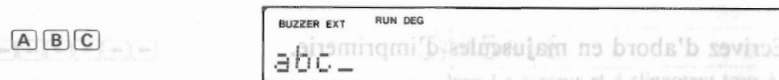
Exemple:

Afficher les minuscules a, b et c.

Spécifier tout d'abord le mode extension.

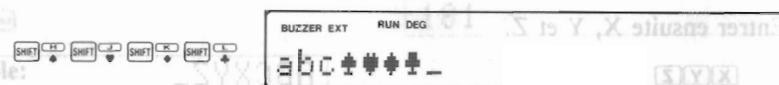


Entrer ensuite A,B et C.

**Exemple:**

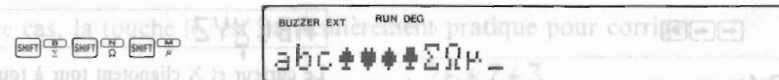
Afficher les marques ♣♣♣♣.

L'ordinateur étant déjà en mode extension, il suffit d'appuyer sur chacune des touches correspondantes après avoir appuyer sur la touche [SHIFT].

**Exemple:** $\times 7 + 23 = 205$

Afficher les symboles ΣΠ.

Appuyer sur les touches suivantes en mode extension.



Les marques et les symboles précédents existent et peuvent être diversement utilisés. Pour libérer le mode extension appuyer une nouvelle fois sur [EXT] et le symbole "EXT" disparaît.

Vous devez maintenant avoir compris le fonctionnement des touches. Pendant que vous pratiquez sur le clavier, il peut arriver que "Error 2" apparaisse sur l'écran et que la touche appelée soit bloquée. Il ne s'agit pas d'une panne mais d'un message indiquant qu'une opération erronée a été effectuée. C'est pourquoi ce type de message est dit "message erreur". Dans un tel cas, appuyer sur la touche [BRK]. Le message erreur disparaît de l'écran et l'ordinateur est de nouveau opérationnel. Il existe divers types de message erreur. Pour le détail voir page 248.

2-2 Quatre opérations arithmétiques pour commencer

Essayons des calculs simples. Noter tout de suite qu'il existe un ordre de priorité dans les opérations, à savoir que multiplication et division prennent le pas sur addition et soustraction.

Exemple 1:

$$23 + 4.5 - 53 = -25.5$$

Opération:

23+4.5-53=

-25.5

* Désormais, les chiffres ne seront plus encadrés pour indiquer une opération au clavier.

Exemple 2:

$$56 \times (-12) \div (-2.5) = 268.8$$

Opération:

56*(-12)/2.5=

268.8

* Pour les nombres négatifs, appuyer sur la touche [(-)] avant d'appuyer sur la touche numérique.

Exemple 3:

$$7 \times 8 - 4 \times 5 = 36$$

Opération:

7*8-4*5=

36

* Les multiplications sont effectuées en premier, puis la soustraction.

Exemple 4:

$$(4.5 \times 10^{75}) \times (-2.3 \times 10^{-78}) = -0.01035$$

Opération:

4.5E75*(-2.3E78)=

-0.01035

* Pour afficher l'exposant, entrer celui-ci après avoir appuyer sur la touche [E].

En plus des calculs précédents, cet ordinateur est capable d'effectuer des calculs algébriques à l'aide de variables. Ces calculs sont particulièrement pratiques quand une certaine valeur est utilisée de manière répétitive.

Par exemple:

$$3x + 5 =$$

$$4x + 6 =$$

$$5x + 7 =$$

Si la valeur de x dans ces calculs est 123,456 il sera fastidieux d'appuyer toujours sur les mêmes touches. Le calcul algébrique avec variable est une méthode qui permet de simplifier un tel travail. On utilise la variable X .

Assigner tout d'abord la valeur 123,456 à la variable X .

$\boxed{X} \boxed{=} \boxed{1} \boxed{2} \boxed{3} \boxed{.} \boxed{4} \boxed{5} \boxed{6} \boxed{EXE}$

Où $\boxed{=}$ ne signifie pas "égal à", mais "partie droite affectée à partie gauche". Commençons maintenant les calculs.

$3 \boxed{*} \boxed{X} \boxed{+} \boxed{5} \boxed{EXE}$

375.368

$4 \boxed{*} \boxed{X} \boxed{+} \boxed{6} \boxed{EXE}$

499.824

$5 \boxed{*} \boxed{X} \boxed{+} \boxed{7} \boxed{EXE}$

624.28

En utilisant une variable, des calculs répétitifs peuvent ainsi devenir aussi simples que dans notre exemple.

Cet ordinateur est pourvu de 26 variables, de A à Z, qui permettent la mise en mémoire de diverses valeurs.

Dans l'exemple précédent, la valeur de la variable X est constante, alors que la formule de calcul change.

Noter que dans le cas de calculs où les formules sont constantes mais où la valeur de la variable change, l'ordinateur travaille d'une manière différente. Par exemple, pour le calcul de l'égalité " $3x + 5 =$ " où x varie de 123 à 456 à 789, l'ordinateur se sert d'une fonction pour stocker les expressions numériques (formules de calcul). Cette façon de procéder sera décrite en Chapitre 3.

2-3 Notes à propos des calculs

Ordre de priorité dans les calculs

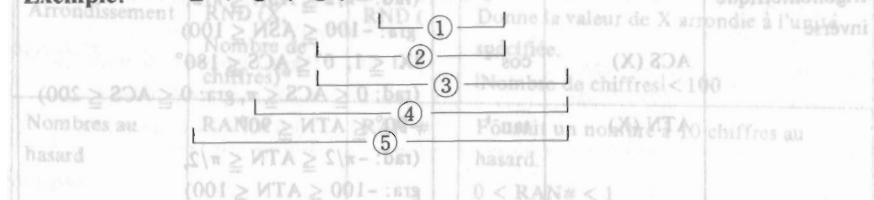
Comme mentionné plus haut, les calculs sont sujets à une règle d'"ordre de priorité" (logique algébrique vraie) qui stipule que multiplication et division prennent le pas sur addition et soustraction. Cet ordinateur décide automatiquement de l'ordre de priorité. Il suffit d'entrer une expression numérique et le résultat exact sera affiché.

L'ordre de priorité dans les calculs est le suivant:

- 1) Fonctions (sin, cos, tan etc...)
- 2) Puissance ($\uparrow$)
- 3) Multiplication ($*$), division ($/$)
- 4) Addition (+), soustraction (-)

Les calculs sont effectués conformément à l'ordre de priorité. Si les calculs à effectuer occupent le même ordre de priorité, le calcul se trouvant à gauche est effectué en premier. Dans le cas de parenthèses, la priorité absolue doit être donnée aux calculs entre parenthèses.

Exemple: $2 + 3 * \text{SIN}(17+13) \uparrow 2 = 2.75$



Nombre de chiffres entrés/sortis et chiffres de calcul

Cet ordinateur accepte une gamme de valeurs entrées (nombre de chiffres entrés) d'un maximum de 12 chiffres pour la mantisse et de 2 chiffres pour un exposant. Les mêmes limites s'appliquent aux calculs internes.

La limite d'affichage d'une valeur (nombre de chiffres sortis) est de 10 chiffres pour la mantisse et de 2 chiffres pour l'exposant.

Exemple:

$1 \boxed{\div} \boxed{2} \boxed{3} \boxed{4} \boxed{5} \boxed{6} \boxed{7} \boxed{8} \boxed{9} \boxed{1} \boxed{2} \boxed{EXE}$

1.234567891

$12345678912 \boxed{*} \boxed{1} \boxed{0} \boxed{0} \boxed{EXE}$

1.234567891E12

$12345678912 \boxed{*} \boxed{-} \boxed{1} \boxed{0} \boxed{0} \boxed{EXE}$

-1.234567891E12

2-4 Calculs de fonctions

En plus des quatre opérations arithmétiques, cet ordinateur peut effectuer des calculs de fonction. Les fonctions peuvent être utilisées dans les programmes, mais nous décrivons ici les opérations manuelles.

Cet ordinateur est pourvu des fonctions suivantes:

Nom des fonctions	Format	Fonction et gamme d'entrée
Fonction trigonométrique	SIN (Expression numérique) * ci-après X	sin $ X < 1440^\circ$ (8π rad, 1600 gra)
	COS (X)	cos $ X < 1440^\circ$ (8π rad, 1600 gra)
	TAN (X)	tan $ X < 1440^\circ$ (8π rad, 1600 gra) sauf quand $ X $ est un multiple impair de 90° ($\pi/2$ rad, 100 gra)
Fonction trigonométrique inverse	ASN (X)	$\sin^{-1}$ $ X \leq 1$, $-90^\circ \leq \text{ASN} \leq 90^\circ$ (rad: $-\pi/2 \leq \text{ASN} \leq \pi/2$, gra: $-100 \leq \text{ASN} \leq 100$)
	ACS (X)	$\cos^{-1}$ $ X \leq 1$, $0^\circ \leq \text{ACS} \leq 180^\circ$ (rad: $0 \leq \text{ACS} \leq \pi$, gra: $0 \leq \text{ACS} \leq 200$)
	ATN (X)	$\tan^{-1}$ $-90^\circ \leq \text{ATN} \leq 90^\circ$ (rad: $-\pi/2 \leq \text{ATN} \leq \pi/2$, gra: $-100 \leq \text{ATN} \leq 100$)
Fonction hyperbolique	HYP SIN (X)	sinh $ X \leq 230,2585092$
	HYP COS (X)	cosh $ X \leq 230,2585092$
	HYP TAN (X)	tanh $ X < 10^{100}$
Fonction hyperbolique inverse	HYP ASN (X)	$\sinh^{-1}$ $ X < 5 \times 10^{99}$
	HYP ACS (X)	$\cosh^{-1}$ $1 \leq X < 5 \times 10^{99}$
	HYP ATN (X)	$\tanh^{-1}$ $ X < 1$
Racine carrée	SQR (X)	$\sqrt{x}$ $X \geq 0$
Racine cubique	CUR (X)	$\sqrt[3]{x}$ $ X < 10^{100}$
Puissance	X ↑ X	x^y $x < 0 \rightarrow y$: nombre naturel

2-4 Calculs de fonctions

Nom des fonctions	Format	Fonction et gamme d'entrée
Fonction exponentielle	EXP (X)	e^x $-10^{100} < X \leq 230,2585092$
Logarithme commun	LOG (X)	$\log_{10} x$ $X > 0$
Logarithme naturel	LN (X)	$\log_e x$ $X > 0$
Intégrale	INT (X)	$[x]$ Donne l'intégrale maximum ne dépassant pas X (égal à la fonction de Gauss $[x]$)
Fraction	FRAC (X)	FRAC Donne la partie décimale de X
Valeur absolue	ABS(X)	$ x $ Donne la valeur absolue de X
Signe	SGN (X)	$\text{sgn } x$ 1 quand $X > 0$ 0 quand $X = 0$ -1 quand $X < 0$
Arrondissement	RND (X, Nombre de chiffres)*	RND () Donne la valeur de X arrondie à l'unité spécifiée. Nombre de chiffres < 100
Nombres au hasard	RAN #	RAN # Fournit un nombre à 10 chiffres au hasard. $0 < \text{RAN} \# < 1$
π	π	π Donne la valeur approximative du quotient de la circonférence d'un cercle par son diamètre.
Conversion décimale → sexagésimale	DMSS (X)*	DMSS () Convertit un nombre décimal donné comme X en chaîne de caractères sexagésimale en degrés, minutes et secondes. $ X < 10^5$
Conversion sexagésimale → décimale	DEG (deg. [, min. [, sec.]])*	DEG () $\text{DEG}(x, y, z) = x + y/60 + z/3600$. $ \text{DEG}(x, y, z) < 10^{100}$

Nom des fonctions	Format	Fonction et gamme d'entrée
Conversion décimale → hexadécimale	HEX\$ (X)*	HEX\$ (Convertit la valeur de X en chaîne de caractères hexadécimale de 4 chiffres -32769 < X < 65536
Conversion hexadécimale → décimale	&H Chaîne de caractères hexadécimale	&Hx La chaîne de caractères contient un nombre hexadécimale dans les limites de 4 caractères.
Factorielle	FACT (X)	x! $0 \leq X \leq 69$ (0 et nombre entier positif)
Permutation	NPR (n, r)*	nPr $0 \leq r \leq n < 10^{10}$ (0 et nombre entier positif)
Combinaison	NCR (n, r)*	nCr $0 \leq r \leq n < 10^{10}$ (0 et nombre entier positif)
Transformation des coordonnées rectangulaires en polaires	POL (X, Y)* X, Y: Expressions numériques	POL ($ X < 10^{100}$, $ Y < 10^{100}$, $ X + Y \neq 0$ r est donné comme valeur de fonction pour affectation à la variable X alors que la valeur de θ est affectée à la variable Y.
Transformation des coordonnées polaires en rectangulaires.	REC (r, θ)* r, θ : Expressions numériques	REC ($0 \leq r < 10^{100}$, $ \theta < 1440^\circ$ (8π rad, 1600 gra) Donne x comme valeur de fonction pour affectation à la variable X alors que la valeur de y est affectée à la variable Y.

Note:

Pour les fonctions repérées par un astérisque, les paramètres doivent être placés entre parenthèses.

*Certaines combinaisons ou permutations peuvent provoquer des erreurs dues à un dépassement de capacité pendant les calculs internes.

Procédons maintenant aux calculs à l'aide des fonctions. Les fonctions utilisées de manière fréquente peuvent être introduites par une seule touche correspondant aux dites fonctions.

• Fonctions trigonométriques (sin, cos, tan) et fonctions trigonométriques inverses ($\sin^{-1}$, $\cos^{-1}$, $\tan^{-1}$)

Avec ces fonctions, ne pas oublier de spécifier l'unité d'angle (degré, radian, grade).

Exemple:
 $\sin 12.3456^\circ = 0.2138079201$

Opération:
[MODE] [4] (Unité d'angle: degré (DEG))
[sin] 12.3456 [EXE] 0.2138079201

Exemple:
 $2 \cdot \sin 45^\circ \times \cos 65.1^\circ = 0.5954345575$

Opération:
2 [x] [sin] 45 [x] [cos] 65.1 [EXE] 0.5954345575

Exemple:
 $\sin^{-1} 0.5 = 30^\circ$

Opération:
[SHIFT] [ASN] 0.5 [EXE] 30

Exemple:
 $\cos(\frac{\pi}{3} \text{ rad}) = 0.5$

Opération:
[MODE] [5] (Unité d'angle: radian (RAD))
[cos] [SHIFT] [pi] [3] [EXE] 0.5

Exemple:

$$\cos^{-1} \frac{\sqrt{2}}{2} = 0.7853981634 \text{ rad}$$

Opération:
 $\text{SHIFT} \text{ACS} \left(\left(\frac{\sqrt{2}}{2} \right) \right) \text{EXE}$

0.7853981634

Exemple:

$$\tan(-35\text{gra}) = -0.6128007881$$

 $\text{MODE} \left(\frac{\square}{\square} \right) \text{ (Unité d'angle: grade (GRA))}$
 $\text{tan} \left(-35 \right) \text{EXE}$

-0.6128007881

- Fonctions hyperboliques (sinh, cosh, tanh) et fonctions hyperboliques inverses ($\sinh^{-1}$, $\cosh^{-1}$, $\tanh^{-1}$)

Avec ces fonctions, appuyer sur hvp puis sur les mêmes touches que pour les fonctions trigonométriques et trigonométriques inverses.

Exemple:

$$\sinh\left(-\frac{\pi}{2}\right) = -2.301298902$$

Opération:
 $\text{hvp} \text{sin} \left(\left(-\frac{\pi}{2} \right) \right) \text{EXE}$

-2.301298902

Exemple:

$$\cosh^{-1} 1.5 = 0.9624236501$$

Opération:
 $\text{hvp} \text{SHIFT} \text{ACS} 1.5 \text{EXE}$

0.9624236501

- Fonctions logarithmiques (log, ln) Fonction exponentielle (e^x) et Fonction puissance (x^y)

Exemple:

$$\log 1.23 = \log_{10} 1.23 = 0.08990511144$$

Opération:
 $\log 1.23 \text{EXE}$

0.08990511144

Exemple:

$$\ln 90 = \log_e 90 = 4.49980967$$

Opération:
 $\ln 90 \text{EXE}$

4.49980967

Exemple:

$$e^5 = 148.4131591$$

Opération:
 $\text{EXP} 5 \text{EXE}$

148.4131591

Exemple:

$$123^2 = 15129$$

Opération:
 $123 \text{X}^2 \text{EXE}$

15129

Exemple:

$$123^3 = 1860867$$

Opération:
 $123 \text{X}^3 \text{EXE}$

1860867

Exemple:

$$10^{1.23} = 16.98243652$$

Opération:
 $\text{INV} 1.23 \text{EXE}$

16.98243652

Exemple:

$$5.6^{2.3} = 52.58143837$$

Opération:
 $5.6 \uparrow 2.3 \text{EXE}$

52.58143837

Exemple:

$$123^{\frac{1}{7}} \approx \sqrt[7]{123} = 1.988647795$$

Opération:
 $123 \uparrow \left(\frac{1}{7} \right) \text{EXE}$

1.988647795

- Autres fonctions ($\sqrt{}$, $\sqrt[3]{}$, SGN, RAN#, RND, ABS, INT, FRAC)

Exemple:

$$\sqrt{2} + \sqrt{5} = 3.65028154$$

Opération:

$\sqrt{}$ 2 $+$ $\sqrt{}$ 5 EXE

3.65028154

Exemple:

$$\sqrt[3]{27} = 3$$

Opération:

$\sqrt[3]{}$ 27 EXE

3

Exemple:

Conversion en signes (nombre positif $\rightarrow$ 1, nombre négatif $\rightarrow$ -1, 0 $\rightarrow$ 0)

Opération:

SGN 6 EXE

1

SGN 0 EXE

0

SGN $\pm$ 2 EXE

-1

Exemple:

Génération de nombre au hasard ($0 < \text{RAN\#} < 1$ nombre pseudo aléatoire)

Opération:

RAN# EXE

0.0756647782

(Cette valeur n'est pas nécessairement affichée)

Exemple:

Pour arrondir le résultat de $12,3 \times 4,56$ au dixième.

$$12.3 \times 4.56 = 56.088$$

Opération:

$\sqrt{}$ 12.3 $\times$ 4.56 $\div$ 2 EXE

56.1

Exemple:

$$|-78.9 \div 5.6| = 14.08928571$$

Opération:

ABS $\div$ 78.9 $\div$ 5.6 EXE

14.08928571

Exemple:

La partie entière de $7800 \div 96$ est 81.

Opération:

INT 7800 $\div$ 96 EXE

81

* INT x donne le nombre entier non supérieur à x.

Exemple:

La partie décimale de $7800 \div 96$ est 0,25.

Opération:

FRAC 7800 $\div$ 96 EXE

0.25

- Conversion décimale-sexagésimale (DEG, DMSS)

Exemple:

$$14^{\circ} 25' 36'' = 14.42666667^{\circ}$$

Opération:

DEG 14 $\div$ 25 $\div$ 36 EXE

14.42666667

Exemple:

$$12.3456^{\circ} = 12^{\circ} 20' 44.16''$$

Opération:

DMSS 12.3456 EXE

12°20'44.16

Exemple:

$$\sin 63^{\circ} 52' 41'' = 0.897859012$$

Opération:

MOD 4 $\div$ 63 $\div$ 52 $\div$ 41 EXE

0.897859012

- Conversion décimale-hexadécimale (&H,HEXS)

* A, B, C, D, E et F en nombres hexadécimaux correspondent aux nombres décimaux 10 à 15.

Exemple:

Conversion d'un nombre hexadécimal en nombre décimal.

Opération:

&H 10 EXE

16

&H 7FFF EXE

32767

&H 8000 EXE

-32768

&H FFFF EXE

-1

Exemple:

Conversion d'un nombre décimal en nombre hexadécimal.

Opération:

SHIFT HEXS 100 EXE

0064

SHIFT HEXS 1000 EXE

03E8

SHIFT HEXS 32768 EXE

8000

SHIFT HEXS 32767 EXE

7FFF

SHIFT HEXS 65535 EXE

FFFF

- Factoriel, Permutation et Combinaison (FACT, NPR et NCR)

Exemple:

$10! = 3628800$

Opération:

SHIFT FACT 10 EXE

3628800

Exemple:

${}_{10}P_4 = 5040$

Opération:

SHIFT NPR 10 4 EXE

5040

Exemple:

$C_4 = 210$

Opération:

SHIFT REC 10 4 EXE

210

- Transformation des coordonnées de rectangulaire à polaire (REC, POL)

On suppose ici que θ des coordonnées polaires (r, θ) est exprimé en radians.

Exemple:

Le point (5, $\pi/6$) dans les coordonnées polaires est (4,330127019, 2,5) dans les coordonnées rectangulaires.

Opération:

MODE S (Unité d'angle: radian)

SHIFT REC 5 SHIFT π 6 EXE

(x)

4.330127019

Y EXE

(y)

2.5

Exemple:

Le point (1,1) dans les coordonnées rectangulaires est (1,414213562, 0,7853981634) dans les coordonnées polaires.

Opération:

SHIFT POL 1 1 EXE

(r)

1.414213562

Y EXE

(θ)

0.7853981634

- * Dans ces fonctions, les résultats sont affectés aux variables X et Y. Les valeurs sorties sont les mêmes que le contenu de la variable X.
- * Dans ces fonctions, spécifier l'unité d'angle est aussi important que pour les fonctions trigonométriques et trigonométriques inverses.

- **Spécification du nombre de chiffres significatifs et du nombre de chiffres après la virgule**

On utilise "SET" pour opérer.

Spécification du nombre de chiffres significatifs SET En ($n = 0$ à 9)

Spécification du nombre de chiffres après la virgule SET Fn ($n = 0$ à 9)

Spécification annulée SET N

* "SET E0" est utilisé pour spécifier le nombre de chiffres significatifs sur 10 chiffres.

* Quand une spécification est effectuée, le résultat est affiché en fonction du nombre de chiffres spécifié. (Le chiffre suivant le dernier chiffre spécifié est arrondi). La valeur originale reste dans l'ordinateur.

Exemple:

$$100 \div 6 = 16.666666666\ldots$$

Opération:

SHIFT SET E 4 EXE (Nombre spécifié de chiffres significatifs: 4)

$$100 \div 6 = 16.667E01$$

Exemple:

$$123 \div 7 = 17.57142857\ldots$$

Opération:

SHIFT SET F 2 EXE (Nombre spécifié de chiffres après la virgule: 2)

$$123 \div 7 = 17.57$$

Exemple:

$$1 \div 3 = 0.3333333333\ldots$$

Opération:

SHIFT SET N EXE (Spécification annulée)

$$1 \div 3 = 0.3333333333$$

2-5 Calculs de statistiques

Au bureau comme à l'atelier, les calculs statistiques s'avèrent indispensables pour l'analyse de données recueillies ou lorsque prévisions ou estimations fournissent de nouvelles données. Cet ordinateur est équipé de fonctions de calcul de statistiques qui permettent d'effectuer ces calculs simplement et facilement. Coefficients de corrélation ou valeurs estimatives seront ainsi rapidement obtenus.

- **Entrée des données statistiques**

Avant de procéder à un calcul statistique, libérer les mémoires de statistique exclusives. Entrer ensuite les données statistiques à l'aide de la touche [STAT] en mode "RUN".

Données à une variable

{ Données individuelles données [STAT]

{ Données identiques multiples . . données [i] fréquence [STAT]

Données à variable appariée

{ Données individuelles données x [x] données y [STAT]

{ Données identiques multiples . . données x [x] données y [i] fréquence [STAT]

Voici un exemple d'entrée de données statistiques à l'aide de la touche [STAT].

x	1	3	5	5	3	3	3
y	2	4	4	6	3	3	3

Données entrées

Efface 1 et 2 en dernier.

Opération:

MODE 0 (Mode RUN)
 SHIFT 9 SHIFT CLEAR (Mémoires statistiques effacées.)
 EXE

1 [x] 2

[STAT] 3 [x] 4 [STAT]

5 [STAT] *1

[x] 6 [STAT] *2

3 [x] 3 [STAT]

[x] 3 2 [STAT]

1 [x] 2 [SHIFT] SDEL *3 (Données "1,2" annulées.)

Ready P0

STAT CLEAR

1,2

STAT 1,2

STAT 3,4

STAT 5,4

STAT 5,6

STAT 3,3

STAT 3,3;2

SDEL 1,2

Appuyer sur la touche $\boxed{\text{STAT}}$ après avoir entré les données. "STAT ..." est affiché.

- * 1. Les données y sont facultatives si on utilise "données x ($\boxed{\text{;}}$ fréquence) $\boxed{\text{STAT}}$ ". Dans un tel cas, les valeurs de y sont inchangées. ($\boxed{\text{;}}$ fréquence) est facultatif.)
- * 2. Si les données x ne sont pas introduites et que l'on écrit simplement "données y ($\boxed{\text{;}}$ fréquence) $\boxed{\text{STAT}}$ ", celles-ci restent inchangées. ($\boxed{\text{;}}$ fréquence) est facultatif.)
- * 3. Si on appuie sur $\boxed{\text{SHIFT}} \boxed{\text{DEL}}$ au lieu de $\boxed{\text{STAT}}$ après l'entrée des données, celles-ci sont effacées de la mémoire statistique.
- * Les données statistiques entrées restent en mémoire même si l'alimentation électrique est coupée sauf si elles ont été effacées comme indiqué précédemment ou si la commande STAT CLEAR (effaçage des statistiques) est exécutée en appuyant sur $\boxed{\text{SHIFT}} \boxed{\text{STAT}} \boxed{\text{SHIFT}} \boxed{\text{CLEAR}} \boxed{\text{EXE}}$. Par conséquent, pour entrer de nouvelles données statistiques, ne pas oublier d'exécuter d'abord la commande STAT CLEAR pour libérer les mémoires de statistiques.
- * Si la "fréquence" n'est pas spécifiée, celle-ci est considérée comme égale à 1.
- * Noter que la touche $\boxed{\text{STAT}}$ et les touches $\boxed{\text{SHIFT}} \boxed{\text{STAT}}$ sont fonctionnellement différentes. La première permet d'entrer des données statistiques dans des calculs manuels alors que les secondes constituent une commande une-touche pour l'entrée de données, STAT CLEAR, STAT LIST etc... dans un programme que nous décrirons plus loin.

Exemple :

Opération	Mode RUN	Mémoire statistique	Effacée
1. 2. 3. 4. 5. 6. 7. 8. 9. 0. =	1. 2. 3. 4. 5. 6. 7. 8. 9. 0. =	1. 2. 3. 4. 5. 6. 7. 8. 9. 0. =	1. 2. 3. 4. 5. 6. 7. 8. 9. 0. =

• Sortie des statistiques

Les statistiques peuvent être obtenues à l'aide des commandes STAT LIST, STAT LIST 1, STAT LIST 2, EOX et EOY. Les calculs de statistiques suivants peuvent être effectués avec cet ordinateur :

Statistiques	Equation
n	Nombre de données statistiques traitées
$\sum x$	Somme des données x
$\sum y$	Somme des données y
$\sum x^2$	Somme des carrés des données x
$\sum y^2$	Somme des carrés des données y
$\sum xy$	Somme des produits des données x et des données y
$\sum x/n$	Moyenne des données x
$\sum y/n$	Moyenne des données y
s_{x-1}	Moyenne des différences d'un échantillon des données x
s_{y-1}	Moyenne des différences d'un échantillon des données y
s_x	Moyenne des différences d'une population des données x
s_y	Moyenne des différences d'une population des données y
a	Constante de régression linéaire
b	Coefficient de régression linéaire
r	Coefficient de corrélation
EOX	Valeur estimée de x pour une valeur donnée de y
EOY	Valeur estimée de y pour une valeur donnée de x

STAT LIST (ou STAT LIST 0) permet de sortir l'ensemble des statistiques. STAT LIST 1 concerne les statistiques à une seule variable uniquement et STAT LIST 2 les statistiques à variable appariée.

Le tableau suivant donne le détail:

STAT LIST {0}	STAT LIST 1	STAT LIST 2
n	n	n
Σx	Σx	Σx
Σy	Σy	Σy
Σx^2	Σx^2	Σx^2
Σy^2	Σy^2	Σy^2
Σxy	Σxy	Σxy
$\bar{x}$	$\Sigma x/n$	$\Sigma x/n$
$\bar{y}$	$\Sigma y/n$	$\Sigma y/n$
$x\sigma_n$	$x\sigma_n$	$x\sigma_n$
$y\sigma_n$	$y\sigma_n$	$y\sigma_n$
$x\sigma_{n-1}$	$x\sigma_{n-1}$	$x\sigma_{n-1}$
$y\sigma_{n-1}$	$y\sigma_{n-1}$	$y\sigma_{n-1}$
a	a	a
b	b	b
r	r	r
Sortie de toutes les statistiques	Sortie des statistiques à variable unique	Sortie des statistiques à variable appariée

Formule de la régression: $y = a + b \cdot x$

EOX et EOY sont traitées comme des fonctions. Elles peuvent être calculées comme des fonctions numériques ordinaires; ("valeur y de EOX" ou "valeur x de EOY". Quand x et y sont des variables ou des valeurs numériques, les parenthèses sont facultatives tout comme dans le cas de SIN etc.

Pour bien maîtriser les calculs de statistiques, effectuer maintenant les exercices suivants:

Exercice:

Le tableau à droite fournit un état d'expédition d'un produit x et d'un produit y. Déterminer la variance des expéditions en calculant la moyenne des différences.

Date	4	5	6	7	8
Produit					
x	2	2	5	8	8
y	1	5	5	5	9

Opération:

Entrer tout d'abord les données statistiques fournies par le tableau. (Utiliser le mode RUN.)

STAT
SHIFT 2 SHIFT CLEAR EXE

2 1 STAT

2 5 STAT

...

Entrer toutes les données x et y.

Sortir ensuite les statistiques à variable appariée à l'aide de STAT LIST 2.

Quand cette opération est terminée, sortir tous les résultats statistiques l'un après l'autre à l'aide de STAT LIST.

(Nombre de données)

STAT
SHIFT 2 SHIFT LIST EXE

(Somme des données x)

EXE

(Somme des données y)

EXE

(Somme des carrés des données x)

EXE

(Somme des carrés des données y)

EXE

(Somme des produits des données x et des données y)

EXE

(Moyenne des données x)

EXE

(Moyenne des données y)

EXE

(Moyenne des différences de x)

EXE

(Moyenne des différences de y)

EXE

$n/n = 5.671428571$
$\Sigma x = 25.49392438$
$\Sigma y = 25.5235819$
$\Sigma x^2 = 161.3473381$
$\Sigma y^2 = 157.5008409$
$\Sigma xy = 149.221646$
$\Sigma x/n = 5.42512973$
$\Sigma y/n = 5.628252383$
$x\sigma_n = 2.683281573$
$y\sigma_n = 2.529822128$

Le reste est omis.

La comparaison des produits x et y, sur la base des résultats des calculs précédents, montre que le total des sommées et les valeurs moyennes sont égales, mais que la moyenne des différences est plus grande pour le produit x. On peut donc penser que la variance dans les expéditions de ce produit est plus grande.

Trouver maintenant les coefficients de corrélation et les valeurs estimées par un calcul de la régression avec les données appariées.

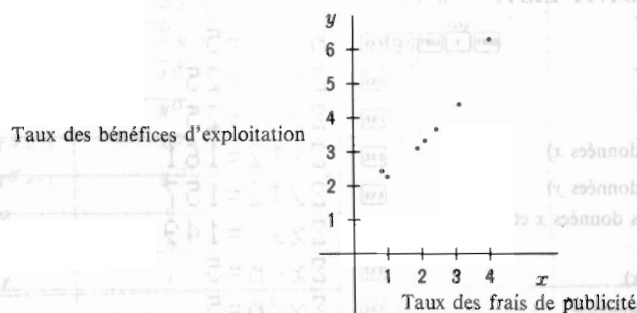
Exercice:

Le tableau ci-dessous donne les taux des frais de publicité (frais de publicité/frais de fonctionnement) et les taux des bénéfices d'exploitation (bénéfice d'exploitation/montant des ventes) pour une chaîne de sept supermarchés l'an dernier. La publicité a-t-elle été efficace?

Magasin	1	2	3	4	5	6	7
Taux des frais de publicité (%)	0.8	2.1	2.5	1.8	3.1	4.0	1.0
Taux des bénéfices d'exploitation (%)	2.5	3.4	3.7	3.2	4.3	6.3	2.3

Approche:

Tracer une courbe de dispersion à partir du tableau.



La courbe de dispersion laisse penser que les bénéfices ont augmenté avec la publicité. La courbe reliant les différents points tracés est dite courbe de régression. Ici elle est pratiquement droite et porte donc le nom de régression linéaire. Elle s'exprime par l'égalité $y = a + b \cdot x$ où a est la constante de régression linéaire et b le coefficient de régression linéaire.

Le coefficient de corrélation (r) est dans les limites de $-1 \leq r \leq 1$. La relation de corrélation est positive quand $0 < r \leq 1$, négative quand $-1 \leq r < 0$ et nulle quand $r = 0$. Entrer maintenant les données concernant les sept magasins et calculer les statistiques.

montrer la variance des expéditions en calculant la moyenne des différences.

Opération:

(Mémoires statistique libérée)

0.8 2.5

2.1 3.4

⋮

(Entrer toutes les données appariées.)

Sortir ensuite les statistiques à variable appariée à l'aide de STAT LIST 2.

(Moyenne de x)

STAT
SHIFT 9 SHIFT LIST 2 EXE

$\Sigma x/n = 2.185714286$

(Moyenne de y)

EXE

$\Sigma y/n = 3.671428571$

(Moyenne des différences de x)

EXE

$x \text{ón} = 1.049392438$

(Moyenne des différences de y)

EXE

$y \text{ón} = 1.245235819$

(Moyenne des différences d'échantillon de x)

EXE

$x \text{ón}-1 = 1.133473381$

(Moyenne des différences d'échantillon de y)

EXE

$y \text{ón}-1 = 1.345008409$

(Constante de régression linéaire)

EXE

$a = 1.174221646$

(Coefficient de régression linéaire)

EXE

$b = 1.142512973$

(Coefficient de corrélation)

EXE

$r = 0.9628252383$

La valeur de r montre de manière évidente qu'il y a corrélation positive entre x et y . Ensuite, calculer le taux des frais de publicité qui permettrait d'amener le taux des bénéfices d'exploitation à 5,7% ? Calculer le taux des bénéfices d'exploitation si le taux de frais de publicité est de 4,5% ? Estimer chacune de ces valeurs.

(Valeur estimée de x)

EQX 5.7 EXE

3.961248986

(Valeur estimée de y)

EQY 4.5 EXE

6.315530022

Les résultats précédents montrent que le taux des frais de publicité nécessaires pour amener les bénéfices d'exploitation à 5,7% est de 3,96% et que si le taux des frais de publicité est de 4,5%, le taux des bénéfices d'exploitation sera de 6,32%.

Opération:

(Mémoriser les données statistiques)

2.1 3.4

2.1 3.4

(Entrer toutes les données séparées)

Sortir ensuite les statistiques à variable apparues

à partir du tableau.

(Moyenne de x)

(Moyenne de y)

(Moyenne des différences de x)

(Moyenne des différences de y)

(Moyenne des différences d'échantillon de x)

(Moyenne des différences d'échantillon de y)

(Constante de régression linéaire)

(Coefficient de régression linéaire)

(Coefficient de corrélation)

(Coefficient de corrélation)

(Coefficient de corrélation)

(Coefficient de corrélation)

(Coefficient de corrélation)

(Coefficient de corrélation)

(Coefficient de corrélation)

(Coefficient de corrélation)

(Coefficient de corrélation)

(Coefficient de corrélation)

(Coefficient de corrélation)

(Coefficient de corrélation)

(Coefficient de corrélation)

(Coefficient de corrélation)

(Coefficient de corrélation)

(Coefficient de corrélation)

(Coefficient de corrélation)

(Coefficient de corrélation)

(Coefficient de corrélation)

(Coefficient de corrélation)

(Coefficient de corrélation)

(Coefficient de corrélation)

(Coefficient de corrélation)

(Coefficient de corrélation)

(Coefficient de corrélation)

(Coefficient de corrélation)

(Coefficient de corrélation)

Utilisation de la fonction mémoire

Ce chapitre est consacré à l'étude de la "Fonction mémoire" qui est l'une des caractéristiques de cet ordinateur. Cette fonction simplifie considérablement les calculs d'équations dans lesquelles des valeurs numériques uniquement sont affectées à une variable.

3-1 Calculs avec une même équation

Cet ordinateur est pourvu d'une fonction particulièrement pratique appelée "Fonction mémoire". Cette fonction permet de procéder facilement à des calculs en affectant simplement des valeurs numériques à la variable à condition que l'équation ait été au préalable mise en mémoire.

Avec cette "Fonction mémoire", on utilise les touches suivantes:

IN Mise en mémoire du contenu présentement écrit.

OUT Affichage du contenu de la mémoire.

EQ Si le contenu de la mémoire est une équation, la valeur numérique souhaitée sera affectée aux variables utilisées et le résultat du calcul sera affiché.

L'exemple simple qui suit permettra d'apprendre à utiliser ces touches.

Exemple:

Obtenir la valeur de y pour chaque valeur affectée à x quand $y = 3,43 \cos x$. (Calculer au millièmè près.)

x	8°	15°	22°	27°	31°
y					

Opération:

Spécifier tout d'abord l'unité d'angle et la précision du calcul.

MODE **2** (Unité d'angle: "DEG")

SHIFT **2** **F** **3** **EXE** (Résultat à 3 chiffres après la virgule par arrondissement)

Entrer ensuite l'équation et appuyer sur la touche **IN** pour mettre en mémoire.

Y **=** **3** **.** **4** **3** **cos** **X** **IN**

Appuyer sur la touche **OUT** pour vérifier que l'équation est bien en mémoire.

OUT

$Y=3.43* \cos X$

Commencer enfin les calculs en appuyant sur la touche **EQ**.

3-1 Calculs avec une même équation

CALC

8 **EXE**

CALC

15 **EXE**

CALC

22 **EXE**

CALC

27 **EXE**

CALC

31 **EXE**

$X ?$
$Y= 3.397$
$X ?$
$Y= 3.313$
$X ?$
$Y= 3.180$
$X ?$
$Y= 3.056$
$X ?$
$Y= 2.940$

L'équation sera résolue successivement pour chaque valeur de x même si on appuie sur la touche **EXE** au lieu d'appuyer sur la touche **EQ**.

Comme le montre cet exemple, la "Fonction mémoire" est idéale pour les calculs d'une équation dans laquelle seules les valeurs numériques affectées à une variable diffèrent.

Exemple:

Obtenir la valeur de V pour les valeurs respectives de r quand $V = 4/3\pi r^3$. (Calculer au millièmè près.)

r	4.579	7.381	9.244	6.133	1.416
V					

Opération:

Spécifier tout d'abord la précision du calcul.

SHIFT **2** **F** **3** **EXE**

Entrer ensuite l'équation.

V **=** **4** **/** **3** ***** **pi** ***** **(R)** **³** **IN**

3-2 Utilisation pour la préparation de tableaux

Il est possible d'écrire de multiples équations en les séparant par le signe deux points (:). Cette méthode permet de préparer facilement des tableaux du type donné dans l'exemple ci-après.

Exemple:

Compléter le tableau suivant. (Effectuer les calculs au millième près.)

X	Y	$P = X \cdot Y$	$Q = X / Y$
4.27	1.17		
8.17	6.48		
6.07	9.47		
2.71	4.36		
1.98	3.62		

Opération:

(Spécification de précision des calculs)
 (Mise en mémoire des équations)
 (Début des calculs)
 4.27
 1.17
 EXE
 EXE
 :

X ?
 Y ?
 P = 4.996
 Q = 3.650
 X ?

Introduire ainsi les valeurs de X et de Y et les valeurs de P et Q seront successivement calculées pour fournir le tableau complété suivant.

X	Y	$P = X \cdot Y$	$Q = X / Y$
4.27	1.17	4.996	3.650
8.17	6.48	52.942	1.261
6.07	9.47	57.483	0.641
2.71	4.36	11.816	0.622
1.98	3.62	7.168	0.547

Il est possible d'utiliser jusqu'à sept caractères (combinaison de lettres majuscules et minuscules) pour le nom d'une variable. Ceci est particulièrement pratique car les noms des variables sont affichés comme des messages au moment de l'opération d'entrée et permettent ainsi de vérifier immédiatement à quelle variable est affectée la valeur introduite.

Exemple:

Compléter le tableau suivant. (Effectuer les calculs au centième près.)

Rayon (r)	Hauteur (h)	Volume d'un cylindre ($U = \pi r^2 h$)	Volume d'un cône ($V = 1/3 U$)
1.205	2.227		
2.174	3.451		
3.357	7.463		

Opération:

(Début des calculs)
 1.205
 2.227
 EXE
 EXE
 :

RADIUS ?
 HEIGHT ?
 CYLINDER = 10.16
 CONE = 3.39
 RADIUS ?

Si les valeurs du rayon (r) et de la hauteur (h) sont entrées de cette manière, le volume (U) du cylindre et le volume (V) du cône seront successivement calculés pour fournir le tableau suivant.

Rayon (r)	Hauteur (h)	Volume d'un cylindre ($U = \pi r^2 h$)	Volume d'un cône ($V = 1/3 U$)
1.205	2.227	10.16	3.39
2.174	3.451	51.24	17.08
3.357	7.463	264.22	88.07

La "Fonction mémoire" permet donc d'effectuer facilement des calculs répétitifs simples sans recourir au "Programme de calcul" qui fait l'objet du Chapitre 5.

4

Utilitaires incorporés

Cet ordinateur est doté de programmes utilitaires incorporés fort pratiques pour les calculs matriciels, les calculs de nombres complexes, les résolutions numériques d'équations, les intégrations numériques ainsi que les opérations binaires, décimales et hexadécimales.

4-1 Menu des utilitaires

Appuyer sur la touche **[F]** pour faire afficher le menu des utilitaires.

* Quel que soit le mode dans lequel était l'ordinateur, dès qu'on appuie sur la touche **[F]**, le mode RUN est systématiquement sélectionné.

Mat/Comp/Newt/Sim/HexBin

Pour choisir un programme utilitaire, appuyer sur la touche correspondante, à savoir:

- | | |
|-----------------------------|----------------------------------|
| [M] (Mat) | Calcul matriciel |
| [C] (Comp) | Calcul de nombres complexes |
| [N] (Newt) | Résolution numérique d'équations |
| [S] (Sim) | Intégration numérique |
| [H] (Hex) | Opération hexadécimale |
| [B] (Bin) | Opération binaire |

* Pour sortir du menu des utilitaires, appuyer sur la touche **[RUN]**.

4-2 Calcul matriciel

Dans le cadre du calcul matriciel, il est possible d'effectuer des additions, des soustractions et des multiplications, d'obtenir des produits scalaires, des déterminants, des matrices inverses, des matrices transposées et de résoudre des systèmes d'équations linéaires.

• Menu des opérations sur les matrices

Avant de commencer un calcul matriciel, choisir d'abord l'option Mat du menu des utilitaires en appuyant sur la touche **[M]**. Le menu initial des opérations sur les matrices (cf. Menu 1 ci-dessous) sera alors affiché.

Appuyer ensuite sur la touche **[F]** pour faire afficher le second menu des opérations sur les matrices (cf. Menu 2 ci-dessous). Pour repasser du menu 2 au menu 1, il suffit d'appuyer sur la touche **[EXE]** ou **[AR]**.

Menu 1

A/B/X/F/ 1:X→A / 2:A↔B

[F]

[AR]

OU **[EXE]**

Menu 2

Det/Inv/Tra/Equ/K/+/-/*

Pour choisir une opération de ces menus, appuyer sur la touche correspondante, à savoir:

- | | |
|--------------------------------|---|
| [A] (A) | Etablissement et vérification de la matrice A |
| [B] (B) | Etablissement et vérification de la matrice B |
| [X] (X) | Vérification de la valeur de la matrice X |
| [1] (1:X → A) | Transposition de la valeur de la matrice X à la matrice A |
| [2] (2:A ↔ B) | Permutation des matrices A et B |
| [F] (F) | Affichage du menu 2 des opérations sur les matrices |
| [D] (Det) | Calcul du déterminant de la matrice A |

[I] (Inv)	Calcul de la matrice inverse de la matrice A, représentée par la matrice X
[T] (Tra)	Calcul de la matrice transposée de la matrice A, représentée par la matrice X
[E] (Equ)	Résolution du système d'équations relatif aux matrices A et B, représenté par la matrice X
[K] (K)	Calcul du produit scalaire de la matrice A, représenté par la matrice X
[+] (+)	Somme de la matrice A et de la matrice B, représentée par la matrice X
[-] (-)	Différence entre la matrice A et la matrice B et représentation du résultat par la matrice X
[*] (*)	Produit des matrices A et B, représenté par la matrice X

• Etablissement des matrices

Pour établir une matrice, choisir d'abord la matrice A ou la matrice B dans le menu 1 des opérations sur les matrices; autrement dit, appuyer sur la touche **[A]** ou **[B]**. A titre d'exercice, suivre la procédure donnée en exemple ci-dessous.

Exemple:

Etablissement de la matrice A à 3 lignes et 4 colonnes représentée ci-contre.

		colonne d'indice j			
ligne d'indice i		1	0	3	4
		2	1	0	-1
		3	1	-2	-3

Opération:

		(menu des opérations sur matrice)			
[A]	(Choix de la matrice A)	A/B/X/F/ 1:X→A / 2:A←+B			
[3] [EXE]	(Nombre de lignes à entrer)	A(i,j)=(2,2) i?			
[4] [EXE]	(Nombre de colonnes à entrer)	A(i,j)=(3,2) j?			
		A(1,1)= 0?			

* Le type de la dernière matrice spécifiée reste en mémoire même si l'ordinateur est éteint (alimentation coupée).

La matrice A à 3 lignes et 4 colonnes étant maintenant établie, entrer ensuite les éléments des indices. Ces éléments sont numérotés de ① à ⑫ ci-contre, les entrer dans cet ordre.

1	0	3	4
①	②	③	④
2	1	0	-1
⑤	⑥	⑦	⑧
3	1	-2	-3
⑨	⑩	⑪	⑫

① **[EXE]**② **[EXE]**

A(1,1)= 0?
A(1,2)= 0?
A(1,3)= 0?

Continuer de la même façon pour entrer les éléments 3, 4, 2, 1, 0, -1, 3, 1, -2 et -3. Une fois que tous les éléments auront été spécifiés, le menu 1 des opérations sur les matrices sera de nouveau affiché.

A/B/X/F/ 1:X→A / 2:A←+B

Vérifier que la matrice a été correctement établie.

[A] (Choix de la matrice A)

[EXE]**[EXE]****[EXE]****[EXE]****[EXE]****[EXE]****[EXE]****[EXE]****[EXE]****[EXE]****[EXE]****[EXE]****[EXE]****[EXE]****[EXE]**

A(i,j)=(3,4) i?
A(i,j)=(3,4) j?
A(1,1)= 1?
A(1,2)= 0?
A(1,3)= 3?
A(1,4)= 4?
A(2,1)= 2?
A(2,2)= 1?
A(2,3)= 0?
A(2,4)= -1?
A(3,1)= 3?
A(3,2)= 1?
A(3,3)= -2?
A(3,4)= -3?
A/B/X/F/ 1:X→A / 2:A←+B

• Correction des données entrées

Si on trouve une erreur avant d'appuyer sur la touche $\boxed{\text{EXE}}$, il suffit d'entrer la valeur correcte et d'appuyer sur la touche $\boxed{\text{EXE}}$. Par contre, si on a déjà appuyé sur la touche $\boxed{\text{EXE}}$, il faut revenir à l'affichage précédent en appuyant sur la touche $\boxed{\text{AR}}$ pour pouvoir corriger l'erreur constatée.

• Somme, différence, produit de lignes

Exemple 1:

Calcul de $A + B$, $A - B$, $A * B$ et $B * A$ en posant pour hypothèse:

$$A = \begin{bmatrix} 1 & 1 \\ 2 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 2 & 3 \\ 2 & 1 \end{bmatrix}$$

Commencer par établir les matrices en choisissant la matrice A puis la matrice B dans le menu des opérations sur les matrices.

Opération:

(Menu des opérations sur matrices)

$\boxed{\text{A}}$ (Choix de la matrice A)

$\boxed{2} \boxed{\text{EXE}} \boxed{2} \boxed{\text{EXE}}$ (2 lignes et 2 colonnes)

$\boxed{1} \boxed{\text{EXE}} \boxed{1} \boxed{\text{EXE}} \boxed{2} \boxed{\text{EXE}} \boxed{1} \boxed{\text{EXE}}$ (Spécification des éléments)

$\boxed{\text{B}}$ (Choix de la matrice B)

$\boxed{2} \boxed{\text{EXE}} \boxed{2} \boxed{\text{EXE}}$ (2 lignes et 2 colonnes)

$\boxed{2} \boxed{\text{EXE}} \boxed{3} \boxed{\text{EXE}} \boxed{2} \boxed{\text{EXE}} \boxed{1} \boxed{\text{EXE}}$ (Spécification des éléments)

```

A/B/X/F/ 1:X-A / 2:A+B
A(i,j)=(2,2) i?
A(1,1)= 0?
A/B/X/F/ 1:X+A / 2:A+B
B(i,j)=(2,2) i?
B(1,1)= 0?
A/B/X/F/ 1:X+A / 2:A+B

```

Les deux matrices étant à présent établies, effectuer les calculs suivants.

$\boxed{\text{F}}$ (Affichage du menu 2)

$\boxed{+}$ (Somme)

```

Det/Inv/Tra/Equ/K/+/-/*
A/B/X/F/ 1:X+A / 2:A+B

```

La somme $A + B$ est calculée et le résultat est représenté par la matrice X.

$\boxed{\text{X}}$ (Vérification du résultat)

$\boxed{\text{EXE}}$

$\boxed{\text{EXE}}$

```

X(i,j)=(2,2)
X(1,1)= 3
X(1,2)= 4

```

$\boxed{\text{EXE}}$

$\boxed{\text{EXE}}$

$\boxed{\text{EXE}}$

```

X(2,1)= 4
X(2,2)= 2
A/B/X/F/ 1:X+A / 2:A+B

```

Dans cet exemple, le résultat de $A + B$ est $\begin{bmatrix} 3 & 4 \\ 4 & 2 \end{bmatrix}$.

$\boxed{\text{F}}$ (Affichage du menu 2)

$\boxed{-}$ (Différence)

```

Det/Inv/Tra/Equ/K/+/-/*
A/B/X/F/ 1:X+A / 2:A+B

```

La différence $A - B$ est calculée et le résultat est représenté par la matrice X.

$\boxed{\text{X}}$ (Vérification du résultat)

$\boxed{\text{EXE}}$

$\boxed{\text{EXE}}$

$\boxed{\text{EXE}}$

$\boxed{\text{EXE}}$

$\boxed{\text{EXE}}$

$\boxed{\text{EXE}}$

```

X(i,j)=(2,2)
X(1,1)=-1
X(1,2)=-2
X(2,1)= 0
X(2,2)= 0
A/B/X/F/ 1:X+A / 2:A+B

```

Dans cet exemple, le résultat de $A - B$ est $\begin{bmatrix} -1 & -2 \\ 0 & 0 \end{bmatrix}$.

$\boxed{\text{F}}$ (Affichage du menu 2)

$\boxed{*}$ (Produit)

```

Det/Inv/Tra/Equ/K/+/-/*
A/B/X/F/ 1:X+A / 2:A+B

```

Le produit $A * B$ est calculé et le résultat est représenté par la matrice X.

$\boxed{\text{X}}$ (Vérification du résultat)

$\boxed{\text{EXE}}$

$\boxed{\text{EXE}}$

$\boxed{\text{EXE}}$

$\boxed{\text{EXE}}$

$\boxed{\text{EXE}}$

$\boxed{\text{EXE}}$

```

X(i,j)=(2,2)
X(1,1)= 4
X(1,2)= 4
X(2,1)= 6
X(2,2)= 7
A/B/X/F/ 1:X+A / 2:A+B

```

Dans cet exemple, le résultat de $A * B$ est $\begin{bmatrix} 4 & 4 \\ 6 & 7 \end{bmatrix}$.

- [2] (Permutation des matrices A et B)
- [F] (Affichage du menu 2)
- [*] (Produit)

A/B/X/F/ 1:X+A / 2:A+B
 Det/Inv/Trn/Eq/K/+/-/*
 A/B/X/F/ 1:X+A / 2:A+B

Le produit $B * A$ est calculé et le résultat est représenté par la matrice X.

- [X] (Vérification du résultat)

EXE
 EXE
 EXE
 EXE
 EXE

X(i,j)=(2,2)
 X(1,1)= 8
 X(1,2)= 5
 X(2,1)= 4
 X(2,2)= 3
 A/B/X/F/ 1:X+A / 2:A+B

Dans cet exemple, le résultat de $B * A$ est $\begin{pmatrix} 8 & 5 \\ 4 & 3 \end{pmatrix}$.

Exemple 2:

Calcul du déterminant pour le système d'équations suivant:

$$\begin{pmatrix} 2 & -1 \\ 0 & -1 \\ 1 & 3 \end{pmatrix} \begin{pmatrix} 3 & -1 & 1 \\ 0 & 2 & 1 \end{pmatrix} + \begin{pmatrix} 1 & 0 & 1 \\ 2 & -3 & 0 \\ 0 & 0 & 2 \end{pmatrix}$$

Calculer d'abord le produit du premier terme; établir pour cela les matrices A et B définies ci-dessous et commander l'exécution de l'opération $A * B$.

$$A = \begin{pmatrix} 2 & 1 \\ 0 & -1 \\ 1 & 3 \end{pmatrix}, B = \begin{pmatrix} 3 & -1 & 1 \\ 0 & 2 & 1 \end{pmatrix}$$

Opération:

Suivre la procédure suivante une fois que le menu des opérations sur les matrices est affiché.

- [A][3][EXE][2][EXE] (Établissement de la matrice A)
- [2][EXE][1][EXE][2][EXE][+][1][EXE][3][EXE]
- [B][2][EXE][3][EXE] (Établissement de la matrice B)
- [3][EXE][+][1][EXE][1][EXE][2][EXE][2][EXE][1][EXE]
- [F][*] (Calcul de $A * B$)
- [1] (Transposition du résultat du calcul de $A * B$ à la matrice A)

A/B/X/F/ 1:X+A / 2:A+B
 A(1,1)= 0?
 A/B/X/F/ 1:X+A / 2:A+B
 B(1,1)= 0?
 A/B/X/F/ 1:X+A / 2:A+B
 A/B/X/F/ 1:X+A / 2:A+B
 A/B/X/F/ 1:X+A / 2:A+B

Calculer ensuite la somme des deux termes.

- [B] (Représentation de la matrice du second terme par la matrice B)
- [3][EXE][3][EXE] (3 lignes et 3 colonnes)
- [1][EXE][2][EXE][1][EXE][2][EXE][+][3][EXE][2][EXE][2][EXE]
- [2][EXE][2][EXE]
- [F][+]
- [X] (Vérification du résultat)

EXE
 EXE
 EXE
 EXE
 EXE
 EXE
 EXE
 EXE
 EXE
 EXE

B(i,j)=(2,3) i?
 B(1,1)= 0?
 A/B/X/F/ 1:X+A / 2:A+B
 A/B/X/F/ 1:X+A / 2:A+B
 X(i,j)=(3,3)
 X(1,1)= 7
 X(1,2)= 0
 X(1,3)= 4
 X(2,1)= 2
 X(2,2)= -5
 X(2,3)= -1
 X(3,1)= 3
 X(3,2)= 5
 X(3,3)= 6
 A/B/X/F/ 1:X+A / 2:A+B

Dans cet exemple, le résultat du calcul est:

$$\begin{pmatrix} 2 & 1 \\ 0 & -1 \\ 1 & 3 \end{pmatrix} \begin{pmatrix} 3 & -1 & 1 \\ 0 & 2 & 1 \end{pmatrix} + \begin{pmatrix} 1 & 0 & 1 \\ 2 & -3 & 0 \\ 0 & 0 & 2 \end{pmatrix} = \begin{pmatrix} 7 & 0 & 4 \\ 2 & -5 & -1 \\ 3 & 5 & 6 \end{pmatrix}$$

• Déterminant, matrice inverse, matrice transposée

Exemple 3:

Calcul du déterminant, de la matrice inverse et de la matrice transposée pour la matrice à 3 lignes et 3 colonnes définie ci-dessous.

$$\begin{bmatrix} 2 & 0 & 0 \\ 3 & 1 & 2 \\ 4 & 2 & 3 \end{bmatrix}$$

Opération:

(Menu 1 des opérations sur les matrices)

A **3** **EXE** **3** **EXE**

A/B/X/F/ 1:X+A / 2:A+B
A(1,1)= 0?

(Etablissement de la matrice A de l'exemple)

2 **EXE** **2** **EXE** **3** **EXE** **3** **EXE** **1** **EXE** **2** **EXE** **4** **EXE**

2 **EXE** **3** **EXE**

F (Affichage du menu 2 des opérations sur les matrices)

D (Calcul du déterminant)

EXE **F**

1

A/B/X/F/ 1:X+A / 2:A+B
Det/Inv/Tra/Equ/K/+/-/*
Det=-2
Det/Inv/Tra/Equ/K/+/-/*
A/B/X/F/ 1:X+A / 2:A+B

Calcul de la matrice inverse: le résultat est représenté par la matrice X.

X (Vérification du résultat)

EXE

EXE

EXE

EXE

EXE

EXE

EXE

EXE

EXE

EXE

X(i,j)=(3,3)
X(1,1)= 0.5
X(1,2)= 0
X(1,3)= 0
X(2,1)= 0.5
X(2,2)=-3
X(2,3)= 2
X(3,1)=-1
X(3,2)= 2
X(3,3)=-1
A/B/X/F/ 1:X+A / 2:A+B

Dans cet exemple, la matrice inverse (A^{-1}) de la matrice A est

$$\begin{bmatrix} 0.5 & 0 & 0 \\ 0.5 & -3 & 2 \\ -1 & 2 & -1 \end{bmatrix}$$

Calculer ensuite la matrice transposée.

F (Affichage du menu 2 des opérations sur les matrices)

T

Det/Inv/Tra/Equ/K/+/-/*
A/B/X/F/ 1:X+A / 2:A+B

Calcul de la matrice transposée: le résultat est représenté par la matrice X.

X (Vérification du résultat)

EXE

EXE

EXE

EXE

EXE

EXE

EXE

EXE

EXE

EXE

X(i,j)=(3,3)
X(1,1)= 2
X(1,2)= 3
X(1,3)= 4
X(2,1)= 0
X(2,2)= 1
X(2,3)= 2
X(3,1)= 0
X(3,2)= 2
X(3,3)= 3
A/B/X/F/ 1:X+A / 2:A+B

Dans cet exemple, la matrice transposée (A^T) de la matrice A est

$$\begin{bmatrix} 2 & 3 & 4 \\ 0 & 1 & 2 \\ 0 & 2 & 3 \end{bmatrix}$$

• Produit scalaire

Exemple 4:

Calcul du produit des matrices définies ci-dessous par un scalaire.

$$3 \begin{pmatrix} 1 & 2 \\ 2 & 1 \end{pmatrix} \begin{pmatrix} 3 & 1 \\ 0 & 2 \end{pmatrix}$$

Etablir les matrices A et B telles qu'elles sont définies ci-dessous et calculer le produit de la matrice B par le triple de la matrice A.

$$A = \begin{pmatrix} 1 & 2 \\ 2 & 1 \end{pmatrix}, B = \begin{pmatrix} 3 & 1 \\ 0 & 2 \end{pmatrix}$$

Opération:

(Menu 1 des opérations sur les matrices)

A **2** **EXE** **2** **EXE** (Établissement de la matrice A)

1 **EXE** **2** **EXE** **2** **EXE** **1** **EXE**

F (Affichage du menu 2 des opérations sur les matrices)

K (Sélection du produit scalaire)

3 **EXE**

A/B/X/F/ 1:X+A / 2:A+B
A(1,1)= 0?
A/B/X/F/ 1:X+A / 2:A+B
Det/Inv/Tra/Equ/K/+/-/*
k*A k?
A/B/X/F/ 1:X+A / 2:A+B

Multiplication de la matrice A par le scalaire 3; le résultat est représenté par la matrice X.

1 (Substitution de la matrice X par la matrice A)

B **2** **EXE** **2** **EXE** (Établissement de la matrice B)

3 **EXE** **1** **EXE** **2** **EXE** **2** **EXE**

F ***** (Calcul de A * B)

X (Vérification du résultat)

EXE

EXE

EXE

EXE

EXE

A/B/X/F/ 1:X+A / 2:A+B
B(1,1)= 0?
A/B/X/F/ 1:X+A / 2:A+B
A/B/X/F/ 1:X+A / 2:A+B
X(i,j)=(2,2)
X(1,1)= 9
X(1,2)= 15
X(2,1)= 18
X(2,2)= 12
A/B/X/F/ 1:X+A / 2:A+B

Dans cet exemple, le résultat du calcul est $3 \begin{pmatrix} 1 & 2 \\ 2 & 1 \end{pmatrix} \begin{pmatrix} 3 & 1 \\ 0 & 2 \end{pmatrix} = \begin{pmatrix} 9 & 15 \\ 18 & 12 \end{pmatrix}$.

• Système d'équations

Exemple 5:

Résolution du système d'équations linéaires suivant:

$$x_1 + 2x_2 + x_3 = 8$$

$$2x_1 - x_2 + 2x_3 = 6$$

$$2x_1 + 2x_2 - x_3 = 3$$

Pour résoudre ce système d'équations, affecter les constantes du premier membre de chaque équation à la matrice A et les constantes du second membre à la matrice B, à savoir:

$$A = \begin{pmatrix} 1 & 2 & 1 \\ 2 & -1 & 2 \\ 2 & 2 & -1 \end{pmatrix}, B = \begin{pmatrix} 8 \\ 6 \\ 3 \end{pmatrix}$$

Opération:

(Menu 1 des opérations sur les matrices)

A **3** **EXE** **3** **EXE** (Établissement de la matrice A)

1 **EXE** **2** **EXE** **1** **EXE** **2** **EXE** **1** **EXE** **2** **EXE**

2 **EXE** **2** **EXE** **-1** **EXE**

B **3** **EXE** **1** **EXE** (Établissement de la matrice B)

3 **EXE** **6** **EXE** **3** **EXE**

F

E

EXE

EXE

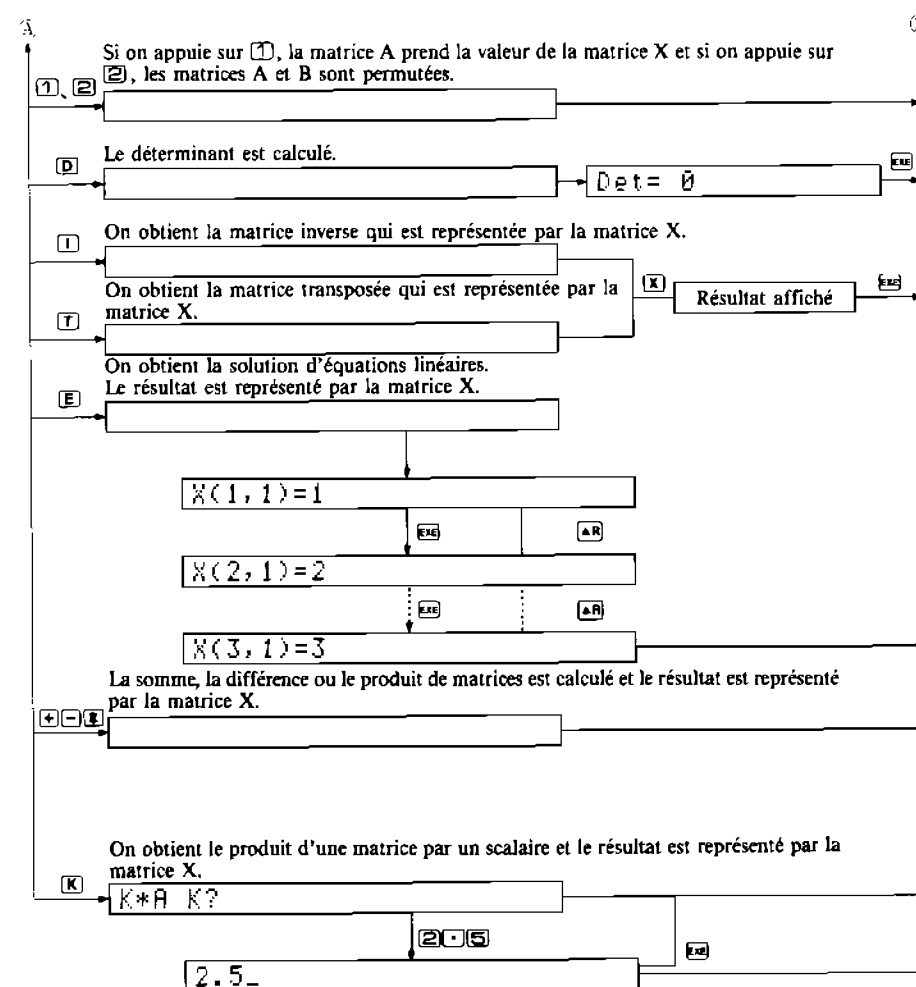
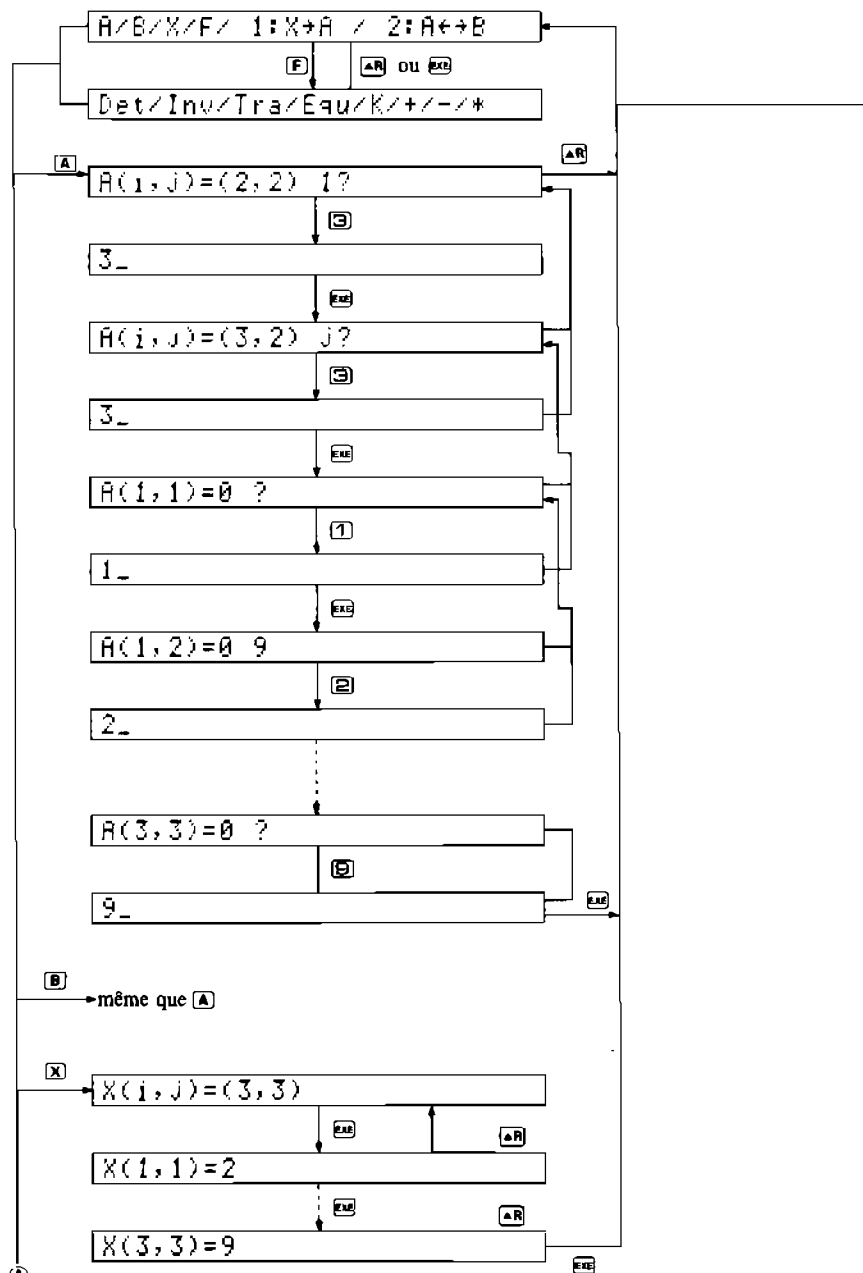
EXE

A/B/X/F/ 1:X+A / 2:A+B
A(1,1)= 0?
A/B/X/F/ 1:X+A / 2:A+B
B(1,1)= 0?
A/B/X/F/ 1:X+A / 2:A+B
Det/Inv/Tra/Equ/K/+/-/*
X(1,1)= 1
X(2,1)= 2
X(3,1)= 3
A/B/X/F/ 1:X+A / 2:A+B

D'après les valeurs affichées, on peut en déduire que les valeurs des inconnues sont:

$$x_1 = 1 \quad x_2 = 2 \quad x_3 = 3$$

• Sommaire des opérations sur les matrices



Notes:

- 1) A ou B est l'objet du calcul dans une opération sur une matrice et le résultat obtenu est représenté par la matrice X.
 - 2) Les conditions de calcul lors des opérations sur les matrices sont les suivantes (les indices de la matrice A étant notés (i, j) et ceux de la matrice B (m, n)).
 - a) Somme et différence
Le résultat est représenté par X à condition que $i = m$ et $j = n$.
 - b) Produit
Le résultat est représenté par X à condition que $i = m$.
 - c) Déterminant
Le résultat est représenté par X à condition que $i = j$.
 - d) Matrice inverse
Le résultat est représenté par X à condition que $i = j$ et que le déterminant soit différent de 0.
 - e) Système d'équations linéaires
Le résultat est représenté par X à condition que $i = j$, que le déterminant soit différent de 0 et que $j = m$.
- Si ces conditions ne sont pas remplies, le programme utilitaire décrète un état d'erreur FX.
- 3) Les opérations sur les matrices doivent être exécutées avec le plus grand soin car il peut arriver qu'un résultat précis ne puisse être obtenu du fait que le déterminant est proche de zéro ou en raison de l'arrondissement interne des valeurs.

4-3 Calcul de nombres complexes

Le calcul des nombre complexes englobe les opérations arithmétiques et permet d'obtenir des valeurs absolues, des arguments, des carrés, des cubes, des racines carrées et des racines cubiques.

Le calcul d'un nombre complexe commence par le choix de l'option Comp du menu des utilitaires; appuyer pour cela sur la touche $\boxed{\text{C}}$ et le message suivant sera affiché.

a+bi a?

Après avoir entré les valeurs de a et de b, l'on peut choisir parmi les opérations suivantes.

- $\boxed{+}$ Addition
- $\boxed{-}$ Soustraction
- $\boxed{\times}$ Multiplication
- $\boxed{\div}$ Division
- $\boxed{X}$ Calcul de la valeur absolue
- $\boxed{Y}$ Argument
- $\boxed{X^2}$ Elévation au carré
- $\boxed{X^3}$ Elévation au cube
- $\boxed{\sqrt{}}$ Extraction de la racine carrée
- $\boxed{\sqrt[3]{}}$ Extraction de la racine cubique
- $\boxed{\text{EAB}}$ Entrée de nouvelles données

• Opérations arithmétiques

Exemple 1:

Calcul des formules suivantes:

- ① $(2 + 3i) + (3 - 2i)$
- ② $4 - (3 + 2i)$
- ③ $(1 + 2i) \times (3 - 2i)$
- ④ $(2 - 3i) \div (1 + 2i)$

Opération:

- ①
- $\boxed{\text{C}} \boxed{\text{EXE}} \boxed{\text{C}} \boxed{\text{EXE}}$ (Entrée de $2 + 3i$)
- $\boxed{+}$ (Addition)

a+bi a?
2+3i
+(c+di)c?

3 **EXE** **-** **2** **EXE** (Entrée de $3 - 2i$)

5 + i

On obtient ainsi pour la formule ① : $(2 + 3i) + (3 - 2i) = 5 + i$.

②

EXE

a+bi a?

4 **EXE** **2** **EXE** (Entrée de 4)

4

- (Soustraction)

-(c+di) c?

3 **EXE** **2** **EXE** (Entrée de $3 + 2i$)

1 - 2i

On obtient ainsi pour la formule ② : $4 - (3 + 2i) = 1 - 2i$.

③

EXE

a+bi a?

1 **EXE** **2** **EXE** (Entrée de $1 + 2i$)

1 + 2i

***** (Multiplication)

***(c+di) c?**

3 **EXE** **-** **2** **EXE** (Entrée de $3 - 2i$)

7 + 4i

On obtient ainsi pour la formule ③ : $(1 + 2i) \times (3 - 2i) = 7 + 4i$.

④

EXE

a+bi a?

2 **EXE** **-** **3** **EXE** (Entrée de $2 - 3i$)

2 - 3i

/ (Division)

/(c+di) c?

1 **EXE** **2** **EXE** (Entrée de $1 + 2i$)

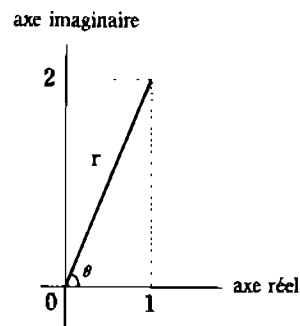
-0.8 - 1.4i

On obtient ainsi pour la formule ④ : $(2 - 3i) \div (1 + 2i) = -0,8 - 1,4i$ (le point sur l'affichage ayant valeur de virgule décimale).

• Valeur absolue et argument

Exemple 2:

Calcul de la valeur absolue (r) et de l'argument (θ) de $(1 + 2i)$.



Opération:

1 **EXE** **2** **EXE**

X (Calcul de la valeur absolue)

Y (Calcul de l'argument)

EXE

a+bi a?

1 + 2i

X(r) = 2.236067977

Y(arg) = 63.43494882

1 + 2i

On obtient ainsi la valeur absolue (r) 2,236067977 et l'argument (θ) 63,43494882 pour la formule $(1 + 2i)$.

• Carré, cube, racine carrée et racine cubique

Exemple 3:

Calcul des formules suivantes:

① $(2 + i)^2$

② $(2 - 3i)^3$

③ $\sqrt{-7 + 24i}$

④ $\sqrt[3]{-117 + 44i}$

Opération:

①

2 **EXE** **1** **EXE** (Entrée de $2 + i$)

x^2 (Élévation au carré)

a+bi a?

2 + i

3 + 4i

On obtient ainsi pour la formule ① : $(2 + i)^2 = 3 + 4i$.

②

EXE

2 **EXE** **-** **3** **EXE**

x^3 (Élévation au cube)

a+bi a?

2 - 3i

-46 - 9i

On obtient ainsi pour la formule ② : $(2 - 3i)^3 = -46 - 9i$.

③

[EXE]

[] [7] [EXE] [2] [4] [EXE]

[√] (Extraction de la racine carrée)

[EXE]

a+bi a?
-7+24i
3+4i
-3-4i

On obtient ainsi pour la formule ③ : $\sqrt{(-7+24i)} = 3+4i$ et $-3-4i$.

④

[EXE]

[] [1] [1] [7] [EXE] [4] [4] [EXE]

[√] (Extraction de la racine cubique)

[EXE]

[EXE]

a+bi a?
-117+44i
3+4i
-4.9641+0.598076i
1.9641-4.59808i

On obtient ainsi pour la formule ④ : $\sqrt[3]{(-117+44i)} = 3+4i$ et $-4,9641 + 0,598076i$ ainsi que $1,9641 - 4,59808i$.

• Sommaire des opérations sur les nombres complexes

Opérations arithmétiques

Lorsqu'un résultat est affiché, appuyer sur la touche [+], [-], [×] ou [÷] et entrer ensuite les valeurs de c et d.

Exemple:

2+3i
+
+(c+di) c?
[2] [EXE]
+(c+di) d?
[3] [EXE]
4+4i

* Si on appuie sur la touche [EXE] sans entrer aucune valeur numérique pour c et d, ce sont les valeurs précédemment entrées pour c et d qui seront choisies par défaut.

Entrée de nouvelles données

Lorsqu'un résultat est affiché, appuyer sur la touche [EXE] ; le résultat est remplacé par le message suivant et dès lors il est possible d'entrer de nouvelles valeurs.

a+bi a?

Valeur absolue (r) et argument (θ)

Lorsqu'un résultat est affiché, appuyer sur la touche [X] pour faire afficher la valeur absolue (r) et sur [Y] pour l'argument (θ). Le résultat est de nouveau affiché quand on appuie sur la touche [EXE] au moment où la valeur absolue (r) ou l'argument (θ) est affiché.

Exemple:

3+4i
[EXE]
[X]
X(r)= 5
[EXE]
[Y]
Y(arg)= 53.13010235
[EXE]

Carré et cube

Pour faire afficher le carré ou le cube, appuyer sur la touche [x²] ou [x³] au moment où un résultat est affiché.

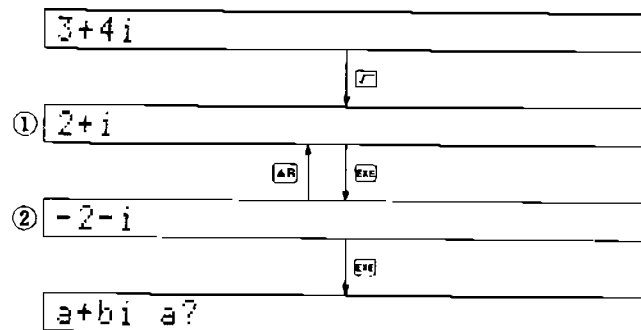
Exemple:

3+4i
[x²]
-7+24i

Racine carrée et racine cubique

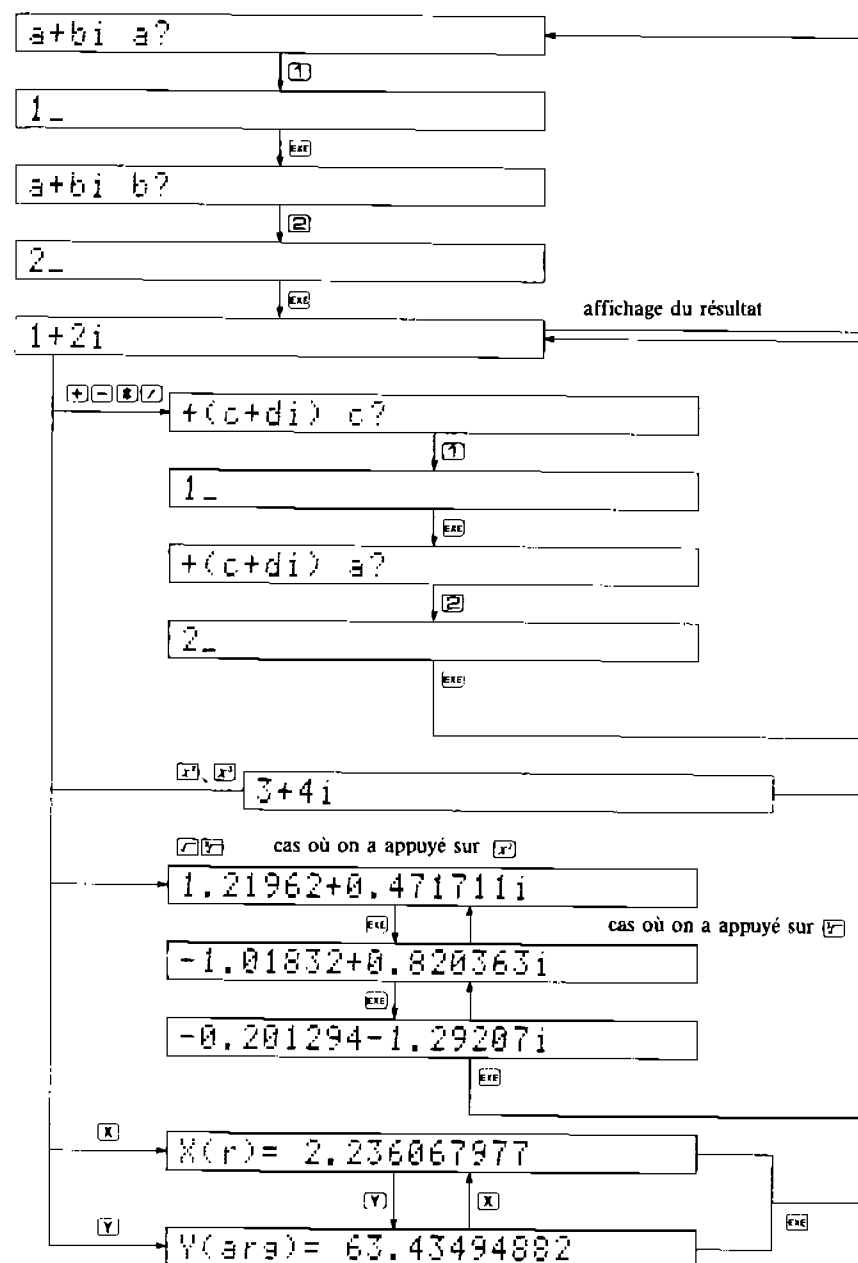
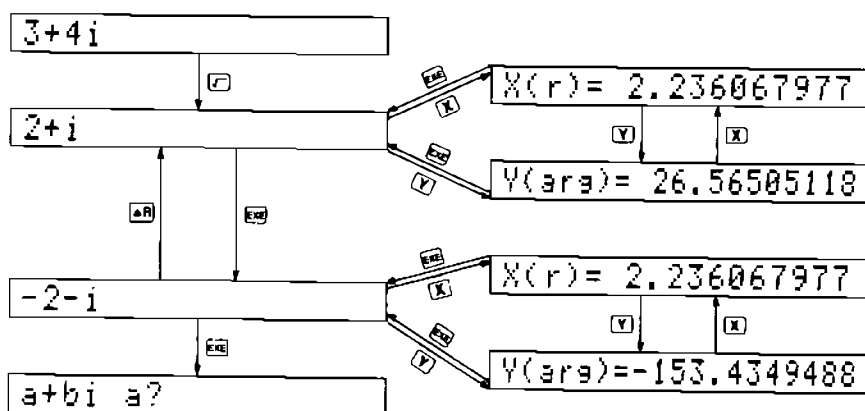
Pour faire afficher la racine carrée ou la racine cubique, appuyer sur la touche $\sqrt{}$ ou $\sqrt[3]{}$. Les différentes racines sont affichées successivement chaque fois que l'on appuie sur la touche EAC (2 racines carrées et 3 racines cubiques).

Exemple:



Il est possible d'effectuer les opérations arithmétiques x^2 , x^3 , r et θ aussi bien à partir de l'état ① que de l'état ②. D'autre part, il est également possible de vérifier la valeur absolue (r) et l'argument (θ) puis de revenir à l'affichage de la racine et de poursuivre l'opérations interrompue.

Exemple:



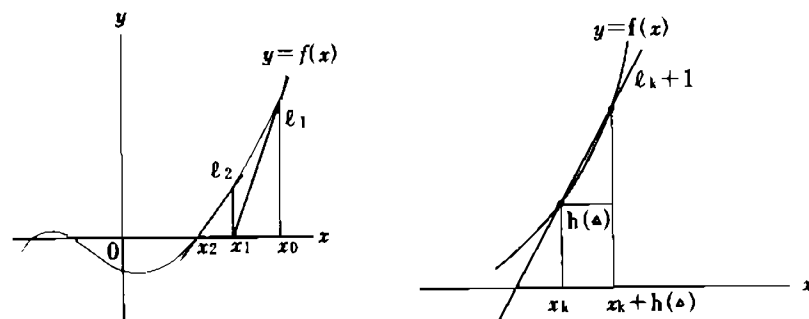
4-4 Résolution numérique d'équations

La méthode de résolution numérique d'une équation consiste à définir une suite d'approximations de x de plus en plus précises en partant d'une solution initiale (approchée) x_0 . (Voir le schéma ci-dessous.)

Autrement dit, la solution initiale x_0 étant donnée, obtenir la tangente l_1 à la courbe au point $(x_0, f(x_0))$ et poser x_1 au point d'intersection de la tangente l_1 et de l'axe x , puis opérer de la même façon pour obtenir la tangente l_2 en $(x_1, f(x_1))$ et réitérer ce procédé n fois jusqu'à obtenir une valeur x_n suffisamment approchée de x .

Le calcul prend fin et x_n est considéré comme la solution lorsque $|x_n - x_{n-1}| \leq$ erreur relative. Cependant, si la valeur est supérieure au nombre de réitérations défini pour n , l'équation ne sera pas résolue (cas rencontré lorsque la solution initiale choisie x_0 n'est pas suffisamment proche de x). Cette méthode de résolution est appelée "méthode de Newton" et est utilisée dans cet ordinateur pour la résolution numérique des équations.

Pour obtenir la tangente on définit la première dérivée (fonction dérivée) de la dérivation de $f(x)$ ainsi que l'intervalle minimum h . La droite passant par le point $(x_k, f(x_k))$ et le point $[(x_k + h(\Delta)), f(x_k + h(\Delta))]$ est considérée dans cet ordinateur comme la tangente.



• Menu de résolution numérique d'équations

Pour résoudre numériquement une équation, choisir d'abord l'option Newt du menu des utilitaires en appuyant sur la touche **N**. Le menu de résolution numérique d'équations, représenté ci-dessous, sera affiché.



A partir de ce menu, on peut choisir parmi les différentes opérations suivantes:

F (F(X))	Entrée et calcul de la fonction et de la valeur initiale
X (X)	Vérification du résultat
H (H(Δ))	Définition de l'intervalle minimum lors du calcul de la dérivée
E (Err)	Entrée de la plage d'erreur relative (Err) et du nombre maximum de réitérations (Maxloop)

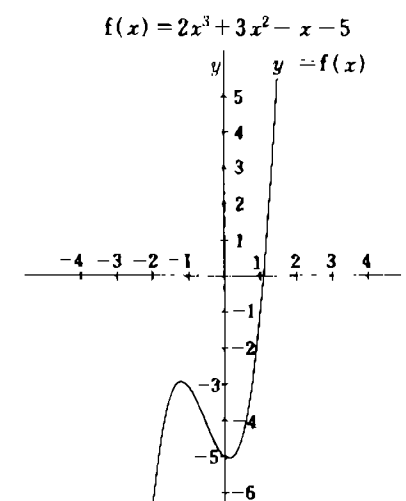
Il est possible d'utiliser les fonctions suivantes:

$+$, $-$, $*$, $/$, $\uparrow$, SIN, COS, TAN, ASN, ACS, ATN, LOG, LN, EXP, SQR et HYP. La variable de la fonction $f(x)$ doit être enregistrée sous le nom de X et la plage d'erreur relative doit être supérieure à $1E98$.

Les calculs internes étant effectués avec 12 chiffres, cela n'aurait pas de sens d'entrer une valeur extrêmement petite.

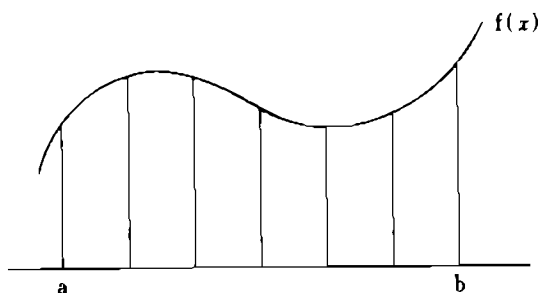
Exemple:

Résolution de l'équation $f(x) = 2x^3 + 3x^2 - x - 5 = 0$ avec une erreur relative de 0,0001 et un nombre maximum de réitérations de 30.



4-5 Intégration numérique

L'intégration numérique est une méthode de calcul par approximation de l'intégrale $\int_a^b f(x) dx$ d'une fonction donnée. (Voir le schéma ci-dessous.) Diviser l'intervalle (a, b) en n parties égales (n devant être un nombre pair) et en déduire l'aire à partir de la parabole approchée de la courbe de chaque subdivision. (méthode de Simpson)

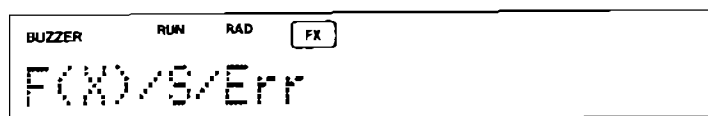


Cet ordinateur fixe automatiquement le nombre de subdivisions et le résultat est affiché lorsque l'erreur relative entre la valeur S_n (en cas de n subdivisions) et la valeur S_{n-2} (en cas de n-2 subdivisions) est comprise dans la plage d'erreur relative.

Etant donné que le résultat n'est affiché que lorsque l'erreur relative $Err > |S_n - S_{n-2}|$, si cette condition n'est pas satisfaite, même si n est 146, un message indiquant que le calcul est impossible sera affiché.

• Menu d'intégration numérique

Pour calculer numériquement une intégrale, choisir d'abord l'option Sim du menu des utilitaires en appuyant sur la touche $\boxed{S}$. Le menu d'intégration numérique, représenté ci-dessous, sera affiché.

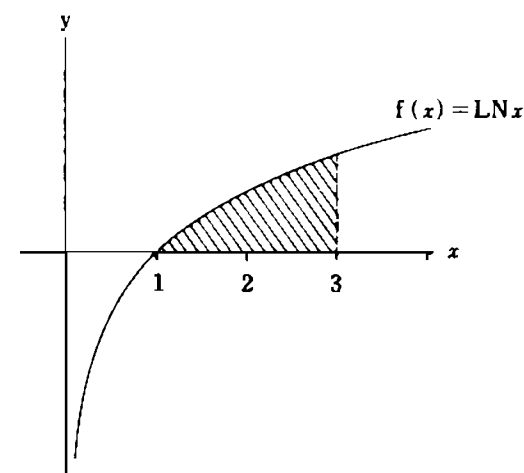


A partir de ce menu, on peut choisir parmi les différentes opérations suivantes:

- $\boxed{F}$ (F(X)) Entrée et calcul de la fonction et de l'intervalle a, b
- $\boxed{S}$ (S) Vérification de l'aire
- $\boxed{E}$ (Err) Entrée de la plage d'erreur relative

Exemple:

Calcul de l'aire de la partie hachurée sur le schéma ci-dessous, l'erreur relative étant 0,0001.



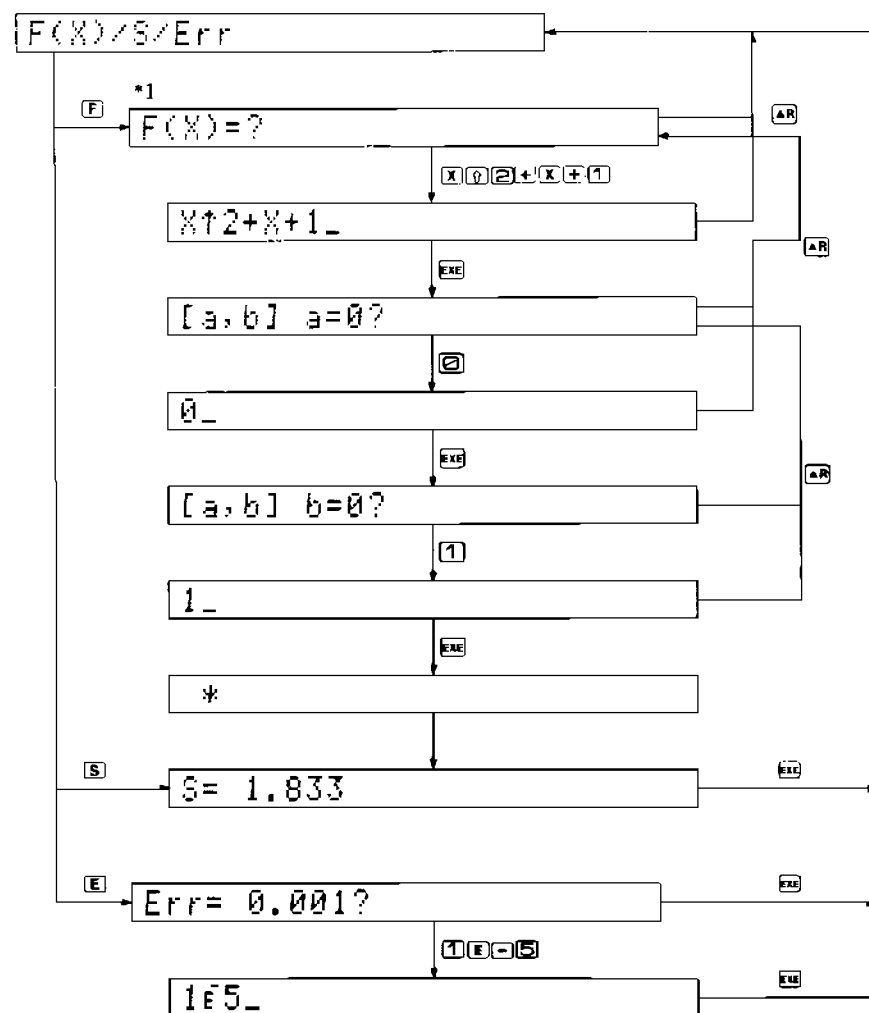
Opération:

- $\boxed{E}$ (Sélection de l'option erreur)
- $\boxed{0} \boxed{.} \boxed{0} \boxed{0} \boxed{0} \boxed{1} \boxed{ERR}$ (Entrée du chiffre d'erreur relative)
- $\boxed{F}$ (Sélection de l'option fonction)
- $\boxed{ln} \boxed{X} \boxed{EXE}$ (Entrée de la fonction f(x))
- $\boxed{1} \boxed{EXE} \boxed{3} \boxed{EXE}$ (Entrée de l'intervalle)

F(X)/S/Err
Error= 0.001?
F(X)/S/Err
F(X)=?
[a,b] a= 0?
*
S= 1.2958

Le résultat obtenu est $S = 1,2958$.

• Sommaire des opérations d'intégration numérique



*1 Si une équation a déjà été entrée pour F(x), cette équation sera affichée.

4-6 Opérations binaires, décimales et hexadécimales

Les opérations arithmétiques, les opérations booléennes, le calcul de complément à deux, le décalage logique ainsi que la conversion de système de numération peuvent être effectués en mode de calcul binaire, décimal ou hexadécimal.

Pour travailler en mode binaire, choisir l'option Bin du menu des utilitaires en appuyant sur la touche **[B]** et pour le mode hexadécimal, choisir l'option Hex en appuyant sur la touche **[H]**. Avec ce programme, il est possible d'effectuer des opérations sur des nombres entiers de 32 bits (éléments binaires) compris entre -2147483648 et 2147483647.

Dans les explications qui suivent, toutes les opérations sont en binaire.

(Choix de l'option binaire)

[B]

[1][0][0][0][EXE] (Entrée d'une valeur numérique)

[←] (Affichage des bits de poids supérieur)

Mat/Comp/Newt/Sim/HexBin
Bin?
+00000000000000001000(Lower)
(Upper)0000000000000000+

* Utiliser les touches de déplacement du curseur **[←]** et **[→]** pour faire afficher les bits de poids supérieur et de poids inférieur.

* Le fait d'entrer des valeurs autres que 0 ou 1 en mode de calcul binaire, des valeurs non comprises entre 0 et 9 ou A et F en mode de calcul hexadécimal ou des valeurs supérieures à 32 bits provoque l'annulation de la valeur entrée.

Après avoir entré une valeur numérique, on peut choisir l'une quelconque des opérations suivantes en appuyant sur la touche correspondante.

[+]	Addition
[-]	Soustraction
[*]	Multiplication
[/]	Division
[A] (AND)	Opération booléenne d'intersection (ET)
[O] (OR)	Opération booléenne d'union (OU)
[N] (NOT)	Opération booléenne de négation (NON)
[X] (XOR)	Opération booléenne de disjonction (OU exclusif)
[C]	Calcul de complément à deux

- R** Décalage logique à droite
- L** Décalage logique à gauche
- B** Conversion de numération décimale ou hexadécimale en numération binaire
- H** Conversion de numération binaire ou décimale en numération hexadécimale
- D** Conversion de numération binaire ou hexadécimale en numération décimale
- Affichage des bits de poids supérieur au cours d'une opération binaire
- Affichage des 16 bits de poids inférieur au cours d'une opération binaire
- EXE** Entrée d'une nouvelle valeur

• Opérations arithmétiques

Exemple:

Calcul de:

- ① $1011001_B + 1100_B$
- ② $100010_H - 1101_B$
- ③ $2A_{CH} \times 1B_H$
- ④ $F32D_H \div C_H$

(B représente les binaires et H les hexadécimaux.)

Opération:

①

B (Choix de mode binaire)	Bin?
1011001 EXE (Entrée de la valeur numérique)	+00000000001011001(Lower)
+ (Choix de l'addition)	00000000001011001 +?
1100 EXE	+00000000001100101(Lower)

On obtient ainsi pour ① $1011001_B + 1100_B = 1100101_B$.

②

EXE (Mode binaire pour nouvelle entrée)	Bin?
100010 EXE (Entrée de la valeur numérique)	+0000000000100010(Lower)
- (Choix de la soustraction)	0000000000100010 -?
1101 EXE	+000000000010101(Lower)

On obtient ainsi pour ② $100010_B - 1101_B = 10101_B$.

③

FX (Affichage du menu des utilitaires)	Mat/Comp/Newt/Sim/HexBin
H (Choix de mode hexadécimal)	Hex?
2A C EXE (Entrée de la valeur numérique)	(Hex) 000002AC
* (Choix de la multiplication)	000002AC *?
1B EXE	(Hex) 00004824

On obtient ainsi pour ③ $2A_{CH} \times 1B_H = 4824_H$.

④

EXE (Mode hexadécimal pour nouvelle entrée)	Hex?
F32D EXE (Entrée de la valeur numérique)	(Hex) 0000F32D
/ (Choix de la division)	0000F32D /?
C EXE	(Hex) 00001443

On obtient ainsi pour ④ $F32D_H \div C_H = 1443_H$.

• Opérations booléennes

Exemple:

Résolution des quatre problèmes suivants, sachant que A est égal à 110101B et B à 10111B.

- ① A OR B (union)
- ② A AND B (intersection)
- ③ A XOR B (disjonction)
- ④ NOT A (négation)

Opération:

①

FX	Mat/Comp/Newt/Sim/HexBin
B (Choix de mode binaire)	Bin?
110101 EXE (Entrée des valeurs numériques)	+0000000000110101(Lower)
O (Choix de OR)	0000000000110101 OR?
10111 EXE	+0000000000111111(Lower)

On obtient ainsi pour ① $A \text{ OR } B = 11111_B$.

②

EXE (Mode binaire pour nouvelle entrée)
1 1 0 1 0 1 EXE (Entrée de la valeur numérique)
A (Choix de AND)
1 0 1 1 1 1 EXE

Bin?
+0000000000110101(Lower)
0000000000110101 AND?
+0000000000100101(Lower)

On obtient ainsi pour ② $A \text{ AND } B = 100101B$.

③

EXE (Mode binaire pour nouvelle entrée)
1 1 0 1 0 1 EXE (Entrée de la valeur numérique)
X (Choix de XOR)
1 0 1 1 1 1 EXE

Bin?
+0000000000110101(Lower)
0000000000110101 XOR?
+000000000011010(Lower)

On obtient ainsi pour ③ $A \text{ XOR } B = 11010B$.

④

EXE (Mode binaire pour nouvelle entrée)
1 1 0 1 0 1 EXE (Entrée de la valeur numérique)
N (Choix de NOT)
← (Affichage des bits de poids supérieur)

Bin?
+0000000000110101(Lower)
+1111111111001010(Lower)
(Upper) 1111111111111111→

On obtient ainsi pour ④ $\text{NOT } A = 11111111111111111001010$.

• Complément et décalage logique

Exemple:

Résolution des problèmes suivants:

- ① Calculer le complément à deux de 11001010B
- ② Décaler 110000B d'un bit vers la gauche
- ③ Décaler 1FCH de deux bits vers la droite

Opération:

①

FX
B (Choix de mode binaire)
1 1 0 0 1 0 1 0 EXE (Entrée de la valeur numérique)

Mat/Comp/Newt/Sim/HexBin
Bin?
+0000000011001010(Lower)

- C** (Choix du complément à deux)
← (Affichage des bits de poids supérieur)

+1111111100110110(Lower)
(Upper) 1111111111111111→

Solution de ① : le complément à deux de 11001010B est 1111111111111111100110110.

②

- EXE** (Mode binaire pour nouvelle entrée)
1 1 0 0 0 0 EXE (Entrée de la valeur numérique)
L (Décalage à gauche)

Bin?
+0000000000110000(Lower)
+0000000000110000(Lower)

Solution de ② : en décalant 110000B d'un bit vers la gauche, on obtient 1100000B.

③

- FX** (Affichage du menu des utilitaires)
H (Choix du mode hexadécimal)
1 F C EXE (Entrée de la valeur numérique)
R R (Décalage à droite de 2 bits)

Mat/Comp/Newt/Sim/HexBin
Hex?
(Hex) 00001FC
(Hex) 0000007F

Solution de ③ : en décalant 1FCH de deux bits vers la droite, on obtient 7FH.

• Conversion de système de numération

Exemple:

Expression de l'hexadécimal AF3C en numération décimale et numération binaire.

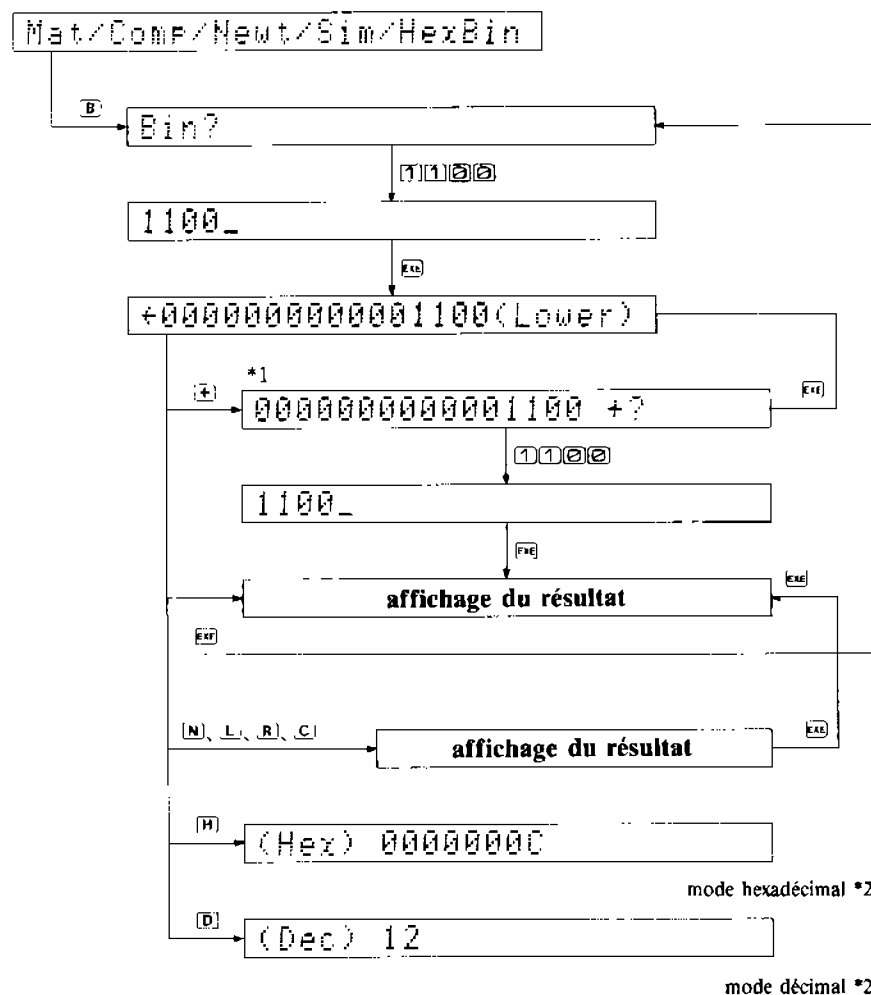
Opération:

- H** (Choix du mode hexadécimal)
A F 3 C EXE (Entrée de la valeur numérique)
D (Conversion en décimal)
B (Conversion en binaire)
← (Affichage des bits de poids supérieur)

Hex?
(Hex) 0000AF3C
(Dec) 44860
+1010111100111100(Lower)
(Upper) 0000000000000000→

L'hexadécimal AF3C est donc exprimé 44860 en numération décimale et 1010111100111100 en numération binaire.

• Sommaire des opérations binaires, décimales et hexadécimales



*1 La procédure est la même pour les opérations suivantes: [−] (soustraction), [×] (multiplication), [÷] (division), [A] (intersection) et [O] (union).

*2 Cette opération est possible aussi bien en mode décimal qu'en mode hexadécimal.

• Sommaire des opérations réalisables avec les utilitaires

FX	Touche	Fonction
Calcul matriciel	A	Etablissement ou vérification de la matrice A
	B	Etablissement ou vérification de la matrice B
	D	Calcul du déterminant de la matrice A
	E	Résolution du système d'équations $AX=B$
	F	Affichage du menu des opérations sur les matrices
	I	Calcul de la matrice inverse de la matrice A ($A^{-1} \rightarrow X$)
	K	Calcul du produit scalaire de la matrice A ($kA \rightarrow X$)
	T	Calcul de la matrice transposée de la matrice A ($A^T \rightarrow X$)
	X	Vérification de la matrice X (résultat)
	+	$A + B \rightarrow X$
	−	$A - B \rightarrow X$
	*	$A * B \rightarrow X$
	1	$X \rightarrow A$
	2	$A \leftrightarrow B$
Résolution numérique d'équations	F	Opération (entrée de fonction et de solution initiale)
	X	Solution
	H	Intervalle minimal
Intégration numérique	E	Erreur relative et nombre maximum de répétitions
	F	Opération (entrée de fonction et d'intervalle)
	S	Aire
	E	Erreur relative

* Ces touches permettent de choisir une option du menu affiché ce moment là

FX	Touche	Fonction
Calcul de nombres complexes	X	r (valeur absolue)
	Y	θ (argument)
	x^2	Carré
	x^3	Cube
	$\sqrt{}$	Racine carrée
	$\sqrt[3]{}$	Racine cubique
	+	Addition
	−	Soustraction
	*	Multiplication
	/	Division
Opérations binaires, décimales et hexadécimales	A	AND (intersection)
	B	Conversion en binaire
	C	Complément à deux
	D	Conversion en décimal
	H	Conversion en hexadécimal
	L	Décalage à gauche
	N	NOT (négation)
	O	OR (union)
	R	Décalage à droite
	X	XOR (disjonction)
	+	Addition
	−	Soustraction
	*	Multiplication
	/	Division

* Ces touches permettent de commander l'exécution des fonctions correspondantes

Notes:

- 1) Lorsque l'on utilise les programmes utilitaires, les variables autres que les variables en tableau définies par l'instruction DIM perdent toute signification. (Les tableaux spécifiés par l'instruction DEFNM sont également effacés.) Il importe également de procéder avec circonspection lors de la programmation en BASIC car les variables en tableau A (), B (), X (), D () et D\$ () sont également utilisées pour d'autres opérations, notamment lors du calcul matriciel.
- 2) On peut rencontrer les deux types d'erreur mentionnés ci-dessous quand on travaille sur les programmes utilitaires.
 Erreur FX: Erreur mathématique. Revenir à l'état initial en appuyant sur la touche **EXE**.
 Erreur de 1 à 9: Mêmes erreurs que pour le BASIC. Après avoir appuyé sur la touche **FX**, appuyer de nouveau sur la touche **BRK**.
- 3) En règle générale, on sort des programmes utilitaires en appuyant sur la touche **BRK**.
 On peut également quitter ces programmes en appuyant sur la touche **IN** ou **CLS** ou encore sur les touches **MODS** , **MODS**  ou **MODS**  lorsqu'un point d'interrogation (?) ou le curseur est affiché.
- 4) Ne pas utiliser la touche **STOP** quand on travaille sur un programme utilitaire.

Programmation en langage BASIC

Ce chapitre est consacré à une explication des programmes en langage BASIC. Le BASIC est un des langages de programmation et est actuellement utilisé dans pratiquement tous les miniordinateurs. Divers ouvrages consacrés au langage BASIC sont disponibles dans le commerce. Nous examinerons principalement ici les divers points spéciaux demandant une attention particulière pour effectuer une programmation en langage BASIC. Le détail concernant les commandes et la syntaxe en BASIC sera fourni en Chapitre 7 "Détail des commandes".

5-1 Ecrire les programmes

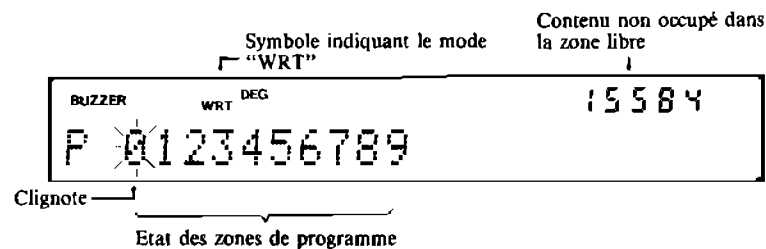
L'exemple ci-dessous est un programme en BASIC dans lequel la valeur de y est calculée à partir de l'équation $y = 2x^2 + 91x + 125$ en introduisant la valeur de x .

```
10 INPUT "x=";X
20 Y=2*X^2+91*X+125
30 PRINT "y=";Y
40 GOTO 10
50 END
```

Un programme en BASIC est un ensemble de "lignes", chaque ligne étant composée d'un nombre (tout nombre entier entre 1 et 9999) appelé numéro de ligne et d'instructions (des commandes telles que INPUT ou des équations telles que $Y=2*X^2+91*X+125$). Le programme est exécuté en ordre successif en commençant par le numéro de ligne le plus petit.

Voyons tout d'abord comment entrer le programme précédent.

Appuyer sur **MOD** **1**. Le symbole "WRT" doit apparaître à la partie supérieure de l'écran.

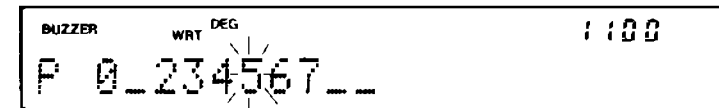


Le programme ne doit être écrit que lorsque l'ordinateur est en mode "WRT" comme illustré ci-dessous. Les caractères en majuscule "P 0123456789" indiquent l'état d'utilisation des zones de programme.

* L'ordinateur est pourvu de 10 zones de programme (P0 à P9) dans lesquelles de multiples programmes peuvent être mis en mémoire séparément. Pour changer de zone de programme, appuyer sur l'une des touches quelconque **⇐** à **⇐** après avoir appuyé sur **⇐**.

5-1 Ecrire les programmes

Le numéro clignotant indique la zone de programme actuellement spécifiée. Les chiffres affichés indiquent les zones ouvertes. Si un programme a été écrit, le chiffre est remplacé par un curseur (—). Sur l'affichage donnée ci-dessous à titre d'exemple, la zone spécifiée est P5 et on voit que les zones P1, P8 et P9 contiennent déjà des programmes.



Le nombre affiché à la partie supérieure droite de l'écran indique la capacité restante (nombre d'octets) de la zone dans laquelle des programmes ou des données peuvent être écrits (zone libre). La zone libre sera de 15584 octets quand rien n'est écrit et ces nombres diminueront au fur et à mesure de l'entrée de programmes ou de données.

Pour effacer tous les programmes entrer ce qui suit:

NEWALL

Cette opération efface les programmes dans toutes les zones de programme ainsi que les contenus des variables et prépare la zone de programme P0.

Ecrivons maintenant le programme mentionné plus haut.

```
10 INPUT "x=";X
20 Y=2*X^2+91*X+125
30 PRINT "y=";Y
40 GOTO 10
50 END
```

Ce programme sera mis en mémoire à l'aide des touches précédentes. **Ne pas oublier d'appuyer sur la touche **EXE** à la fin de chaque ligne** car la ligne ne peut être mise en mémoire que si cette touche **EXE** est enclenchée. Des touches spéciales existent pour certaines commandes d'usage fréquent comme INPUT et PRINT et permettent donc de travailler plus rapidement. (Par exemple, les cinq caractères de "INPUT" peuvent être écrits en appuyant sur **⇐** **X**. On peut aussi écrire **INPUT**.)

• Comment corriger une erreur de programmation

- 1) Si l'erreur est repérée avant d'appuyer sur **EXE** :
Amener le curseur au niveau de l'erreur à l'aide des touches (**←** ou **→**) et corriger l'erreur. (Comme en chapitre 2)
- 2) Si l'erreur est repérée après mise en mémoire à l'aide de la touche **EXE** :
Entrer LIST No. de ligne **EXE** en mode WRT pour afficher le contenu de la ligne et corriger comme en 1) ci-dessus.

Quel que soit le cas, toujours appuyer sur la touche **EXE après avoir corrigé car les corrections ne seront effectives que si cette touche est enclenchée.**

- * Appuyer sur **STOP** **EXE** pour afficher et corriger la ligne se trouvant immédiatement au-dessus de celle présentement affichée.
- * Dans le cas de 2), s'il existe des programmes supplémentaires après la ligne corrigée, la ligne suivante sera affichée quand **EXE** est enclenchée. Appuyer sur **END** pour ramener l'affichage à l'état initial du mode WRT.

• Addition et annulation de lignes

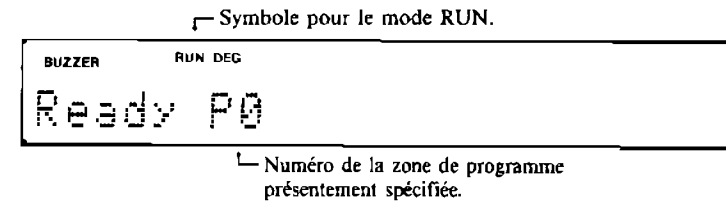
- 1) Pour ajouter une ligne il suffit d'entrer celle-ci après avoir libéré l'affichage en appuyant sur la touche **CLS** en mode WRT. Si une ligne est entrée avec un numéro de ligne déjà utilisé, cette ligne aura priorité et l'ancienne ligne sera effacée.
- 2) Pour effacer une ligne, entrer son numéro après avoir libéré l'affichage en appuyant sur **CLS** et appuyer sur **EXE**.

• Pour effacer les programmes

- 1) Pour effacer un programme dans la zone de programme présentement spécifiée, entrer NEW **EXE** en mode WRT.
 - 2) Pour effacer les programmes dans toutes les zones de programme (P0 à P9), entrer NEW ALL **EXE** en mode WRT.
- * Tout programme en mémoire le restera même si l'alimentation électrique est coupée.

5-2 Exécuter un programme

Pour exécuter un programme en mémoire, spécifier le mode RUN en appuyant sur **MODE** **EXE**. L'écran doit alors afficher ce qui suit:

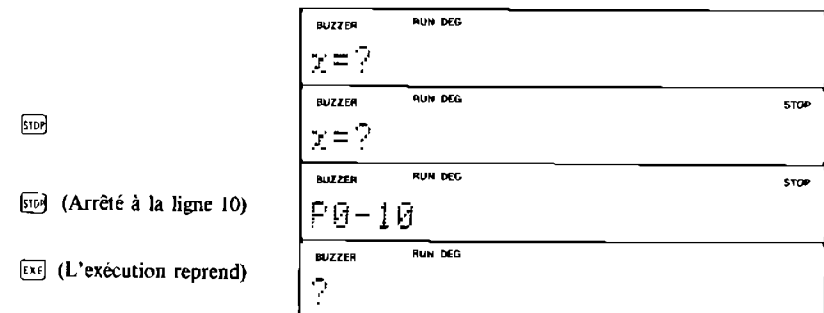


Le programme se trouvant dans la zone de programme présentement spécifiée est exécuté dès que l'ordre RUN est donné. Si le programme mis précédemment en mémoire est exécuté, le résultat sera le suivant.

MODE **EXE**
R **U** **N** **EXE**
EXE (Le programme commence)
2 (2 est entré pour x)
EXE (La valeur de y est affichée)
EXE (Saut à la ligne 10)

Ready P0
RUN_
x=?
2_
y= 315
x=?

Pour suspendre l'exécution d'un programme appuyer sur la touche **STOP**. Le mot "STOP" apparaît alors en haut et à droite de l'écran et l'exécution s'arrête. Si on appuie de nouveau sur la touche **STOP**, le numéro de la zone de programme présentement spécifiée et le numéro de la ligne où l'exécution du programme a été suspendue sont affichés. L'exécution du programme reprend dès que l'on appuie sur la touche **EXE**.



Pour terminer l'exécution d'un programme appuyer sur . L'affichage revient alors à l'état initial du mode RUN.



Ready P0

• **Autres méthodes d'exécution d'un programme**

1) RUN Numéro de ligne :

Cette opération permet de commencer un programme à une ligne donnée.

2) ~ :

Si on appuie sur une touche numérique après avoir appuyé sur la touche  en mode RUN, le programme se trouvant dans la zone spécifiée par la touche numérique sera exécuté en commençant par la première ligne.

* Les variables ne sont pas annulées par l'exécution d'un programme.

5-3 Variables

L'ordinateur est pourvu normalement de vingt-six variables, A à Z, permettant de mettre en mémoire des résultats de calculs et des valeurs numériques. Ces variables ont déjà été utilisées pour effectuer des calculs manuels et serviront de "mémoires" quand l'ordinateur sera utilisé comme calculateur.

On utilise les lettres alphabétiques A à Z pour désigner les variables qui sont dans ce cas dites variables simples. Il est également possible d'ajouter un chiffre entre parenthèses au nom de la variable comme par exemple A(3) et X(4,5). Les variables sous cette forme sont appelées variables de tableau et sont utilisées pour le maniement de volumes importants de données.

Toutes les variables mentionnées jusqu'à maintenant se sont toutes vues attribuer des valeurs numériques et sont pour cette raison appelées variables numériques. Les variables auxquelles sont affectées des chaînes de caractères sont dites variables caractère. Les variables caractère s'écrivent accompagnées du signe dollar (\$) à la suite du nom de la variable comme par exemple A\$. Les variables caractère peuvent également être utilisées comme variables de tableau comme par exemple C\$(12). En plus de ces variables, l'ordinateur est pourvu d'une variable caractère exclusive \$.

Pour résumer, l'ordinateur est pourvu des variables suivantes:

	Variable simple	Variable de tableau
Variable numérique	A, B, C, . . . , Z	A(0), X(2, 2) etc.
Variable caractère	A\$, B\$, . . . , Z\$, \$	A\$(0), X\$(2, 2) etc.

Une variable numérique peut stocker des valeurs numériques comprenant un maximum de 12 chiffres (12 chiffres pour une mantisse et 2 chiffres pour un exposant) et une variable caractère peut stocker une chaîne de caractères comprenant un maximum de 7 caractères. La variable caractère exclusive \$ peut stocker une chaîne de caractères d'un maximum de 62 caractères.

< Deux déclarations de tableau >

Cet ordinateur est capable de déclarations de variable de tableau par instructions DIM et de déclarations de tableau par instructions DEFM. Avec une instruction DIM il est possible d'utiliser des variables de tableau jusqu'à trois dimensions. On utilise l'instruction DEFM pour augmenter le nombre de variables mais on peut également l'utiliser pour déclarer des variables de tableau. Avec une instruction DIM on peut simultanément utiliser jusqu'à huit variables de tableau.

< Mode DIM et mode DEFM >

Il n'est pas possible de spécifier un tableau avec une instruction DIM et une instruction DEFM en même temps. En d'autres termes, l'ordinateur choisit soit le mode DIM soit le mode DEFM et affiche le mode sélectionné, DEFM en mode DEFM par exemple. Quand l'ordinateur est mis sous tension, il revient à l'état qui était le sien avant que l'alimentation soit coupée. Si une instruction DIM était exécutée, l'ordinateur revient en mode DIM et si une instruction DEFM était exécutée, il revient en mode DEFM.

< Précautions à la commutation >

Certaines précautions doivent être prises pour la commutation de mode car le tableau déclaré avec DIM sera effacé par la commutation du mode DIM en mode DEFM, et de même, le tableau déclaré en mode DEFM sera effacé par la commutation du mode DEFM au mode DIM.

< Déclaration de tableau avec une instruction DIM >

Cet ordinateur est capable de déclarer des variables de tableau jusqu'à trois dimensions. Ces variables de tableau consistent en deux types qui sont les variables de tableau caractères et les variables de tableau numériques.

1) Déclaration de variables de tableau à une dimension

On utilise une instruction DIM pour déclarer les variables de tableau et une variable de tableau à une dimension est spécifiée comme suit:

`DIM A ( 10 )` * Le nombre 10 entre parenthèses est appelé indice.

Ce tableau comprend 11 éléments A(0), A(1), A(2) ... A(10). Ce éléments de tableau existent de manière indépendante par rapport aux variables du type A, B, C etc... qui sont des variables numériques. Pour effacer ces tableaux on utilisera donc une instruction CLEAR ou ERASE. Pour déclarer deux variables de tableau, les séparer par une virgule.

`DIM A ( 3 ) , B ( 3 )`

Il est alors possible d'utiliser des variables de tableau du type A(0), A(1), A(2), A(3), B(0), B(1), B(2), B(3).

Pour les variables de tableau caractères, ajouter le symbole \$ après le nom de la variable.

`DIM A$ ( 4 )`

Les éléments de tableau A\$(0), A\$(1), A\$(4) sont alors déclarés variables de tableau caractères.

2) Déclaration de variables de tableau à deux dimensions

Pour déclarer les variables de tableau à deux dimensions avec une instruction DIM, séparer les indices par une virgule.

`DIM A ( 2 , 3 )`

La variable de tableau est alors déclarée à deux dimensions.

Comme le montre le tableau il y a 12 éléments de tableau.

		y			
x	A(0, 0)	A(0, 1)	A(0, 2)	A(0, 3)	
	A(1, 0)	A(1, 1)	A(1, 2)	A(1, 3)	
	A(2, 0)	A(2, 1)	A(2, 2)	A(2, 3)	

Pour les variables de tableau caractères, ajouter le symbole \$ après le nom de la variable. Comme dans le cas des variables de tableau à une dimension, des variables de tableau multiples peuvent être déclarées en même temps par séparation par des virgules.

3) Déclaration de variables de tableau à trois dimensions

Comme pour la déclaration des variables de tableau à deux dimensions, séparer les indices par des virgules.

```
DIM A$(1,2,3)
```

La déclaration de tableau est à trois dimensions avec une variable de tableau caractère A\$. Il y a 24 éléments de tableau ($2 \times 3 \times 4$) et comme ils sont indépendants de la variable de caractère A\$, le contenu de la variable de tableau reste inchangé quelle que soit la substitution à A\$. Comme pour les variables de tableau à une dimension, il est possible de déclarer en même temps des variables de tableau multiples par séparation par des virgules.

4) Gamme effective des variables de tableau

Un tableau peut être utilisé dans de multiples programmes puisque les variables de tableau sont communes à toutes les zones de programme de 0 à 9. Chaque élément de tableau nécessite huit octets et il faut par conséquent prendre certaines précautions quant à la capacité mémoire disponible avant de déclarer différentes variables de tableau dans chaque programme.

Pour la méthode d'utilisation des variables de tableau avec une instruction DIM, voir en Chapitre 6 "Bibliothèque de programmes" et en Chapitre 7 "Détail des commandes".

< Extension de variable >

Bien que 26 variables uniquement (A à Z) soient fournies normalement sur cet ordinateur, des variables supplémentaires sont possibles. Le nombre des variables peut être étendu à l'aide de la commande DEFM et en spécifiant avec le format "DEFM Nombre supplémentaire (expression numérique)".

Exemple:

Pour ajouter 20 variables et donc disposer de 46 variables en tout, procéder comme indiqué ci-dessous en mode RUN ou WRT.

```
DEFM20
```

```

BLUZZER  RUN DEG  DEFM
A * Z:26  DEFM:20
  
```

En opération manuelle, le nombre des variables sera affiché comme il apparaît ci-dessus.

Le nombre maximum de variables est de 1974.

La capacité restante de la zone libre diminuera de 8 octets pour chaque variable supplémentaire, donc attention à conserver un espace suffisant pour créer le programme. Le tableau ci-dessous donne le rapport entre le nombre des variables et la capacité maximum de la zone libre.

Nombre supplémentaire	Nombre de variables	Zone libre
0	26	15584
1	27	15576
2	28	15568
⋮	⋮	⋮
171	197	14216
172	198	14208
173	199	14200
⋮	⋮	⋮
939	965	8072
940	966	8064
941	967	8056
⋮	⋮	⋮
1195	1221	6024
1196	1222	6016
1197	1223	6008
⋮	⋮	⋮
1947	1973	8
1948	1974	0

↑
Capacité avec rien en mémoire

Les nouvelles variables supplémentaires sont utilisées comme variables de tableau à la suite de la variable Z.

La relation qui existe entre chaque variable est indiquée plus loin.

```

A(26) = B(25) = ..... = Y(2) = Z(1)
A(27) = B(26) = ..... = Y(3) = Z(2)
⋮      ⋮      ⋮      ⋮      ⋮
A(1973) = B(1972) = ..... = Y(1949) = Z(1948)
  
```

La commande DEFM peut non seulement être utilisée en opération manuelle mais également dans un programme. Par exemple, pour utiliser les variables de tableau Z(0) à Z(20) dans un programme et affecter une valeur numérique J à Z(J), le programme peut être créé comme suit:

Exemple:

```

10 DEFM 20      ... 20 variables supplémentaires
20 FOR J=0 TO 20 }
30 Z(J)=J       } Les valeurs numériques sont affectées
40 NEXT J        } aux variables de tableau
                  :

```

* Si la commande DEFM est utilisée dans un programme, le nombre des variables ne sera pas affiché pendant l'exécution.

La commande DEFM est également utilisée pour afficher le nombre présent de variables. DEFM ne sera exécuté que dans ce cas.

DEFM

RUN DEFM
A..Z:26 DEFM:20

* Quand la seule instruction 'DEFM' est utilisée dans un programme, le nombre de variables sera affiché à l'exécution du programme.

Notes:

1) Si le nombre de variables a été augmenté, cette spécification sera mise en mémoire même si l'alimentation électrique est coupée. Pour revenir aux 26 variables normales, exécuter DEFM 0.

* Le nombre des variables revient au nombre standard de 26 quand NEW ALL [EXE] est effectué en mode WRT ou quand une commutation est effectuée vers le mode DIM.

DEFM0

RUN DEFM0
A..Z:26 DEFM:0

2) Si le nombre d'octets utilisé pour augmenter le nombre des variables est spécifié en excès de la zone libre restante, une erreur MEMORY OVER (mémoire épuisée) (Erreur 1) se produira pour protéger les programmes et les données en mémoire.

<Précautions d'utilisation des variables>

1) Si les noms d'une variable numérique et d'une variable caractère sont les mêmes, le même espace mémoire est utilisé. Pour cette raison il n'est pas possible d'utiliser en même temps une variable numérique A et une variable caractère A\$. Si le programme est exécuté, une erreur (Erreur 6) va se produire à la ligne 20. (Pour le détail voir page 249)

```

10 A$="CASIO"..... Affecte la chaîne de caractères "CASIO" à
                        la variable caractère A$.
20 PRINT A           ..... Affiche le contenu de la variable numérique
30 END               A.

```

* A l'exclusion des variables de tableau en mode DIM

2) Attention à l'utilisation des variables de tableau en mode DEFM car l'espace occupé en mémoire est le même que les variables simples. Dans l'exemple ci-dessous, les variables combinées avec le signe égal (=) occupent le même espace mémoire. (Bien qu'ici il s'agisse de variables numériques, ces relations sont les mêmes que pour les variables caractère.)

```

A = A(0)
B = A(1) = B(0)
C = A(2) = B(1) = C(0)
D = A(3) = B(2) = C(1) = D(0)
:   :       :       :       :
Y = A(24) = B(23) = C(22) = ..... = Y(0)
Z = A(25) = B(24) = C(23) = ..... = Y(1) = Z(0)

```

↑
Variables simples Variables de tableau

Si, par exemple, nous exécutons le programme suivant, le contenu de la variable C sera 10.


```
10 DEFM
20 C=0
30 A(2)=10
40 PRINT C
50 END
```

- 3) Etant donnée que les variables sont communes dans tous les programmes, une attention particulière doit être portée à l'affectation des variables quand des programmes sont créés à l'aide de zones de programme multiples.

5-4 Méthode de calcul de la longueur d'un programme

La capacité maximum de la zone libre utilisée pour les programmes et DATA BANK est de 15584 octets. Cette zone libre diminue au fur et à mesure que programmes et données sont écrits. (Le nombre indiqué en haut de l'écran en mode WRT est le nombre d'octets restant.)

Le nombre d'octets nécessaire pour écrire programmes et données se calcule de la manière suivante.

- **Numéro de ligne** Deux octets pour un numéro pour tout nombre entre 1 et 9999.
- **Commande** Un octet pour une commande.
- **Fonction** Un octet pour une fonction.
- **Caractère** Un octet pour un caractère (un espace étant considéré comme un caractère).
- **Touche** EXE Un octet est nécessaire quand cette touche EXE est enclenchée en fin de ligne.
- Si une variable de tableau est déclarée avec une instruction DIM, huit octets sont nécessaires par élément de tableau.
- Si le nombre des variables est augmenté avec une instruction DEFM, huit octets sont nécessaires pour chaque variable supplémentaire.

Lorsqu'on utilise de nombreuses variables de tableau ou pour écrire de longs programmes, il est nécessaire d'évaluer la longueur du programme en fonction des indications précédentes et de supprimer toutes les parties inutiles.

Voici deux techniques qu'il est utile de connaître pour effectuer une programmation.

• Utilisation des zones de programme

Cette unité est pourvue de 10 zones de programme (P0 à P9) dans lesquelles il est possible de mettre des programmes en mémoire. Une des manières d'utilisation de ces zones est de placer le programme principal dans une zone de programme et le sous-programme dans une autre. Un exemple simple est donnée ci-dessous.

Exemple:

Zone de programme P0 (Programme principal)

```
10 X=12
20 PRINT
30 PRINT CSRX;"@";
40 GOSUB #1
50 GOTO 20
60 END
```

Zone de programme P1 (Sous-programme)

```
10 K$=KEY$
20 IF K$="4" THEN X=X-1: IF X<0 THEN
  X=0
30 IF K$="6" THEN X=X+1: IF X>23 THEN
  X=23
40 RETURN
```

Quand le programme principal en P0 est exécuté, @ (repère du prix unitaire) est indiqué au milieu de l'écran. Appuyer sur la touche [4] pour déplacer @ vers la gauche et sur la touche [6] pour déplacer @ vers la droite. @ ne peut être déplacé que par ces seules touches.

Dans ce programme, le programme principal en P0 affiche @ et le sous-programme en P1 calcule la position de déplacement de @ par entrée sur clavier. L'accès au sous-programme en P1 se fait par GOSUB #1 en ligne 40 du programme principal.

Cette méthode qui permet d'affecter des programmes avec des fonctions cohérentes dans une zone de programme facilite l'emploi des programmes longs.

• Utilisation de tableaux en mode DIM

Dans cet exemple, deux types de données sont traités à l'aide de variables de tableau à deux dimensions.

Exemple:

Mettre en mémoire le nom (NAME) et la taille (HEIGHT) de 15 personnes.

```
P0 10 DIM A$(1,14)
    20 FOR C=0 TO 14
    30 INPUT "NAME",A$(0,C)
    40 INPUT "HEIGHT",A$(1,C)
    50 NEXT C
    60 END

P1 10 INPUT "NAME",N$
    20 FOR C=0 TO 14
    30 IF N$=A$(0,C) THEN PRINT
      A$(1,C);"cm":GOTO 10
    40 NEXT C
    50 PRINT "NO NAME"
    60 GOTO 10
```

Entrer les noms et les tailles dans P0 puisque c'est le programme d'entrée. P1 est le programme permettant d'afficher la taille d'une personne quand son nom est entré.

La variable C est utilisée pour commander la boucle FOR ~ NEXT et la variable N\$ pour la mise en mémoire provisoire pendant la recherche d'un nom.

Le tableau A\$(0,0) ~ A\$(0,14) est une variable de tableau qui mémorise les noms des 15 personnes et le tableau A\$(1,0) ~ A\$(1,14) est une variable de tableau qui mémorise la taille de ces 15 personnes.

* A la ligne 30 de P1, quand le nom de la personne entré est concerné, sa taille est affichée. L'exécution passe alors à la ligne 10. Pour entrer un autre nom, appuyer sur la touche **ENT** quand la taille de la personne précédente est affichée.

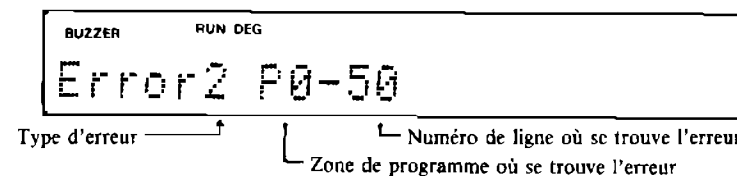
5-6 Messages erreur et mise au point

Une faute dans un programme est appelée "erreur" et l'opération de repérage et de correction de cette erreur est dite "mise au point".

La mise au point peut être effectuée facilement dans l'ordinateur car celui-ci est pourvu d'un mécanisme de contrôle automatique qui affiche un message erreur quand une erreur existe dans l'exécution d'un programme ou dans la syntaxe du langage BASIC. Il est particulièrement important de rechercher les erreurs car parfois une erreur n'est pas détectée en tant que telle mais empêche l'obtention de résultats.

1. Mise au point avec un message erreur

Quand l'affichage est comme ci-dessous, le message erreur indique le type d'erreur, la zone de programme et le numéro de ligne où se trouve l'erreur.



L'Erreur 2 affichée ici est dite "code d'erreur" et indique le type de l'erreur. Pour les codes d'erreur, se reporter aux "Messages d'erreur" en page 248.

Si une erreur se produit, spécifier le mode WRT en appuyant sur **WR** **ENT** après avoir libéré l'erreur à l'aide de la touche **DEL** et corriger en appelant la ligne erronée à l'aide de la commande LIST.

2. Mise au point quand l'erreur n'est pas affichée

Si le résultat souhaité ne peut pas être obtenu sans affichage d'un message erreur, il y a un "bug" (erreur) quelque part dans le programme qui ne devient pas une erreur. Dans ce cas, la mise au point est effectuée pendant l'exécution du programme.

- **Mise au point avec la commande STOP**

L'exécution du programme sera suspendue si la commande STOP est inscrite immédiatement après la ligne contenant la variable à contrôler, donc afficher et contrôler le contenu de la variable en entrant la variable à contrôler suivi de **END**. Pour reprendre l'exécution du programme appuyer sur **END**.

- **Mise au point en mode TRACE**

Appuyer sur **MODE** **2** pour afficher "TRACE ON". Ce mode est appelé le mode TRACE. Si un programme est exécuté en mode TRACE, une pause sera effectuée après chaque ligne (après chaque instruction si on utilise des multi-instructions). Appuyer sur **END** pour avancer à l'exécution suivante.

Le mode TRACE est utilisé pour trouver une erreur en défilant le programme. "TRACE ON" disparaît de l'écran et le mode TRACE est libéré quand on appuie sur **MODE** **3**.

Les autres causes d'erreur peuvent être considérées comme des "erreurs dans la variable" et des "erreurs dans l'indice d'une variable de tableau". Il faut alors vérifier consciencieusement le programme.

5-7 Périphériques utiles

Bien que cet ordinateur se suffise à lui-même, plusieurs périphériques sont disponibles en option.

Imprimante de caractères avec interface cassette (FP-40)

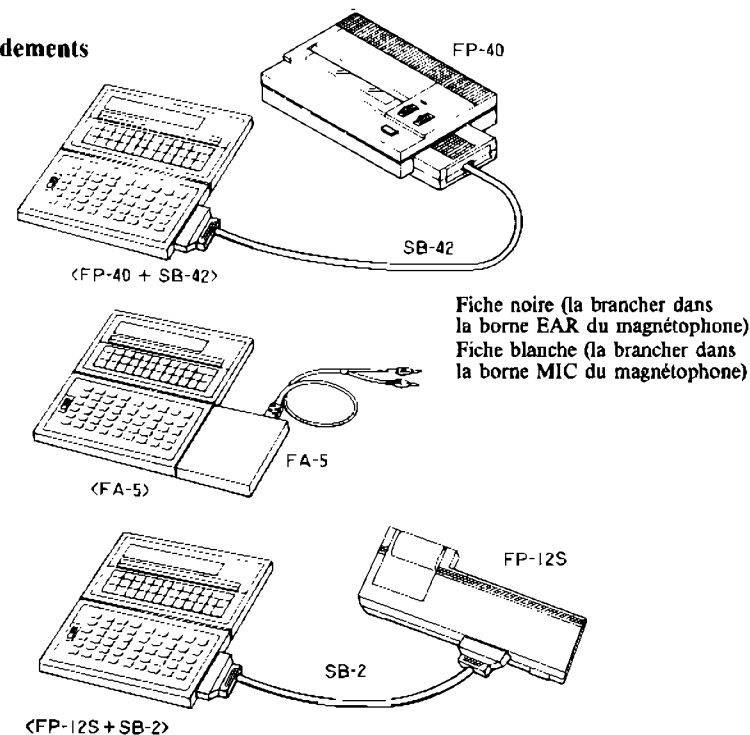
Interface cassette (FA-5)

Imprimante de caractères (FP-12S)

Module d'interface (SB-42)

Les interfaces cassette permettent de charger rapidement sur des cassettes des programmes se trouvant dans l'ordinateur et inversement. Il est également possible de stocker des données dans les variables et dans la DATA BANK. Les imprimantes de caractères impriment le contenu des programmes, les données et les résultats des calculs.

- **Raccordements**



* Avant de raccorder l'ordinateur à l'un quelconque des périphériques, couper l'alimentation électrique.

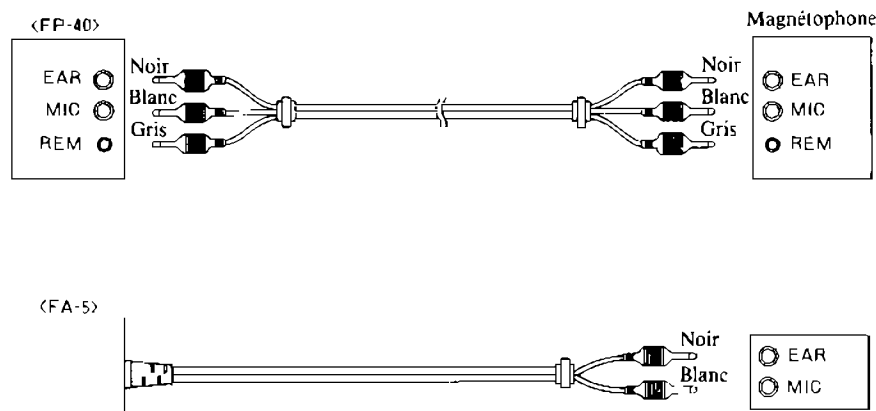
Nous donnerons maintenant une brève description de chaque fonction. Pour plus de détail, voir la description des commandes en Chapitre 7.

• Chargement des programmes et données sur une cassette

Pour charger des programmes et données sur une cassette, raccorder un magnétophone à l'ordinateur à l'aide de FP-40 ou FA-5.

< Raccordement des bornes >

Raccorder les bornes de la manière suivante



MIC : Borne MIC. Pour un magnétophone stéréo, utiliser la borne DROITE.

EAR : Borne EAR, MONITOR ou EX SP. Pour un magnétophone stéréo, raccorder la même borne que celle utilisée pour l'enregistrement.

REM : Raccorder la borne REM à la borne REMOTE (télécommande) du magnétophone.

* Bien que les deux câbles puissent être raccordés en même temps, certains magnétophones produisent du bruit quand les bornes EAR et MIC sont toutes deux raccordées. Dans un tel cas, raccorder la seule borne MIC pendant l'enregistrement (de l'ordinateur à la bande magnétique), puis la seule borne EAR pendant l'écoute (de la bande magnétique à l'ordinateur).

Opération du magnétophone

< Avec borne REMOTE raccordée (FP-40) >

- Exécuter la commande SAVE ou PUT, le magnétophone étant réglé sur "RECORD" (enregistrement).
- Exécuter la commande LOAD ou GET, le magnétophone étant réglé sur "PLAYBACK" (écoute).

< Avec borne REMOTE non raccordée (FA-5) >

- Exécuter la commande SAVE ou PUT, le magnétophone étant réglé sur "RECORD" (enregistrement).
- Régler le magnétophone sur "PLAYBACK" et le mettre en marche après avoir d'abord exécuté la commande LOAD ou GET.

Mise en mémoire et chargement des programmes

Il arrive parfois qu'un programme ne puisse pas être mis en mémoire en raison de la capacité de la zone libre. Tout programme effacé ne peut plus être utilisé. Dans les cas de ce type, un interface cassette est très utile. Les commandes de mise en mémoire des programmes sur cassette sont "SAVE" ou "SAVE ALL". "SAVE" ne peut mettre en mémoire qu'un programme se trouvant dans une zone de programme alors que "SAVE ALL" permet de mettre simultanément en mémoire les programmes se trouvant dans toutes les zones de programme.

Commande SAVE



Ready Pn

Le programme se trouvant dans cette zone de programme sera mis en mémoire.

Commande SAVE ALL

Les programmes se trouvant dans toutes les zones de programme seront mis en mémoire.

Les commandes SAVE et SAVE ALL sont exécutées manuellement.

Exemple:

```

SAVE [EXE]
SAVE "CASIO" [EXE]
SAVE ALL [EXE]
SAVE ALL "FX" [EXE]

```

Les caractères placés entre guillemets (") après SAVE et SAVE ALL sont les noms des fichiers accompagnant les programmes en mémoire. Ces programmes pourront ensuite être chargés en spécifiant leur nom. Un nom de fichier peut être composé d'un nombre maximum de 8 caractères.

Les commandes LOAD et LOAD ALL servent à charger les programmes dans l'ordinateur depuis une cassette. L'utilisation de l'une ou de l'autre commande dépend de la commande SAVE ou SAVE ALL qui a été utilisée pour mettre les programmes en mémoire.

Mise en mémoire \ Chargement	LOAD	LOAD "nom de fichier"	LOAD ALL	LOAD ALL "nom de fichier"
SAVE	○	×	×	×
SAVE "nom de fichier"	○	○	×	×
SAVE ALL	×	×	○	×
SAVE ALL "nom de fichier"	×	×	○	○

"○" indique que le chargement est possible et "×" que le chargement est impossible.

Les noms des fichiers doivent être identiques.

Exemple:

```

LOAD [EXE]
LOAD "nom de fichier" [EXE]
LOAD ALL [EXE]
LOAD ALL "nom de fichier" [EXE]

```

Quand les programmes sont chargés à l'aide de LOAD ou LOAD ALL, un affichage fonction du format de mise en mémoire apparaît.

Format de mise en mémoire	Affichage
SAVE	PF:
SAVE "nom de fichier"	PF: nom de fichier
SAVE ALL	AF:
SAVE ALL "nom de fichier"	AF: nom de fichier

Un programme mis en mémoire à l'aide de la commande SAVE peut être chargé dans l'une quelconque des zones de programme à l'aide de la commande LOAD.

Exemple:

Mettre en mémoire le programme de P0.

↓

Le charger en P9.

Précautions:

Il arrive parfois qu'il soit difficile de mettre en mémoire ou de charger un programme. Le cas échéant, vérifier ce qui suit.

- "Error 9" est affiché pendant la mise en mémoire.

[Point de contrôle]

Vérifier que l'ordinateur est bien raccordé à l'interface cassette.

- "Error 9" est affiché pendant le chargement.

[Points de contrôle]

Si la bande magnétique est détendue, changer de cassette. Si les têtes de lecture sont sales, les nettoyer.

Régler la commande de tonalité du magnétophone sur médium.

- Aucune erreur n'est affichée mais le chargement est impossible.

[Points de contrôle]

Si le volume de sortie du magnétophone est bas, l'amener sur MAX. Vérifier dans le manuel de l'utilisateur que la caractéristique de sortie du magnétophone est conforme à celle de l'interface cassette.

Mise en mémoire et chargement des données dans la DATA BANK

Toutes les données se trouvant dans la DATA BANK peuvent être mises en mémoire sur une cassette en une seule fois à l'aide de "SAVE #".

SAVE # "Nom de fichier"

Jusqu'à 8 caractères.

Un nom de fichier identique à celui du programme mis en mémoire peut comprendre jusqu'à huit caractères placés entre guillemets " ".

Exemple:

SAVE # "MEMO" 

"LOAD #" sert à charger les données depuis une cassette dans la DATA BANK de l'ordinateur.

Les anciennes données sont effacées quand les nouvelles données sont chargées. Toutefois, si ",M" est spécifié à la fin de la commande LOAD #, les données sont chargées depuis la cassette à la suite des anciennes données qui ne sont pas effacées.

LOAD# "Nom de fichier"

Jusqu'à 8 caractères.

Exemple:

LOAD# "MEMO" 

Quand les données dans la DATA BANK sont chargées, un affichage fonction du format de mise en mémoire apparaît.

Format de mise en mémoire	Affichage
SAVE #	MF:
SAVE # "nom de fichier"	MF:nom de fichier

Mise en mémoire et chargement des données

Un programme comprend toujours des données et il peut s'avérer fastidieux d'entrer à chaque fois ces données depuis le clavier.

Il existe une méthode qui permet de mettre en mémoire les données sur une cassette puis de les charger dans l'ordinateur.

Pour mettre les données en mémoire sur une cassette on utilise "PUT".

Des variables sont spécifiées dans une commande PUT. On peut aussi spécifier un nom de fichier.

PUT "Nom de fichier" **Variable 1, Variable 2**

Jusqu'à 8 caractères.

Un nom de fichier identique à celui du programme mis en mémoire peut comprendre jusqu'à huit caractères placés entre guillemets " ".

Si on utilise la variable caractère exclusive (\$), la spécifier en premier. Spécifier ensuite les noms de deux variables suivantes pour déterminer le début et la fin des variables à mettre en mémoire.

Exemple:

Mettre en mémoire la variable caractère exclusive (\$) et 13 variables de A à M.

PUT \$,A,M

Mise en mémoire du contenu de la variable de tableau A(10) sous le nom de fichier "DATA" en mode DIM.

PUT "DATA" A(*)

Mise en mémoire du contenu des 36 variables de A à Z(10) sous le nom de fichier "CASIO" en mode DEFM.

PUT *CASIO A,Z(10)

* En supposant que le nombre de variables ait été augmenté.

Etant donnée que les noms des variables spécifient le début et la fin des variables à mettre en mémoire, les placer dans l'ordre alphabétique (par ex. "A,Z"). Il n'est pas possible de spécifier "Z, A".

Si les variables sont des variables caractère, on peut spécifier "A,Z" au lieu de "A\$,Z\$".

"GET" sert à charger les données de la cassette dans l'ordinateur. Des variables sont spécifiées dans une commande GET. Un nom de fichier peut également être spécifié.

GET "Nom de fichier" Variable 1, Variable 2

Jusqu'à 8 caractères.

Exemple:

Charger les données sur la variable caractère exclusive (\$) et les 3 variables X à Z.

GET \$,X,Z

Charge les données du fichier "DATA" dans la variable de tableau A(15) en mode DIM.

GET "DATA" A(*)

Charge les données du fichier "FX" dans les variables de tableau G(0) à G(59) en mode DEFM.

GET *FX* G(0),G(59)

* En supposant que le nombre de variables ait été augmenté.

Pour le détail, voir GET et PUT en Chapitre 7 "Détail des commandes". Quand les données sont chargées à l'aide d'une commande GET, un affichage fonction du format de mise en mémoire apparaît.

Format de mise en mémoire	Affichage
PUT \$, A, Z	VF: _____
PUT "nom de fichier" G, P	VF: nom de fichier _____

• Impression des programmes, des données et des résultats des calculs

Une imprimante à caractères exclusive (FP-40 ou FP-12S) permet d'imprimer les programmes, les données et les résultats de calcul sur papier. Voir la méthode de raccordement en page 109.

L'impression est commandée (à partir du clavier) en mode PRINT ("PRT ON" sur l'écran). Le mode PRINT peut être spécifié à l'aide des touches   et libéré à l'aide des touches  .

1. Impression du contenu des programmes

Exécuter la commande LIST après avoir appuyé sur les touches   en mode RUN. Pour imprimer le contenu de toutes les zones de programme de P0 à P9, entrer LIST ALL .

Quand l'impression est terminée, ne pas oublier d'appuyer sur   pour libérer le mode PRINT.

2. Impression des résultats des calculs

Pour imprimer les résultats des calculs, spécifier le mode PRINT en appuyant sur   ou en écrivant "MODE 7" dans le programme. Il est plus pratique d'écrire "MODE 7" dans le programme si une partie seulement doit être imprimée.

* Dans le programme, écrire     au lieu d'appuyer sur la touche .

3. Impression des données dans la DATA BANK

Pour imprimer les données se trouvant dans la DATA BANK (voir Chapitre 8), exécuter la commande LIST # après avoir appuyé sur   en mode RUN.

* Quand l'impression est terminée, libérer le mode PRINT en appuyant sur  .

5-8 Utilisation d'un programme PB-100

Cet ordinateur accepte, entre autres, les programmes préparés pour le PB-100, le PB-300, le FX-700P, le FX-802P, le FX-785P et le FX-790P.

Cet ordinateur comprend plus de commandes que ces modèles et son utilisation est plus pratique.

Le langage BASIC utilisé avec cet ordinateur est pratiquement identique à celui utilisé avec les ordinateurs précédents.

Différents points

• Commande supplémentaires

PASS (Protection de programme)

BEEP (Sonnerie)

READ (Lecture des données depuis une instruction DATA)

DATA (Ecriture de données)

RESTORE (Spécification des données à lire)

ON ~ GOTO (Spécification indirecte d'une instruction GOTO)

ON ~ GOSUB (Spécification indirecte d'une instruction GOSUB)

REM (Instruction commentaire)

DIM (Déclaration de tableau)

ERASE (Effacement du tableau)

LIST V (Confirmation de la variable de tableau)

LIST # (Liste des données mémoire dans la DATA BANK)

LOAD # (Chargement des données mémoire dans la DATA BANK)

SAVE # (Mise des données mémoire dans la DATA BANK)

NEW # (Efface des données mémoire dans la DATA BANK)

RESTORE # (Recherche les données mémoire dans la DATA BANK)

READ # (Lecture des données mémoire dans la DATA BANK)

WRITE # (Ecriture des données mémoire dans la DATA BANK)

• Fonctions supplémentaires

DEG (Conversion sexagésimale → décimale)

DMS\$ (Conversion décimale → sexagésimale)

STR\$ (Conversion d'une valeur numérique en chaîne de caractères)

&H (Conversion hexadécimal — décimal)

HEX\$ (Conversion décimal — hexadécimal)

• Commandes modifiées

Cet ordinateur	Série PB-100
NEW (NEW ALL)	CLEAR (CLEAR A)
CLEAR	VAC
IF ~ THEN	IF ~;
SAVE ALL	SAVE A
LOAD ALL	LOAD A
VERIFY	VER
DEFM (Peut être écrit dans un programme)	DEFM (Manuellement uniquement)

• Fonctions modifiées

Cet ordinateur	Série PB-100
KEY\$	KEY
MID\$	MID

En dépit de ces différences, un programme préparé pour les ordinateurs de la série PB-100 peut en principe être utilisé sur cet ordinateur.

Toutefois, il est préférable de réécrire les programmes pour en faciliter l'utilisation ou la révision ultérieure.

Exemple:

Programme PB-100.

```
10 VAC
20 FOR A=1 TO 20
30 INPUT Z(A)
40 IF Z(A)>80;B=B+1;GOTO 90
50 IF Z(A)<60;C=C+1;GOTO 90
60 IF Z(A)>40;D=D+1;GOTO 90
70 IF Z(A)>20;E=E+1;GOTO 90
80 F=F+1
90 NEXT A
```

Cet exemple est une partie de programme pour l'entrée de données et leur distribution en fonction de leur taille. Le programme est utilisable tel quel mais opérer les corrections suivantes.

A la ligne 10, changer "VAC" per "CLEAR".

```
10 CLEAR
```

Aux lignes 40 à 70, changer ";" par "THEN".

```
40 IF Z(A)>80 THEN B=B+1:GOTO 90
```

```
...
```

Une extension de variable étant nécessaire dans ce programme, écrire la commande DEFM au début, commande qui doit être exécutée manuellement sur les modèles PB-100/PB-300/FX-700P/FX-802P.

```
5 DEFM 20
```

Exemple:

Programme PB-100

```
10 INPUT "I=1/O=2/P=3",N
20 IF N<1 THEN 10
30 IF N>3 THEN 10
40 GOTO N*100
```

```
...
```

Ce programme est utilisé pour déterminer la destination d'un branchement en fonction du travail.

Pour l'adapter à cet ordinateur le modifier comme suit en utilisant une instruction ON ~ GOTO.

```
10 INPUT "I=1/O=2/P=3",N
20 ON N GOTO 100,200,300
30 GOTO 10
```

```
...
```

Le programme est simplifié par l'utilisation de l'instruction ON ~ GOTO; l'essai de la donnée N est supprimé.

Les programmes et les données mis en mémoire sur cassette par des ordinateurs de poche CASIO peuvent être chargés tels quels sur cet ordinateur. Toutefois, l'opération inverse n'est pas toujours possible et certaines précaution doivent être prises. Les compatibilités se définissent comme suit.

Cet ordinateur — PB-110, PB-220, PB-410, FX-720P, FX-820P,
FX-730P, FX-770P, FX-785P, FX-790P

SAVE LOAD	PF	AF	MF	Avec mot de passe		
				PF	AF	MF
LOAD	○	/	/	○	/	/
LOAD ALL	/	○	/	/	○	/
LOAD#	/	/	○	/	/	○

Cet ordinateur — PB-100F, FX-710P

SAVE LOAD	PF	AF	MF	Avec mot de passe		
				PF	AF	MF
LOAD	○	/	/	○	/	/
LOAD ALL	/	○	/	/	○	/

Cet ordinateur → PB-100, PB-300, FX-700P, FX-802P

SAVE LOAD	PF	AF	MF	Avec mot de passe		
				PF	AF	MF
LOAD	○	/	/	/	/	/
LOAD ALL	/	○	/	/	/	/

○ : Chargement possible
/ : Chargement impossible

PRECAUTIONS

- Il est possible de transférer un programme préparé à l'aide de cet ordinateur sur un autre ordinateur CASIO (sauf PB-110, PB-220, PB-410, FX-720P, FX-820P, FX-770P, FX-785P et FX-790P) si le programme ne comprend pas une commande READ#, WRITE# ou RESTORE#. Avec les ordinateurs de la série PB-100, utiliser KEY et MID à la place de KEY\$ et MID\$.
- Il n'est pas possible d'exécuter correctement un programme créé sur un autre ordinateur CASIO.

Si une expression numérique est utilisée à un branchement IF ~ THEN, une erreur se produit. Dans ce cas, corriger en branchement IF ~ THEN GOTO.

- Il est possible de transférer un programme préparé à l'aide de cet ordinateur sur un autre ordinateur CASIO (sauf FX-785P et FX-790P), si le programme ne comprend pas une commands DIM, ERASE.

Utiliser une variable de tableau dans l'instruction DIM après la première modification en instruction DEFM.

Attention à la méthode de détermination de la variable de tableau au moment de la modification. (Voir page 101)

Bibliothèque de programmes

Ce chapitre est consacré à des programmes d'ensemble légèrement plus longs. Le premier de ceux-ci est un programme de tri dans lequel les données sont redispesées. Dans ce programme, on utilise des variables de tableau de manière intéressante. Le deuxième programme est un jeu amusant. A partir de l'exemple constitué par ces deux programmes, notez l'emploi des diverses commandes.

6-1 Redispotion des données (Tri)

Pour créer des programmes pratiques, il s'avère parfois important de trier les données disposées dans le désordre. Le programme présenté ici fait appel à la méthode dite de "tri par permutation".

Soit un programme permettant d'afficher le classement de huit candidats numérotés de 101 à 108 par triage des points attribués.

• Liste du programme

```

10 REM bubble sort
20 INPUT "Number of data",C
30 REM DIMENSION
40 ERASE D:DIM D(3,C)
50 REM data set
60 FOR A=1 TO C
70 READ D(1,A),D(2,A)
80 NEXT A
90 PRINT "sorting":
100 REM sort
110 FOR A=1 TO C-1
120 FOR B=A+1 TO C
130 IF D(2,A)≥D(2,B) THEN 160
140 D=D(1,A):D(1,A)=D(1,B):D(1,B)=D
150 D=D(2,A):D(2,A)=D(2,B):D(2,B)=D
160 NEXT B
170 NEXT A
180 FOR A=1 TO C:D(3,A)=A: NEXT A
190 FOR A=1 TO C-1
200 IF D(2,A)>D(2,A+1) THEN 220
210 D(3,A+1)=D(3,A)
220 NEXT A
300 REM output
310 FOR A=1 TO C
320 BEEP : PRINT
330 PRINT D(3,A): CSR5;"No":D(1,A): CS
R15;"+":D(2,A)
340 NEXT A
350 PRINT "OVER"

```

```

360 END
1000 DATA 101,40
1010 DATA 102,60
1020 DATA 103,80
1030 DATA 104,60
1040 DATA 105,60
1050 DATA 106,80
1060 DATA 107,20
1070 DATA 108,95

```

• Liste des variables

Variable	Rôle	Variable	Rôle
A	Variables de commande de boucle	D (2,1)	Points
B		:	
C	Nombre de données	:	
D	Utilisée pour conversion	D (2,C)	Classement
D (1,1)	Nombre de candidats	D (3,1)	
:		:	
D (1,C)		D (3,C)	

• Description du programme

Ligne No. 20 — 40: Entrée du nombre de données et déclaration de tableau.

Ligne No. 50 — 80: Chargement des données dans la ligne No. 1000 puis dans le tableau.

Ligne No. 100 — 170: Programme principal de tri. Les totaux de points sont successivement comparés et le total des points et le numéro du candidat sont échangés avec la ligne numéro 140 et 150 si le total le plus grand et le plus petit sont en ordre inverse.

Ligne No. 180 — 220: Un rang est assigné aux données triées en fonction de la séquence des totaux de points.

Pour une personne de rang A, la correspondance entre les variables de tableau sera comme indiqué ci-dessous.

No. du candidat Total des points Rang
 $D(1, A) \longleftrightarrow D(2, A) \longleftrightarrow D(3, A)$

Ligne No. 300 — 360: Affiche les données triées dans l'ordre de classement, No. des candidats et total des points.

Exemple d'exécution

Opération	Affichage
RUN	Number of data?
↳	8
END	sortins
	1 No 108 + 95
END	2 No 106 + 80
END	2 No 103 + 80
END	4 No 105 + 60
END	4 No 102 + 60
END	4 No 104 + 60
END	7 No 101 + 40
END	8 No 107 + 20
END	OVER

- Comme les données triées restent dans le tableau $D(1,1) \sim D(3,C)$, elles peuvent être utilisées dans un programme différent.

6-2 Jeu des courses de chevaux

Placez vos enjeux compte tenu des chances de 4 chevaux dans la course suivante. Commencez avec 20 dollars et faites fortune en trouvant l'outsider.

• Liste du programme

```

10 REM Horse Race
20 CLEAR : DIM A(3,4)
30 R=1:$="****"
40 PRINT "< Horse Race >";
50 FOR J=1 TO 5: BEEP : BEEP 1: NEXT
   J
60 PRINT
70 PRINT "HORSE":
80 FOR J=1 TO 4: PRINT J: MID$(J,1):
   NEXT J
90 GOSUB 1000: GOSUB 1000
100 PRINT : BEEP
110 INPUT "How many players ",P
120 IF P>5 THEN 110
130 IF P<1 THEN 110
140 PRINT "ALL PLAYERS HAVE $20":
150 GOSUB 1000
160 DIM X(2,P):Y$(P)
170 FOR J=1 TO P:X(2,J)=20: NEXT J
180 REM initialize
190 G=0
200 FOR J=1 TO 4
210 A(1,J)=0:A(2,J)=RAN#:A(3,J)=1+ IN
   T(10*(1.2-A(2,J)))
220 NEXT J
230 PRINT : PRINT "<RACE>:R:>": GOSU
   B 1000
240 REM bet money
250 FOR J=1 TO P
260 PRINT :X(1,J)=0:Y$(J)=""
270 IF X(2,J)=0 THEN 450
280 PRINT "PLAYER":J:" HAS $":X(2,J):
290 GOSUB 1000
300 PRINT : PRINT "RATE ";
```

```

310 FOR K=1 TO 4
320 PRINT CSRK*5; MID$(K,1):A(3,K);
330 NEXT K
340 GOSUB 1000: BEEP
350 PRINT CSR0;"P":J;"<";
360 A$= KEY$: IF A$="" THEN 360
370 IF A$<"0" THEN 360
380 IF A$>"4" THEN 360
390 N= VAL(A$): IF N=0 THEN 450
400 A$= MID$(N,1):V$(J)=A$
410 BEEP : PRINT : PRINT "PLAYER":J;"
    ":A$:
420 INPUT " MONEY ",X(1,J)
430 IF X(2,J)<X(1,J) THEN 410
440 X(2,J)=X(2,J)-X(1,J)
450 NEXT J
460 PRINT
470 PRINT " < START! >":
480 FOR K=1 TO 10: BEEP : NEXT K
490 PRINT
500 REM main loop
510 IF G=2 THEN 600
520 FOR J=1 TO 4
530 IF G=1 THEN 560
540 PRINT CSRA(1,J):" ":
550 IF RAN#*(0.9+A(2,J)/10)>0.7 THEN A
    (1,J)=A(1,J)+1
560 IF A(1,J)=23 THEN G=G+1
570 PRINT CSRA(1,J): MID$(J,1);
580 NEXT J
590 GOTO 500
600 REM goal
610 PRINT CSR0;"GOAL!":
620 FOR J=1 TO 7: BEEP : BEEP 1: NEXT
    J
630 GOSUB 1000
640 FOR J=1 TO 4
650 IF A(1,J)=23 THEN H=A(3,J):A$= MID
    $(J,1)

```

```

660 NEXT J
670 F=0
680 FOR J=1 TO P
690 M=0: IF X(1,J)=0 THEN 730
700 IF V$(J)=A$ THEN M=X(1,J)*H
710 PRINT : BEEP
720 PRINT "PLAYER":J;" <PRIZE $":M;
730 X(2,J)=X(2,J)+M: GOSUB 1000
740 PRINT : BEEP : IF X(2,J)=0 THEN F=
    F+1
750 PRINT "PLAYER":J;" HAS $":X(2,J):
    GOSUB 1000
760 NEXT J
770 PRINT : BEEP : IF F=P THEN 830
780 PRINT "REPLAY [Y/N] ?":
790 A$= KEY$: IF A$="" THEN 790
800 IF A$="Y" THEN R=R+1: GOTO 180
810 IF A$="N" THEN 830
820 GOTO 790
830 PRINT : PRINT "GAME OVER"
840 END
1000 REM timer sub
1010 FOR K=1 TO 150: NEXT K
1020 RETURN

```

• Liste des variables

Variable	Rôle	Variable	Rôle
A\$	Pour touches et caractères.	K	Variable de commande de boucle
A(1,1)	Position du pique (♠)	M	Pour calcul du prix
A(1,2)	Position du coeur (♥)	N	Pour le No. du cheval
A(1,3)	Position du carreau (♦)	P	Nombre de joueurs
A(1,4)	Position du trèfle (♣)	R	No. de la course
A(2,1)	Nombre aléatoire du pique (♠)	X(1,1)	Enjeu joueur 1
A(2,2)	Nombre aléatoire du coeur (♥)	X(1,2)	Enjeu du joueur 2
A(2,3)	Nombre aléatoire du carreau (♦)	⋮	⋮
A(2,4)	Nombre aléatoire du trèfle (♣)	X(1,P)	Enjeu du joueur P
A(3,1)	Cote du pique (♠)	X(2,1)	Avoir du joueur 1
A(3,2)	Cote du coeur (♥)	X(2,2)	Avoir du joueur 2
A(3,3)	Cote du carreau (♦)	⋮	⋮
A(3,4)	Cote du trèfle (♣)	X(2,P)	Avoir du joueur P
F	Détermine la fin du jeu	\$	Sélection du caractère du cheval (♠, ♥, ♦ ou ♣)
G	Détermine le but		
H	Cote d'un cheval à la gagne		
J	Variable de commande de boucle		

• Description du jeu

Jusqu'à cinq personnes peuvent participer à ce jeu, chaque joueur commençant avec 20 dollars. Il y a quatre chevaux, numérotés de 1 à 4, le No.1 étant représenté par pique (♠), le 2 par coeur (♥), le 3 par carreau (♦) et le 4 par trèfle (♣). Choisir un cheval entre 1 et 4. Un joueur qui choisit 0 fait impasse sur la course puisque ce numéro n'est pas attribué.

La cote de chacun des chevaux est affichée pour chaque course. Si le cheval choisi par l'un des joueurs gagne, ce dernier est payé d'un montant conforme à la cote de son enjeu. Si le cheval ne gagne pas, le joueur perd son enjeu. Quand un joueur a perdu la totalité de son avoir, il ne peut plus participer au jeu. Le jeu est terminé quand tous les joueurs ont perdu la totalité de leur avoir.

Voici, à titre d'exemple, une description des affichages successifs et des opérations sur le clavier.

1) Début du jeu.

RUN (Affichage du titre)

(Description des chevaux)

2) Entrer le nombre de joueurs.

2 (2 joueur sont entrés)

ERE (Avoir initial)

(Première course)

3) Entrer chevaux et paris.

(Avoir du joueur 1)

(Cotes des chevaux)

2 (Cheval ♥ choisi)

10 (Pari de 10 dollars)

ERE (Avoir du joueur 2)

(Cotes des chevaux)

(Le joueur 2 choisit son cheval)

3 (Cheval ♦ choisi)

5 (Pari de 5 dollars)

<Horse Race>

HORSE 1♠ 2♥ 3♦ 4♣

How many players ?

2

ALL PLAYERS HAVE \$20

<RACE 1>

PLAYER 1 HAS \$ 20

RATE ♠ 12 ♥ 3 ♦ 7 ♣ 12

1 2 3 4

P 1+ ♠ 12 ♥ 3 ♦ 7 ♣ 12

PLAYER 1 ♥ MONEY ?

10

PLAYER 2 HAS \$ 20

RATE ♠ 12 ♥ 3 ♦ 7 ♣ 12

P 2+ ♠ 12 ♥ 3 ♦ 7 ♣ 12

PLAYER 2 ♦ MONEY ?

5

4) La course commence.

EXE

(La course se déroule)

```
< START! >
♣♣♣♣
```

5) Fin de la course

(Arrivée. ♣ a gagné)

(Le joueur 1 perd)

(Avoir du joueur 1)

(Le joueur 2 gagne 35 dollars)

(Avoir du joueur 2)

(Voulez-vous jouer de nouveau ?)

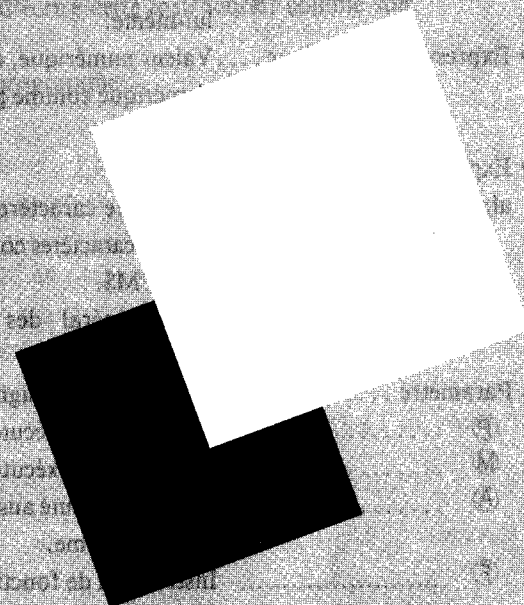
```
GOAL!           ♣   ♣   ♣
PLAYER 1 +PRIZE $ 0
PLAYER 1 HAS $ 10
PLAYER 2 +PRIZE $ 35
PLAYER 2 HAS $ 50
REPLAY [Y/N] ?
```

6) Appuyer sur **[Y]** pour passer à la course suivante ou sur **[N]** pour "GAME OVER" (fin du jeu).

- Ce jeu est programmé de telle manière que plus la cote d'un cheval est élevée plus celui-ci a de chances de gagner et inversement. Bonne chance!

Détail des commandes

7



On trouvera ci-dessous la liste des symboles et des termes fréquemment utilisés dans la syntaxe.

- { ×××× } Choisir l'un des élément entre { }.
Ne pas écrire les { } mêmes.
- [○○○○] L'élément entre [] est facultatif.
Ne pas écrire les [] mêmes.
- ○○○○* L'élément repéré par un * en haut et à droite peut être utilisé de manière répétée. Ne pas écrire l'* lui-même.
- Expression numérique Valeur numérique, expression de calcul et variable numérique comme par exemple 10, 2 + 3, A, S * Q.
- Expression alphanumérique Constante caractère, variable caractère et expression de caractères comme par exemple "ABC", X\$, N\$ + M\$.
- Expression Nom général des expressions numériques et alphanumériques.
- Paramètre Élément accompagnant une commande.
- (P) Ne peut être exécuté que dans un programme.
- (M) Ne peut être exécuté que manuellement.
- (A) Peut être exécuté aussi bien manuellement que dans un programme.
- (F) Instruction de fonction qui peut être exécutée aussi bien manuellement que dans un programme.

Exemple:

DATA [données][,[données]]*

Toutes les données étant placées entre crochets [], on pourra se contenter d'écrire simplement "DATA", Comme [données] est accompagné de []*, cet élément peut être écrit de manière répétée. On pourra donc écrire "DATA data, data, ...". Si le premier [données] est omis, on pourra également écrire "DATA, données, données, ...".

GOTO { No. de ligne
No. de zone de programme }

Il y a deux façons différentes d'écrire cette instruction, comme suit.

- 1) GOTO No. de ligne
- 2) GOTO# No. de zone de programme

NEW [ALL]**Fonction:**

Effacement de programme. Permet d'effacer programmes et variables.

Paramètre:

Quand ALL est spécifié, tous les programmes P0 à P9 et toutes les variables sont effacés.

Explication:

- 1) Si ALL n'est pas spécifié, le programme effacé est celui dans la zone de programme présentement spécifié. Le contenu des variables n'est pas annulé et les variables supplémentaires ne sont pas effacées.
 - 2) Si ALL est spécifié, les programmes dans toutes les zones de programme et les variables sont effacés. Le nombre des variables spécifiées par DEFM peut être initialisé à 26 et l'ordinateur sera en mode DIM.
 - 3) Ne peut pas être exécuté si un mot de passe a été spécifié.
 - 4) Ne peut pas être utilisé dans un programme.
 - 5) Ne peut être exécuté qu'en mode WRT.
- * NEW ALL peut être abrégé par NEW A.

Exemple:

NEW

RUN [Ligne de départ d'exécution]

No. de ligne

Fonction:

Exécution d'un programme.

Paramètre:

Numéro de ligne de départ:

$1 \leq \text{numéro de ligne} < 10000$

Explication:

- 1) Exécute un programme à partir de la ligne spécifiée (si le numéro de ligne est omis, l'exécution commence au début du programme).
- 2) Si le numéro de ligne spécifié n'existe pas, l'exécution commence à partir de la ligne de numéro supérieur le plus proche.
- 3) Les variables ne sont pas annulées.

Exemple:

```
10 PRINT "LINE 10"
20 PRINT "LINE 20"
30 END
```

RUN

RUN 20

LINE 10
LINE 20

LIST

{ { No. ligne
ALL
V } }

**Fonction:**

Affichage du contenu du programme ou du nom des variables et des indices des tableaux déclarés.

Paramètre:

No. de ligne: No. de la première ligne à afficher.
 ALL: Affichage du contenu de tous les programmes P0 à P9 en séquence.
 V: Affichage du nom des variables et des indices des tableaux déclarés.

Explication:**1. Mode RUN**

- 1) Affichage en séquence du contenu d'un programme à partir d'un certain numéro de ligne si celui-ci est spécifié ou du début dans le cas contraire.
- 2) Le contenu d'un programme étant automatiquement affiché en séquence, appuyer sur la touche **END** pour arrêter l'affichage. Appuyer sur la touche **EXE** pour reprendre l'affichage à partir de la ligne suivante.
- 3) Les noms des variables de tableau sont affichés en séquence, un à la fois. Appuyer sur **EXE** pour afficher le nom de variable de tableau suivant.
- 4) En mode PRINT (quand "PRT ON" est affiché), l'affichage n'est pas arrêté mais exécuté en séquence à grande vitesse.

2. Mode WRT

- 1) Affichage en séquence du contenu d'un programme à partir d'un certain numéro de ligne si celui-ci est spécifié ou du début dans le cas contraire.
- 2) En mode WRT chaque ligne est affichée pour correction et si aucune correction n'est nécessaire, appuyer sur la touche **EXE** pour passer à la ligne suivante. Si, par ailleurs, la touche **END** est enclenchée avant la touche **EXE**, la ligne précédente est affichée.

- 3) Les noms des variables de tableau sont affichés un à la fois avec LIST V. Appuyer sur **EXE** pour afficher le nom de variable de tableau suivant.

- Quand ALL est spécifié, le contenu de tous les programmes P0 à P9 est affiché en séquence. Dans un tel cas, même en mode WRT, les programmes sont affichés en séquence et aucune correction ne peut être effectuée.
- Cette commande ne peut pas être utilisée si un mot de passe a été spécifié.
- * LIST ALL peut être abrégé par LIST A.

Exemple:

```
LIST EXE
LIST 30 EXE
LIST V EXE
```

PASS

"Mot de passe"
Chaîne de caractères

**Fonction:**

Désigne ou annule un mot de passe.

Paramètre:

Mot de passe: $1 \leq \text{chaîne de caractères} \leq 8$

Explication:

- 1) Si cette commande est exécutée alors qu'aucun mot de passe n'a été désigné, un mot de passe est désigné pour toutes les zones de programme (P0 à P9).
- 2) Si cette commande est exécutée quand un mot de passe a été désigné, ce mot de passe n'est annulé que lorsque le mot de passe correspondant a été entré. Si les mots de passe ne se correspondent pas, une erreur protection (Error 8) se produit.

- 3) Un mot de passe est constitué d'une chaîne de 1 à 8 caractères dans laquelle on peut utiliser des lettres, des chiffres, des espaces, des symboles spéciaux etc. Il n'est toutefois pas possible d'utiliser le signe ("").
- 4) Quand un mot de passe a été désigné, il n'est pas possible d'utiliser des commandes du type LIST, LIST ALL, LIST #, LIST V, NEW, NEW ALL et NEW #. Il n'est pas possible non plus d'écrire (Mode WRT); le cas échéant, une erreur (Error 8) se produit.
- 5) Ne peut pas être utilisé dans un programme.
- 6) Le mot de passe est sauvegardé même quand l'alimentation électrique est coupée.
- 7) Lorsqu'un mot de passe est désigné et qu'un programme est mis en mémoire dans une cassette à l'aide de la commande SAVE ou SAVE ALL, le mot de passe est également mis en mémoire. Quand un programme muni d'un mot de passe est chargé depuis une cassette à l'aide de la commande LOAD ou LOAD ALL, le mot de passe est également chargé. De plus, si le mot de passe désigné dans l'unité de traitement centrale diffère de celui du programme chargé depuis une cassette, il est impossible d'appeler le programme depuis la cassette (Error 8).

Précaution:

Si le mot de passe a été oublié une fois spécifié, appuyer sur le bouton RESET se trouvant sous l'interrupteur d'alimentation et annuler le mot de passe. Noter que cette opération efface tous les programmes et le contenu des variables.

Exemple:

PASS "CASIO" EXE

* La marche à suivre pour la désignation et l'annulation d'un mot de passe est la même.

SAVE [ALL]

["Nom de fichier"]
Chaîne de caractères



Fonction:

Mise en mémoire d'un programme sur une cassette.

Paramètre:

ALL: Mise en mémoire des programmes dans toutes les zones de programme.
Nom de fichier: 1 ≤ chaîne de caractères ≤ 8. Facultatif.

Explications:

- 1) Si ALL n'est pas spécifié, le contenu dans la zone de programme présentement spécifié est mis en mémoire.
- 2) Quand ALL est utilisé, le contenu de tous les programmes P0 à P9 est mis en mémoire.
- 3) Quand un mot de passe est désigné, la mise en mémoire est effectuée y compris ce mot de passe. Par conséquent, le mot de passe est le même que celui mis en mémoire quand le programme est chargé à l'aide de la commande LOAD.

* SAVE ALL peut être abrégé par SAVE A.

Exemple:

SAVE EXE
SAVE "CASIO" EXE
SAVE ALL "FX" EXE

LOAD [ALL] ["Nom de fichier"]

Chaîne de caractères

**Fonction:**

Chargement d'un programme depuis une cassette.

Paramètre:

ALL Chargement des programmes dans toutes les zones de programme.

Nom de fichier: $1 \leq \text{chaîne de caractères} \leq 8$. Facultatif.

Explication:

- 1) Si ALL n'est pas spécifié, c'est le programme mis en mémoire à l'aide de la commande "SAVE" qui est lu dans la zone de programme présentement spécifiée.
- 2) Si ALL est spécifié, les programmes mis en mémoire à l'aide de "SAVE ALL" sont lus dans les zones de programme P0 à P9.
- 3) Avec la commande "LOAD ALL", les programmes en mémoire dans l'ordinateur seront effacés et un nouveau programme sera chargé depuis la cassette.
- 4) Si un nom de fichier a été attribué, le programme portant le même nom de fichier sera cherché et chargé depuis la cassette. Si le nom de fichier a été omis, c'est le premier programme trouvé sur la cassette qui sera chargé.
- 5) Si le programme chargé comprend un mot de passe, ce dernier est également chargé.

* LOAD ALL peut être abrégé par LOAD A.

Relation entre SAVE et LOAD

	LOAD	LOAD "Nom de fichier"	LOAD ALL	LOAD ALL "Nom de fichier"
SAVE	○	×	×	×
SAVE "Nom de fichier"	○	○	×	×
SAVE ALL	×	×	○	×
SAVE ALL "Nom de fichier"	×	×	○	○

* Les noms de fichier sont supposés identiques.

○ ... Peut être chargé.

× ... Ne peut pas être chargé.

VERIFY ["Nom de fichier"]

Chaîne de caractères

**Fonction:**

Vérification de l'état d'un programme et des données en mémoire dans une cassette.

Paramètre:

Nom du fichier: $1 \leq \text{chaîne de caractères} \leq 8$. Facultatif.

Explication:

- 1) Quand un nom de fichier est spécifié, le fichier est vérifié avec son nom.
- 2) Quand aucun nom de fichier n'est spécifié, c'est le premier fichier qui apparaît sur la cassette qui est vérifié.
- 3) La vérification d'un format en mémoire est effectuée par un système de contrôle de parité.

Exemple:

```
VERIFY   
VERIFY "PROG1" 
```

CLEAR

Fonction:

Effacement de toutes les variables y compris les variables de tableau.

Explication:

- 1) Efface toutes les variables; toutes les variables numériques sont ramenées à 0 et toutes les variables alphanumériques sont annulées.
- 2) Cette commande peut être utilisée dans un programme aussi bien que manuellement.
- 3) Les variables de contrôle étant également effacées dans une boucle FOR ~ NEXT (voir page 158), une erreur (Error 7) se produit pendant l'exécution de l'instruction NEXT.

* La commande CLEAR fonctionne comme VAC.

7-2 Commandes de programme

END

Fonction:

Termine l'exécution d'un programme.

Explication:

L'exécution du programme étant terminée, le programme suivant n'est pas exécuté même s'il existe.

STOP

Fonction:

Suspend provisoirement l'exécution d'un programme.

Explication:

- 1) Suspension momentanée de l'exécution d'un programme et affichage du message "STOP", puis attente d'entrée.
- 2) Après la suspension, la touche  permet de reprendre l'exécution.
- 3) Si la touche  est enclenchée quand l'exécution est suspendue par une instruction STOP, le numéro de la zone de programme et le numéro de ligne sont affichés.
- 4) Des calculs peuvent être effectués manuellement quand l'exécution est suspendue par STOP.

(LET) { Variable numérique = expression numérique
Variable alphanumérique = expression alphanumérique } ^(P)

Fonction:

Assigne la valeur à droite du signe égal (=) à la variable se trouvant à gauche.

Explication:

- 1) Une expression numérique correspond à une variable numérique et une expression alphanumérique à une variable alphanumérique.
- 2) LET est facultatif.

Exemple:

```
10 LET X=12
20 Y=X↑2+2*X-1 ..... LET est facultatif.
30 PRINT Y
40 A$="CASIO" ..... LET est facultatif.
50 LET B$=A$+"FX"
60 PRINT B$
70 END
```

REM Commentaire
Chaîne de caractères ^(P)

Fonction:

Instruction exprimant un commentaire.

Explication:

- 1) Ecrit dans un programme. Le contenu après REM sur une ligne est traité comme une instruction commentaire et par conséquent n'est pas exécuté.

```
10 REM TEST:A=50
           ↘ Non exécuté.
```

- 2) Si une commande à exécuter est écrite sur la même ligne, marquer un signe de multi-instruction (:) avant l'instruction REM.

Exemple:

```
10 REM AREA ..... Détermine le nom du programme.
20 INPUT "R=",R
30 S=π*R↑2:REM CALCULATION ..... Ajoute un commentai-
40 PRINT S                                     re sur la ligne.
50 END
```

INPUT

["Instruction message",] nom de variable (P)
 Chaîne de caractères
 [, ["Instruction message",] Nom de variable]*
 Chaîne de caractères

Fonction:

Permet d'assigner une entrée à une variable à partir du clavier.

Paramètre:

Message: Chaîne de caractères
 Nom de variable: Nom de la variable numérique ou nom de la variable alphanumérique.

Explication:

- 1) L'entrée depuis le clavier est affectée à une variable quand la touche **[ENT]** est enclenchée.
- 2) Si un message est spécifié, "?" sera affiché après le message et l'ordinateur sera en état d'attente d'entrée par clavier.
- 3) Si aucun message n'est spécifié, seul le signe "?" sera affiché.
- 4) En état d'attente d'entrée par clavier avec l'instruction INPUT, le message et le signe "?" seront effacés avec la première entrée par clavier.
- 5) Une erreur (Error 2) se produit si une donnée chaîne de caractères est entrée sur une variable numérique. Quand la touche **[CLS]** est enclenchée, le signe "?" est de nouveau affiché et une entrée de valeur numérique est demandée, effectuer alors une nouvelle entrée. (Si la touche **[BKK]** est enclenchée, l'exécution du programme est terminée.)
- 6) On peut utiliser une expression numérique comme entrée sur une variable numérique qui se verra assigner cette valeur numérique. Dans un tel cas, même une seule lettre alphabétique sera considérée comme une expression numérique (variable).
- 7) Si la seule touche **[ENT]** est enclenchée, sans entrée de valeur ou d'expression numérique, la valeur dans la variable numérique ne sera pas modifiée.

- 8) Si la seule touche **[ENT]** est enclenchée sans entrée d'une expression alphanumérique au moment de l'entrée sur une variable alphanumérique, une absence d'information sera affectée à la variable alphanumérique.
- 9) Si une donnée constituée d'une chaîne de caractères de plus de sept caractères est entrée sur une variable alphanumérique, les sept premiers caractères seront significatifs et les huitième caractères et suivants seront ignorés.

Exemple:

```
10 INPUT A
20 INPUT "NAME=", B$
30 INPUT "C$=", C$, "D$=", D$
40 INPUT "FORM=", $
  :
```

KEY\$

(P)

Fonction:

Entrée d'un caractère à partir du clavier.

Explication:

- 1) N'accepte l'entrée que d'un seul caractère à partir du clavier.
 - 2) Chiffres, lettres alphabétiques et symboles sont acceptés.
 - 3) Les données entrées le seront sous la forme d'un caractère simple.
 - 4) Si aucune entrée clavier n'est effectuée, une absence d'information en résulte.
 - 5) Le signe "?" n'étant pas affiché et l'attente d'entrée ne se produisant pas non plus, KEY\$ est en général utilisée en conjonction avec une instruction IF.
- * KEY\$ peut être abrégé par KEY.

Exemple:

```
10 PRINT "BEEP" ;
20 A$=KEY$
30 IF A$="0" THEN BEEP 0
40 IF A$="1" THEN BEEP 1
50 IF A$="E" THEN 70
60 GOTO 20
70 PRINT:PRINT "END"
80 END
```

Se répète jusqu'à ce que la touche [E] soit enclenchée.

* Un son grave est produit quand la touche [G] est enclenchée et un son aigu quand c'est la touche [H] qui est enclenchée. Quand la touche [E] est enclenchée, "END" est affiché et il est mis fin au programme.

PRINT [Elément de sortie] | { ' ; } [Elément de sortie]]*

(P)

Fonction:

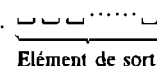
Affichage d'un élément de sortie.

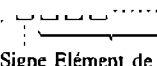
Paramètre:

Elément de sortie: Fonction de contrôle de sortie (CSR), expression numérique, expression alphanumérique.

Explication:

- 1) Affichage d'un élément de sortie. Quand une fonction de commande de sortie est ajoutée, l'élément est affiché à l'endroit déterminé par cette fonction.
- 2) Les valeurs sont affichées pour les expressions numériques et les expressions alphanumériques.
- 3) Quand l'élément de sortie est une expression numérique, une position pour le signe (+, -) est prévue devant la valeur. Mais le signe + n'est pas affiché en clair.

- Affichage alphanumérique 

Elément de sortie
- Affichage numérique 

Signe Elément de sortie

- 4) Si l'élément de sortie est une expression numérique et si la mantisse est de plus de 10 chiffres, le 11ème chiffre est arrondi. Si, en plus de la mantisse, un exposant est présent, le signe (E) de l'exposant et ses deux chiffres sont affichés.
- 5) "," et ";" peuvent être utilisés en guise de ponctuation entre des éléments de sortie. Quand on utilise ",", l'exécution s'arrête (STOP est affiché) après affichage du premier élément de sortie et l'élément de sortie suivant est affiché quand on appuie sur la touche [R]. Avec ";", l'élément de sortie suivant est affiché immédiatement après le premier élément.
- 6) Si aucun élément de sortie n'a été spécifié (seul PRINT est écrit), l'affichage est effacé mais n'est pas arrêté.

- 7) L'affichage n'est pas arrêté pendant l'impression en mode impression ().
- 8) Le format pour l'affichage des valeurs numériques peut être spécifié par une instruction SET.

Exemple:

```
10 PRINT 1/3
20 PRINT "A=" ; A
30 PRINT "SIN 30" , SIN 30
40 PRINT "END" ;
50 PRINT
60 END
```

CSR

Spécification de l'emplacement de sortie
Expression numérique

**Fonction:**

Affichage d'un élément de sortie à partir d'un emplacement spécifié.

Paramètre:

Spécification de

l'emplacement de sortie: Expression numérique. Les valeurs après la virgule sont ignorées.

$0 \leq \text{spécification} < 24$

Explication:

- Utilisé dans une instruction PRINT pour spécifier l'emplacement d'un élément de sortie.
- L'emplacement de sortie à l'extrême gauche est 0.

**Exemple:**

```
10 FOR L=0 TO 23
20 PRINT CSRL ; "A " ; CSR 23-L ; "B "
30 NEXT L
40 END
```

- * Les caractères A et B passent respectivement de gauche à droite chaque fois que la touche  est enclenchée.

GOTO

No. de ligne de rupture de séquence
No. de ligne
No. de zone de programme #
Numéro 0 à 9

**Fonction:**

Rompit inconditionnellement une séquence vers un emplacement spécifié.

Paramètre:

No. de ligne: 1 à 9999.

No. de zone de programme: 0 à 9.

Explication:

- Rupture d'une séquence vers un emplacement spécifié.
- Si l'emplacement de la rupture de séquence est un numéro de ligne, la rupture de séquence s'effectue vers la ligne spécifiée de la zone de programme en cours et le programme est exécuté. Si le numéro de ligne de rupture de séquence n'existe pas, une erreur (Error 4) se produit.
- Si l'emplacement de la rupture de séquence est un numéro de zone de programme, la rupture de séquence s'effectue vers la zone de programme spécifiée et le programme est exécuté depuis le début.

* Une expression numérique peut être utilisée pour le numéro de ligne de rupture de séquence et le numéro de la zone de programme.

Exemple:

```

10 PRINT "START";
20 GOTO 100
30 PRINT "LINE 30"
40 END
100 BEEP 0:BEEP 1
110 PRINT
120 PRINT"LINE 120"
130 GOTO 30

```

ON Condition de rupture de séquence (P)
 Expression numérique
GOTO [Emplacement de rupture de séquence]
 [, [Emplacement de rupture de séquence]]*

★ Emplacement de rupture de séquence { No. de ligne
 No. de zone de programme #

Fonction:

Effectue une rupture de séquence sur un emplacement spécifié en fonction de la condition de rupture de séquence.

Paramètre:

Condition de rupture de séquence: Expression numérique. Les valeurs après la virgule sont ignorées.

No. de ligne: 1 à 9999.

No. de zone de programme: 0 à 9.

Explication:

- 1) La rupture de séquence s'effectue en fonction de la partie nombre entier de la valeur dans une expression de condition de rupture de séquence. Les emplacements de rupture de séquence sont alloués en séquence en fonction de la valeur de l'expression.

ON A GOTO $\frac{100}{A=1}, \frac{200}{A=2}, \frac{300}{A=3}, \dots$

- 2) Si la valeur de l'expression est inférieure à 1 ou s'il n'existe pas d'emplacement de rupture de séquence approprié, l'instruction suivante est exécutée sans rupture de séquence.
- 3) Il est possible d'écrire autant d'emplacements de rupture de séquence qu'une ligne peut en contenir.

Exemple:

```

10 INPUT "A=", A
20 ON A GOTO 100, 200, 300
30 PRINT "OTHER"
40 GOTO 10
100 PRINT "LINE 100":GOTO 10
200 PRINT "LINE 200":GOTO 10
300 PRINT "LINE 300":GOTO 10

```

- * Si 1 à 3 est entré, les ruptures de séquence vers 100 à 300 sont respectivement effectuées, sinon "OTHER" est affiché.

IF Expression conditionnelle
Expression de comparaison

(P)

THEN { Instruction [: instruction]*
Emplacement de rupture de séquence }

★ Emplacement de rupture de séquence { No. de ligne
No. de zone de programme# }

Fonction:

Quand une expression conditionnelle est vraie, l'instruction suivant THEN est exécutée. De plus, si l'instruction suivant THEN est un emplacement de rupture de séquence, une rupture de séquence est effectuée.

Paramètre:

Expression conditionnelle: Utilisée sous la forme d'une expression de comparaison.

No. de ligne: 1 à 9999.

No. de zone de programme: 0 à 9.

Explication:

- 1) Si une expression conditionnelle est vraie, les instructions suivant THEN sont exécutées ou une rupture de séquence est effectuée.
- 2) Si l'expression conditionnelle est fausse, la ligne suivante est exécutée.
- 3) Opérateurs utilisés dans les expressions conditionnelles:
 - = L'item placé à gauche est égal à l'item placé à droite.
 - ≠ L'item placé à gauche n'est pas égal à l'item placé à droite.
 - < L'item placé à droite est plus grand que l'item placé à gauche.
 - > L'item placé à droite est plus petit que l'item placé à gauche.
 - ≧ L'item placé à droite est plus grand ou égal à l'item placé à gauche.
 - ≦ L'item placé à droite est plus petit ou égal à l'item placé à gauche.
- 4) Si deux expressions conditionnelles ou plus existent, plusieurs instructions IF ~ THEN peuvent être écrites en séquence.

IF ~ THEN IF ~ THEN

- 5) Les instructions suivant THEN sur une même ligne n'étant exécutées que si toutes les expressions conditionnelles sont vraies, les instructions qui doivent être exécutées indépendamment de l'expression conditionnelle doivent être écrites sous la forme de multi-instructions avant IF ou sur une ligne séparée.

* Si une instruction existe après THEN, on peut utiliser ";" à place de THEN.

Exemple:

```
10 N=12
20 PRINT CSR N; "@" ;
30 K$=KEY$
40 IF K$="4" THEN N=N-1:IF N<0 THEN N=0
50 IF K$="6" THEN N=N+1:IF N>23 THEN N=23
60 PRINT
70 GOTO 20
```

* " @ " passe à gauche quand la touche [4] est enclenchée et à droite quand la touche [6] est enclenchée.

FOR Nom de la variable de contrôle = Valeur initiale (P)
Expression numérique
TO Valeur finale
Expression numérique **(STEP** Incrément
Expression numérique **)**
NEXT Nom de la variable de contrôle

Fonction:

Répétition du processus compris entre les instructions FOR et NEXT autant de fois que spécifié par la variable de contrôle. La valeur de cette variable passe d'une valeur initiale à une valeur finale par un incrément pour chaque répétition du processus.

Paramètre:

Nom de la variable de contrôle:	Nom de variable de type numérique simple. On ne peut pas utiliser de variable de tableau.
Valeur initiale:	Expression numérique
Valeur finale:	Expression numérique
Incrément:	Expression numérique La valeur 1 est prise par défaut.

Explication:

- 1) Répétition du processus compris entre les instructions FOR et NEXT autant de fois que spécifié par la variable de contrôle. La valeur de cette variable passe d'une valeur initiale à une valeur finale d'un incrément pour chaque répétition du processus. Quand la valeur de la variable de contrôle dépasse la valeur finale, la répétition se termine.
- 2) Si la valeur initiale est plus grande que la valeur finale, l'exécution entre FOR ~ NEXT n'est effectuée qu'une seule fois.
- 3) Lorsque l'incrément est positif, la valeur de la variable de contrôle augmente de chaque incrément. Si l'incrément est 0, l'exécution est répétée à l'infini. Lorsque l'incrément est négatif, la valeur de la variable de contrôle diminue à chaque fois de la valeur de l'incrément. Si l'incrément est omis, une valeur lui sera assignée.

- 4) Une instruction NEXT doit toujours correspondre à une instruction FOR et doit être écrite après cette dernière.
- 5) Les boucles FOR ~ NEXT peuvent avoir la structure imbriquée suivante.

```

10 FOR I=1 TO 10
20 FOR J=11 TO 20
30 PRINT I;" ":"J
40 NEXT J
50 NEXT I
60 END

```

- 6) L'imbrication peut être effectuée jusqu'à 4 niveaux maximum.
- 7) Quand une boucle FOR ~ NEXT est terminée, la valeur de la variable de contrôle est supérieure à la valeur finale de la valeur de l'incrément.
- 8) Il est possible de procéder à une rupture de séquence hors d'une boucle FOR ~ NEXT. Si une rupture de séquence est tentée à l'intérieur d'une boucle FOR ~ NEXT à l'aide d'une instruction IF ou GOTO, une erreur se produit.

GOSUB { No. de ligne de rupture de séquence } (P)
 No. de ligne
 No. de zone de programme #
 No. 0 à 9

Fonction:

Exécution d'une rupture de séquence sur un sous-programme spécifié.

Paramètre:

No. de ligne: 1 à 9999.
No. de zone de programme: 0 à 9.

ON B GOSUB 1000, 2000, 3000.....
B = 1 B = 2 B = 3

- 2) Si la valeur de l'expression est inférieure à 1 ou s'il n'existe pas d'emplacement de rupture de séquence approprié, l'instruction suivante est exécutée sans rupture de séquence.
- 3) Il est possible d'écrire autant d'emplacements de rupture de séquence qu'une ligne peut en contenir.

Exemple:

```
10 INPUT A
20 ON A GOSUB 100,200,300
30 GOTO 10
100 PRINT "SUB 100":RETURN
200 PRINT "SUB 200":RETURN
300 PRINT "SUB 300":RETURN
```

- * Si 1 à 3 est entré, une rupture de séquence vers le sous-programme correspondant est effectuée.

DATA [data] [, [data]]* (P)
 Constante Constante

Fonction:

Mise en mémoire des données.

Paramètre:

Donnée: Constante alphanumérique ou constante numérique.

Explication:

- 1) Utilisé pour écrire les données qui sont lues à l'aide de l'instruction READ.
- 2) Plusieurs données peuvent être écrites séparées par des "," de ponctuation.
- 3) Si une seule instruction DATA est exécutée sans instruction READ, aucune fonction n'est réalisée.

- 4) Si une constante alphanumérique comprend le signe ",", la placer entre guillemets " ".

DATA ABC, DEF, 'GHI,JKL',
1ère 2ème 3ème

- 5) Si une donnée est omise, une chaîne de caractères de longueur 0 est prise par défaut.

DATA A, B → DATA A,"",B
 DATA , → DATA "","" - | — Chaîne absence d'information.
 DATA → DATA "" —

- 6) Même si une commande est ajoutée à une instruction DATA avec deux points (:), elle sera considérée comme une donnée et non exécutée.

DATA 12,34,56: INPUT A

(Considéré comme une donnée.)

Un espace suivant une donnée n'est pas négligé. Par conséquent, si un espace est inséré après une donnée numérique, une erreur sera produite.

Exemple: DATA 1, 2 , 3, 4 ,
 Ces espaces ne sont pas négligés.

READ Nom de variable [, [nom de variable]]* (P)

Fonction:

Lecture du contenu d'une instruction DATA.

Paramètre:

Nom de variable: Variable numérique ou variable alphanumérique.
 On peut utiliser une variable de tableau.

Explication:

- 1) Affecte de manière séquentielle à une variable spécifiée, les données de l'instruction DATA présentement spécifiée.
- 2) Seules des données de type numérique peuvent être lues pour des variables numériques.
- 3) Les données contenues par des instructions DATA sont lues de manière séquentielle à commencer par le numéro de ligne le plus petit, et à partir du début dans une instruction.
- 4) Une fois les données nécessaires lues par une instruction READ, les données suivantes sont lues par l'instruction READ qui suit.
- 5) La première donnée dans une zone de programme où existe une instruction READ est lue par la première exécution de cette instruction, après quoi les données se trouvant à ce moment-là dans la zone de programme sont lues en séquence.
- 6) La spécification désignant la donnée à lire peut être modifiée à l'aide de l'instruction RESTORE.
- 7) Si le nombre des données dans une instruction DATA est plus petit que le nombre des variables dans une instruction READ, une erreur (Error 4) se produit.
- 8) Si un espace est inscrit au début d'une donnée, il est sauté.

Exemple:

```

10 DATA 1,2,3
20 READ A,B
30 PRINT A;B
40 DATA 4,5
50 READ C,D,E
60 PRINT C;D;E
70 END

```

* Les données sont lues en séquence à partir d'une instruction DATA et sont affichées.

RESTORE

[No. de ligne]
Expression numérique

Ⓟ

Fonction:

Spécifie l'emplacement des données lues par une instruction READ.

Paramètre:

No. de ligne: Expression numérique. Les chiffres après la virgule sont négligés.
 $1 \leq \text{No. de ligne} \leq 9999$

Explication:

- 1) Spécifie une instruction DATA à l'endroit où existe la donnée à lire par l'instruction READ.
- 2) Si le numéro de ligne est omis, la spécification donnée est annulée. Ensuite, les premières données de la zone de programme où existe une instruction READ sont spécifiées et lues par la première instruction READ qui est exécutée.
- 3) Si un numéro de ligne est spécifié dans la zone de programme par une instruction RESTORE, les données de l'instruction DATA correspondant à ce numéro de ligne sont lues en séquence par l'instruction READ.
- 4) Si le numéro de ligne spécifié n'existe pas ou si le numéro de ligne spécifié et suivant ne comprennent pas d'instructions DATA, une erreur (Error 4) se produit.

Exemple:

```

10 DATA 1,2,3
20 DATA 4,5
30 READ A,B,C,D,E
40 RESTORE 10
50 READ F,G
60 RESTORE 20
70 READ H,I
80 PRINT A;B;C;D;E;F;G;H;I
90 END

```

PUT

["Nom de fichier"] variable 1 [, Variable 2] *
Chaîne de caractères

Fonction:

Mise en mémoire des données sur une cassette.

Paramètre:

Nom de fichier: $1 \leq$ Nombre de caractères de la chaîne de caractères ≤ 8 . Facultatif.

Variable 1, variable 2: Spécifie la variable à mettre en mémoire.
Les variables de tableau sont également possibles.

Explication:

1) Mise en mémoire dans une cassette du contenu des variables.

2) Les spécifications des variables s'écrivent comme suit:

PUT A Contenu de la variable A.

PUT A,Z Contenu des 26 variables de A à Z.

PUT A(0), A(100) Contenu de 101 variables de A(0) à A(100).

PUT \$,D,W Contenu de la variable alphanumérique exclusive \$ et des 20 variables de D à W.

Si le contenu de la variable alphanumérique exclusive \$ doit être mis en mémoire, écrire \$ d'abord.

3) Si les variables de tableau sont définies en mode DIM, écrire de la manière suivante.

PUT A(5), A(9) Contenu de A(5), A(6), A(7), A(8), A(9)

PUT A(0,0,1), A(1,0,0) Contenu de A(0,0,1), A(0,0,2), A(0,1,0), A(0,1,1), A(0,1,2), A(0,2,0), A(0,2,1), A(0,2,2), et A(1,0,0) quand DIM A(2,2,2) est défini.

PUT A\$(*) Mise en mémoire du contenu de tous les tableaux de A\$. Sera PUT A\$(*) que le tableau soit à deux dimensions ou à trois dimensions. Il n'est pas possible de mettre deux variables de tableau ou plus en mémoire avec une instruction PUT.

* Si les variables de tableau sont définies en mode DEFM, écrire de la manière suivante.

PUT A,A(5) Contenu des six variables A à A(5). Même chose que PUT A,F.

PUT Z,Z(20) Contenu des 21 variables Z à Z(20).

4) Peut être exécuté manuellement comme dans un programme.

5) \$ n'est pas nécessaire même avec une variable alphanumérique.

Exemple:

Entrer PUT A,D quand A et C sont des variables numériques et B et D des variables alphanumériques.

PUT A,D

GET

["Nom de fichier"] variable 1 [, Variable 2] *
Chaîne de caractères

Fonction:

Chargement des données en mémoire dans une cassette sur une variable.

Paramètre:

Nom de fichier: $1 \leq$ Nombre de caractères de la chaîne de caractères ≤ 8 .
Facultatif.

Variable 1, variable 2: Spécifie la variable à mettre en mémoire.
Les variables de tableau sont également possibles.

Explication:

1) Chargement des données en mémoire dans une cassette sur une variable spécifiée.

2) Les spécifications des variables s'écrivent comme suit:

GET ACharge dans la variable A.

GET A,ZCharge dans les variables A à Z.

GET A(0), A(100)Charge dans les variables A(0) à A(100).

GET \$,D,WCharge dans la variable alphanumérique exclusive \$ et dans les 20 variables D à W.

3) Si les variables de tableau sont définies en mode DIM, écrire de la manière suivante.

GET A(5), A(9)Charge dans A(5), A(6), A(7), A(8), A(9).

GET A(0,0,1), A(1,0,0)Charge dans A(0,0,1), A(0,0,2), A(0,1,0), A(0,1,1), A(0,1,2), A(0,2,0), A(0,2,1), A(0,2,2), A(1,0,0) si DIM A(2,2,2) est défini.

GET A\$(*)Charge toutes les données A\$ de tableau mises en mémoire avec PUT A\$(*).

Quand le nombre d'éléments dans GET est plus grand que celui dans PUT, seules les données de PUT sont chargées. Inversement, quand les éléments dans GET sont moins nombreux, tous les éléments de GET sont chargés.

- * Des descriptions du type A\$(*) ne peuvent pas être utilisées en mode DEFM.
- * Si les variables de tableau sont définies en mode DEFM, écrire de la manière suivante.

GET A, A(5)Charge dans les six variables A à A(5). Même chose que GET A,F.

GET Z,Z(20)Charge dans les 21 variables Z à Z(20).

4) Le nom d'une variable mis en mémoire à l'aide de PUT peut être différent du nom lu par GET.

5) Si le nombre des données mises en mémoire est plus petit que le nombre de variables à charger, les données sont chargées en séquence dans les variables à partir de la première variable spécifiée.

6) Si un nom de fichier a été spécifié, les données chargées depuis la cassette sont celles correspondant au même nom de fichier. Si aucun nom de fichier n'est spécifié, les données seront chargées à partir de la première donnée trouvée sur la cassette.

7) Peut être exécuté manuellement comme dans un programme.

8) GET sera exécuté en mode DIM sans distinction entre les variables de caractère et les variables numériques. Une erreur se produira donc à l'exécution d'un tableau si une donnée mémorisée (PUT) sous forme de tableau de caractère est chargée (GET) dans un tableau numérique. Si une donnée mémorisée (PUT) sous forme de tableau numérique est chargée (GET) dans un tableau de caractère, le contenu du tableau de caractère est annulé.

<Exemple 1>

PUT A\$(*)

GET A(*)

↓

PRINT A(0) ← L'erreur 6 se produit dans cette ligne
(à l'exécution d'un tableau).

<Exemple 2>

PUT A(*)

GET A\$(*)

↓

PRINT A\$(0) ← L'affichage sera nul.

BEEP $\left[\begin{pmatrix} 0 \\ 1 \end{pmatrix} \right]$

Ⓐ

Fonction:

Déclenche un signal sonore.

Paramètre:

0: Son grave

1: Son aigu

0 est pris par défaut.

Explication:

- 1) Déclenche un signal sonore de tonalité grave ou aiguë.
- 2) Peut être exécuté manuellement comme dans un programme.

Exemple:

```
10 BEEP:INPUT "N=" ,N
20 FOR B=1 TO N
30 BEEP 0:BEEP 1:PRINT:PRINT B;
40 NEXT B
50 PRINT:GOTO 10
```

* Les signaux sonores retentissent autant de fois que spécifié.

DEFM $\left[\begin{array}{l} \text{Nombre de variables à ajouter} \\ \text{Expression numérique} \end{array} \right]$

Ⓐ

Fonction:

Commutation du mode DIM au mode DEFM.

Permet d'augmenter le nombre de variables.

Paramètre:

Nombre de variables
ajoutées:

Expression numérique. Les valeurs inférieures à la virgule décimale sont négligées. Facultatif.

$0 \leq \text{Nombre de variables ajoutées} \leq 1948$

Explication:

- 1) Permet d'augmenter le nombre de variables.
Les tableaux définis dans le mode DIM seront alors effacés.
- 2) Le nombre de variables ajoutées peut être spécifié en fonction du nombre d'octets disponibles.
- 3) La zone libre pour les programmes et la DATA BANK diminue de 8 octets pour chaque extension de variable.
- 4) Les variables ajoutées sont utilisées comme variables de tableau.
- 5) Si le nombre de variables ajoutées n'est pas spécifié après DEFM, le nombre des variables présentement spécifié est affiché.
- 6) Peut être exécuté manuellement comme dans un programme. Lors d'une exécution manuelle, le nouvel état spécifié (nombre de variables ajoutées + 26 variables de base) est affiché. Quand l'exécution est effectuée dans un programme, le nouvel état spécifié n'est pas affiché.
- 7) Une erreur (Error 1) se produit si l'extension des variables spécifiée est supérieure au nombre d'octets disponibles dans la zone libre.
- 8) L'extension spécifiée sera maintenue même après coupure de l'alimentation électrique. Pour annuler l'extension de variables et revenir aux 26 variables de base, effectuer DEFM 0.

Exemple:

```

DEFM 10 EXT  A ← Z:26 DEFM:10
DEFM EXT  A ← Z:26 DEFM:10

10 DEFM 10: CLEAR
20 FOR J=1 TO 10
30 PRINT "Z( "; J; " )=" ;
40 INPUT Z(J)
50 NEXT J
60 FOR J=1 TO 10
70 S=S+Z(J)
80 NEXT J
90 BEEP:PRINT " Sum=" ; S
100 DEFM 0
110 END

```

* Donne le total des 10 entrées de valeur numérique.

DIM

[Nom de tableau] [,Nom de tableau] *
A
Fonction:

Déclaration d'un tableau.

Paramètres:

Des tableaux en deux ou trois dimensions peuvent être spécifiés.

- 1) Nom de tableau (i) en une dimension
- 2) Nom de tableau (i,j) en deux dimensions
- 3) Nom de tableau (i,j,k) en trois dimensions
- 4) Passe du mode DEFM au mode DIM si le nom de tableau n'est pas attribué.
Les noms de tableau i, j, k peuvent être spécifiés par des expressions numériques: $0 \leq i, j, k \leq 225$. Les chiffres après la virgule ne sont pas considérés.
On peut également utiliser les lettres alphabétiques majuscules pour les noms de tableau.

Explication:

- 1) Déclaration des noms des variables de tableau.
- 2) Des tableaux de caractère peuvent être déclarés en ajoutant le signe \$ immédiatement après le nom de la variable de tableau. La capacité de mémorisation du nom d'un tableau de caractère est limitée à sept caractères. Si la chaîne de caractères comprend plus de sept caractères, tous les caractères suivants le septième caractère seront ignorés.
- 3) Le même nom de tableau peut être utilisé pour un nom de tableau numérique et pour un nom de tableau de caractère.
- 4) On peut déclarer jusqu'à huit tableaux. Une erreur 1 se produira si neuf tableaux ou plus sont déclarés.
- 5) L'erreur 5 se produira si la taille (i, j, k) de l'indice diffère pour un même nom de variable de tableau. Quand des variables de tableau de même taille sont déclarées, le contenu de toutes les variables de tableau deviendra 0 ou sera annulé.

Exemple:

```

10 DIM A(10)
20 A(3)=7
30 DIM A(10)
40 PRINT A(3) ← 0 est affiché.

```

- 6) L'erreur 6 se produira en cas de tentative d'utilisation d'une variable de tableau non déclarée en mode DIM.
- 7) L'erreur 1 se produit quand la mémoire est insuffisante.
- 8) Les variables de tableau multiples peuvent être déclarées en une seule fois par séparation par des virgules.

Exemple:

```
DIM A(3), A$(5), B(20,3)
```

- 9) Le contenu de toutes les variables de tableau sera réinitialisé à zéro ou chaîne nulle par exécution de l'instruction DIM.
- 10) Les déclarations de tableau peuvent être annulées avec CLEAR, ERASE, NEW ALL ou DEFM.
- * Si une instruction DIM est exécutée en mode DEFM, le mode DEFM est annulé ainsi que toutes les variables supplémentaires.
- 11) Chaque tableau déclaré avec une instruction DIM demande huit octets de mémoire.

ERASE

Nom de variable [, nom de variable]*

Ⓐ

Fonction:

Annule les variables de tableau.

Paramètre:

Spécifie les noms des variables de tableau.

Explication:

- 1) Annule les variables de tableau vérifiables à l'aide de LIST V.
- 2) Spécifie un nom de variable de tableau par une lettre alphabétique majuscule.

Exemple:

- ERASE A, A\$, B sera spécifié quand le nom de la variable affiché par LIST V est A(1), A\$(1,1,1), B(1).
- 3) Rien ne se produit si un nom de variable non enregistré est spécifié.
 - 4) Si la capacité de la mémoire s'affaiblit en raison des déclarations de tableau, utiliser la commande ERASE.

MODE

Expression numérique

Ⓟ

Fonction:

Détermine l'état de l'ordinateur.

Paramètre:

Expression numérique: Les valeurs décimales sont rejetées.
 $4 \leq \text{expression numérique} < 9$

Explication:

- 1) Détermine l'unité des angles et le mode PRINT ou libère ces modes selon l'expression numérique utilisée.
- 2) Les déterminations sont les suivantes.
MODE 4.....Détermine l'unité d'angle en degrés.
MODE 5.....Détermine l'unité d'angle en radians.
MODE 6.....Détermine l'unité d'angle en grades.
MODE 7.....Affiche "PRT ON" et détermine le mode PRINT.
MODE 8.....Libère le mode PRINT.
- 3) Mêmes déterminations qu'avec la touche . Toutefois, le mode RUN et le mode WRT ne peuvent pas être déterminés à l'aide de cette commande. De plus, les entrées ne peuvent pas être effectuées à l'aide de la touche mais en appuyant sur les touches .

Exemple:

```
10 MODE 4
20 A=SIN 30:PRINT"A=";A
30 MODE 5
40 B=COS(π/6):PRINT"B=";B
50 END
```

* Pendant l'exécution du programme le symbole "DEG" sur le visuel devient le symbole "RAD".

STAT CLEAR

Ⓐ

Fonction:

Initialise les statistiques de base.

Explication:

- 1) Efface le contenu des mémoires utilisées pour les calculs statistiques; n (nombre de données), Σx (somme des données x), Σy (somme des données y), Σx^2 (somme du carré des données x), Σy^2 (somme du carré des données y) et Σxy (somme du produit des données x et y).
- 2) Initialise les statistiques de base pour commencer un nouveau calcul. Toujours exécuter cette commande pour de nouveaux calculs statistiques.

Exemple:

STAT CLEAR

STAT

valeur de donnée X [, valeur de donnée Y] [;fréquence]]

Ⓐ

Fonction:

Introduit les données statistiques et la fréquence de ces données.

Paramètre:Valeur de donnée x : expression numériqueValeur de donnée y : expression numérique

Fréquence: expression numérique

Explication:

- 1) Introduit les données statistiques et leur fréquence dans la mémoire statistiques.
- 2) Si la valeur de la donnée x n'est pas spécifiée, la valeur précédente sera attribuée à x .

- 3) Si la valeur de la donnée y n'est pas spécifiée, la valeur précédente sera attribuée à y .
- 4) Si la fréquence n'est pas spécifiée, la valeur un sera prise par défaut.

STAT LIST

$$\left[\begin{array}{c} 0 \\ 1 \\ 2 \end{array} \right]$$

Ⓐ

Fonction:

Fournit la quantité de statistiques.

Paramètre:

0 Indique toutes les statistiques.

1 Indique les statistiques pour une variable.

2 Indique les statistiques pour les variables appariées.

Sans indication Indique toutes les statistiques. (Identique à STAT LIST 0.)

Explication:

- 1) Indique les statistiques calculées sur la base des données STAT LIST, STAT LIST 0, STAT LIST 1 ou STAT LIST 2 introduites.
- 2) Voir en pages 37 et 38 les statistiques qui seront fournies et les formules de calcul.

Exemple:

```

10 STAT CLEAR: CLEAR
20 FOR N=1 TO 5
30 READ X,Y
40 STAT X,Y
50 NEXT N
60 STAT LIST
70 END
1000 DATA 3,5,4,9,2,1,6,4,6,9

```

SET $\left\{ \begin{array}{c} Fn \\ En \\ N \end{array} \right\}$

Ⓐ

★ n est un nombre entier entre 0 et 9.**Fonction:**

Spécifie le format de sortie des données.

Paramètre: F_n : Spécifie le nombre de chiffres après la virgule. E_n : Spécifie le nombre de chiffres significatifs. N : Libère la spécification.**Explication:**

- 1) Spécifie le nombre de chiffres après la virgule ou le nombre de chiffres significatifs.
- 2) On utilise une valeur de 0 à 9 pour spécifier le nombre de chiffres après la virgule (F_n).
- 3) On utilise une valeur de 0 à 9 pour spécifier le nombre de chiffres significatifs (E_n). "SET E0" indique une spécification à 10 chiffres.
- 4) Les deux spécifications sont libérées par "SET N".
- 5) Après exécution de cette commande, les chiffres spécifiés sont arrondis.
- 6) Le nombre de chiffres spécifié par cette commande est pour la visualisation uniquement et la mantisse à 12 chiffres peut rester dans l'ordinateur.
- 7) Peut être exécuté manuellement comme dans un programme.

Exemple:

```

10 X=10*SQR2
20 SET F3:PRINT X
30 SET E3:PRINT X
40 SET N:PRINT X
50 END

```

LEN

(Variable alphanumérique simple)

Ⓕ

Fonction:

Permet d'obtenir la longueur d'une chaîne de caractères dans une variable alphanumérique simple.

Paramètre:

Variable alphanumérique simple: On ne peut pas utiliser de variable alphanumérique de tableau.

Explication:

- 1) Compte le nombre de caractères dans une variable simple.
- 2) Les variables alphanumériques utilisables sont des variables alphanumériques simples (A\$, Y\$, \$, etc.). Les variables alphanumériques de tableau du type B\$(3) ne peuvent pas être utilisées.

Exemple:

```
10 INPUT ^String=",$
20 PRINT ^Length=";LEN($ )
30 GOTO 10
```

MID\$

(Emplacement [, Nombre de caractères])
Expression numérique Expression numérique

Ⓕ

Fonction:

Reprend en mémoire le nombre de caractères spécifié depuis l'emplacement spécifié de la variable alphanumérique exclusive (\$).

Paramètre:

Emplacement: Expression numérique. Les valeurs décimales sont rejetées.
 $1 \leq \text{emplacement} < 101$

Nombre de caractères: Expression numérique. Les valeurs décimales sont rejetées.
 $1 \leq \text{nombre de caractères} < 101$

Si rien n'est spécifié, tous les caractères après l'emplacement spécifié sont repris en mémoire.

Explication:

- 1) Reprend en mémoire un nombre spécifié de caractères depuis l'emplacement spécifié de la variable alphanumérique exclusive (\$).
 - 2) Si l'emplacement spécifié n'est pas dans la chaîne de caractères, on obtient une absence d'information.
 - 3) Si la longueur de la chaîne de caractères après l'emplacement spécifié est plus petite que le nombre de caractères spécifié, tous les caractères après l'emplacement spécifié sont repris en mémoire.
- * MID\$ peut être abrégé par MID.

Exemple:

```
10 $="ABCDEFGHIJKLMNOPQRSTUVWXYZ"
20 INPUT ^location=" ,M, ^length=" ,N
30 PRINT MID$(M,N)
40 GOTO 20
```

VAL

(Variable alphanumérique simple)

**Fonction:**

Convertit en valeur numérique les caractères dans une variable alphanumérique simple.

Paramètre:

Variable alphanumérique simple:

On ne peut pas utiliser de variable alphanumérique de tableau.

Explication:

- 1) Convertit en valeur numérique les caractères dans une variable alphanumérique simple.
- 2) Si le contenu d'une variable alphanumérique comprend +, -, *, E ou E⁻, elle est convertie telle que en valeur numérique.
Si A\$ = "-12,3", VAL(A\$) → -12,3
- 3) Si le contenu d'une variable alphanumérique commence par un caractère autre qu'un nombre, +, - ou *, une erreur se produit.
Si A\$ = "A45", VAL(A\$) → erreur (Erreur 2)
- 4) Si un caractère autre qu'un nombre est inséré au milieu de la variable, seule la partie précédant ce caractère est convertie en valeur numérique.
Si A\$ = "78A9", VAL(A\$) → 78

Exemple:

```
10 Z$=" 123"
20 PRINT VAL(Z$)+45
30 END
```

* Si ce programme est exécuté, les valeurs numériques 123 et 45 seront ajoutées et 168 sera affiché.

STR\$

(Expression numérique)

**Fonction:**

Convertit la valeur d'une expression numérique en une chaîne de caractères.

Paramètre:

Expression numérique: Valeur numérique, expression de calcul, variable numérique, variable de tableau numérique.

Explication:

- 1) Convertit la valeur d'une expression numérique en une chaîne de caractères.
- 2) Si l'expression numérique est une expression de calcul, le résultat du calcul est converti en une chaîne de caractères.
- 3) Si l'expression numérique est positive, le chiffre signe est annulé et seuls les nombres sont convertis.

Exemple:

```
10 A=123
20 PRINT STR$(A)+"45"
30 END
```

* Si ce programme est exécuté, les chaînes de caractères "123" et "45" seront connectées et "12345" sera affiché. (Comme pour "123" + "45".)

7-4 Fonctions numériques

SIN	<u>Argument</u> Expression numérique	COS	<u>Argument</u> Expression numérique	Ⓣ
TAN	<u>Argument</u> Expression numérique			

Fonction:

Fournit la valeur d'une fonction trigonométrique pour un argument donné.

Paramètre:

Argument: Expression numérique
 $-1440^\circ < \text{argument} < 1440^\circ$ (degrés)
 $-8\pi < \text{argument} < 8\pi$ (radians)
 $-1600 < \text{argument} < 1600$ (grades)
 Toutefois, pour TAN, " $|\text{Argument}| = (2n - 1) * 1$ angle droit" est exclu.
 $1 \text{ angle droit} = 90^\circ = \frac{\pi}{2} \text{ radians} = 100 \text{ grades}$

Explication:

- 1) Fournit la valeur d'une fonction trigonométrique pour un argument donné.
- 2) La valeur dépend de l'unité déterminée pour l'angle; DEG, RAD ou GRA (à l'aide de la touche  ou de la commande MODE).
- 3) En règle générale, l'argument est placé entre parenthèses mais celles-ci sont facultatives si l'argument est une variable ou une valeur numérique.

7-4 Fonctions numériques

ASN	<u>Argument</u> Expression numérique	ACS	<u>Argument</u> Expression numérique	Ⓣ
ATN	<u>Argument</u> Expression numérique			

Fonction:

Fonction trigonométrique inverse qui donne un angle pour un argument donné.

Paramètre:

Argument: Expression numérique
 Pour ASN, ACS, $-1 \leq \text{argument} \leq 1$

Explication:

- 1) Fonction trigonométrique inverse qui donne un angle pour un argument donné.
- 2) La valeur dépend de l'unité déterminée pour l'angle (à l'aide de la touche  ou de la commande MODE).
- 3) Les valeurs des fonctions sont données dans les limites suivantes.

Degrés (DEG)	Radians (RAD)	Grades (GRA)
$-90^\circ \leq \text{ASN } x \leq 90^\circ$	$-\frac{\pi}{2} \leq \text{ASN } x \leq \frac{\pi}{2}$	$-100 \leq \text{ASN } x \leq 100$
$0^\circ \leq \text{ACS } x \leq 180^\circ$	$0 \leq \text{ACS } x \leq \pi$	$0 \leq \text{ACS } x \leq 200$
$-90^\circ \leq \text{ATN } x \leq 90^\circ$	$-\frac{\pi}{2} \leq \text{ATN } x \leq \frac{\pi}{2}$	$-100 \leq \text{ATN } x \leq 100$

- 4) En règle générale, l'argument est placé entre parenthèses mais celles-ci sont facultatives si l'argument est une variable ou une valeur numérique.

HYPSIN	<u>Argument</u> Expression numérique
HYPCOS	<u>Argument</u> Expression numérique
HYPTAN	<u>Argument</u> Expression numérique

Ⓕ

Fonction:

Fournit la valeur d'une fonction hyperbolique pour un argument donné.

Paramètre:

Argument: Expression numérique
 $|\text{argument}| \leq 230,2585092$ (HYPSIN, HYPCOS)
 $|\text{argument}| < 10^{100}$ (HYPTAN)

Explication:

1) Fournit la valeur d'une fonction hyperbolique pour un argument donné.

$$\text{HYPSIN} : \sinh x = (e^x - e^{-x}) / 2$$

$$\text{HYPCON} : \cosh x = (e^x + e^{-x}) / 2$$

$$\text{HYPTAN} : \tanh x = (e^x - e^{-x}) / (e^x + e^{-x})$$

2) En règle générale, l'argument est placé entre parenthèses mais celles-ci sont facultatives si l'argument est une variable ou une valeur numérique.

HYPASN	<u>Argument</u> Expression numérique
HYPACS	<u>Argument</u> Expression numérique
HYPATN	<u>Argument</u> Expression numérique

Ⓕ

Fonction:

Fournit la valeur d'une fonction hyperbolique inverse pour un argument donné.

Paramètre:

Argument: Expression numérique
HYPASN: $|\text{argument}| < 5 \times 10^{99}$
HYPACS: $1 \leq \text{argument} < 5 \times 10^{99}$
HYPATN: $|\text{argument}| < 1$

Explication:

1) Fournit la valeur d'une fonction hyperbolique inverse pour un argument donné.

$$\text{HYPASN} : \sinh^{-1} x = \log_e (x + \sqrt{x^2 + 1})$$

$$\text{HYPACS} : \cosh^{-1} x = \log_e (x + \sqrt{x^2 - 1})$$

$$\text{HYPATN} : \tanh^{-1} x = \frac{1}{2} \log_e \frac{1+x}{1-x}$$

2) En règle générale, l'argument est placé entre parenthèses mais celles-ci sont facultatives si l'argument est une variable ou une valeur numérique.

LOG Argument
 Expression numérique **LN** Argument ⑤
 Expression numérique

Fonction:

Fournit la valeur d'une fonction logarithmique pour un argument donné.

Paramètre:

Argument: Expression numérique
 $0 < \text{argument}$

Explication:

- 1) Fournit la valeur d'une fonction logarithmique pour un argument donné.
 - **LOG Fonction logarithmique commune** $\log_{10}x$, $\log x$
 - **LN Fonction logarithmique naturelle** $\log_e x$, $\ln x$
- 2) En règle générale, l'argument est placé entre parenthèses mais celles-ci sont facultatives si l'argument est une variable ou une valeur numérique.

EXP Argument
 Expression numérique ⑤

Fonction:

Fournit la valeur d'une fonction exponentielle pour un argument donné.

Paramètre:

Argument: Expression numérique
 $-10^{100} < \text{argument} \leq 230,2585092$

Explication:

- 1) Fournit la valeur d'une fonction exponentielle (e^x) pour un argument donné.
- 2) En règle générale, l'argument est placé entre parenthèses mais celles-ci sont facultatives si l'argument est une variable ou une valeur numérique.

SQR Argument
 Expression numérique ⑤

Fonction:

Fournit la racine carrée pour un argument donné.

Paramètre:

Argument: Expression numérique. $0 \leq \text{argument}$

Explication:

- 1) Fournit la racine carrée ($\sqrt{x}$) pour un argument donné.
- 2) En règle générale, l'argument est placé entre parenthèses mais celles-ci sont facultatives si l'argument est une variable ou une valeur numérique.

CUR Argument
 Expression numérique ⑤

Fonction:

Fournit la racine cubique ($\sqrt[3]{x}$ ou $x^{\frac{1}{3}}$) pour un argument donné.

Paramètre:

Argument: Expression numérique. $|\text{argument}| < 10^{100}$

Explication:

- 1) Fournit la racine cubique ($\sqrt[3]{x}$ ou $x^{\frac{1}{3}}$) pour un argument donné.
- 2) En règle générale, l'argument est placé entre parenthèses mais celles-ci sont facultatives si l'argument est une variable ou une valeur numérique.

ABS

Argument
Expression numérique

Ⓕ

Fonction:

Fournit la valeur absolue pour un argument donné.

Paramètre:

Argument: Expression numérique

Explication:

- 1) Fournit la valeur absolue pour un argument donné.
- 2) En règle générale, l'argument est placé entre parenthèses mais celles-ci sont facultatives si l'argument est une variable ou une valeur numérique.

SGN

Argument
Expression numérique

Ⓕ

Fonction:

Fournit une valeur (1,0 ou -1) correspondant au signe d'un argument donné.

Paramètre:

Argument: Expression numérique

Explication:

- 1) Fournit une valeur qui correspond au signe de l'argument.
 Si l'argument est positif, 1
 Si l'argument est 0, 0
 Si l'argument est négatif, -1
- 2) En règle générale, l'argument est placé entre parenthèses mais celles-ci sont facultatives si l'argument est une variable ou une valeur numérique.

INT

Argument
Expression numérique

Ⓕ

Fonction:

Fournit l'entier maximum qui n'excède pas l'argument donné.

Paramètre:

Argument: Expression numérique

Explication:

- 1) Fournit l'entier maximum qui n'excède pas l'argument donné.

```
INT 12.56 → 12
INT -78.1 → -79
INT 12 → 12
```

- 2) Cette fonction est identique à la fonction de Gauss ($[x]$) utilisée en mathématiques.
- 3) En règle générale, l'argument est placé entre parenthèses mais celles-ci sont facultatives si l'argument est une variable ou une valeur numérique.

FRAC

Argument
Expression numérique

**Fonction:**

Fournit la partie décimale d'un argument donné.

Paramètre:

Argument: Expression numérique

Explication:

- 1) Fournit la partie décimale d'un argument donné. Son signe correspond au signe de l'argument.
- 2) En règle générale, l'argument est placé entre parenthèses mais celles-ci sont facultatives si l'argument est une variable ou une valeur numérique.

RND

(Argument , Emplacement de chiffre)
Expression numérique Expression numérique

**Fonction:**

Fournit la valeur d'un argument donné arrondi à l'emplacement spécifié.

Paramètre:

Argument: Expression numérique
Emplacement: Expression numérique. Les valeurs après la virgule sont rejetées. Affiche l'exposant n quand l'emplacement à arrondir est 10^n .
 $|\text{emplacement}| < 100$

Explication:

- 1) Fournit la valeur d'un argument donné arrondi à l'emplacement spécifié.
- 2) Si RND (x , y) est exécuté, x sera arrondi à 10^y .
Arrondi à la troisième décimale (10^{-3}) $\rightarrow$ RND (x , -3)
Arrondi à la 100ème position (10^2) $\rightarrow$ RND (x , 2)
- 3) Les parenthèses sont obligatoires.

REC

(coordonnée r , coordonnée θ)
Expression numérique

**Fonction:**

Transforme des coordonnées polaires (r, θ) en coordonnées rectangulaires (x, y).

Paramètre:

Coordonnée r : Expression numérique. $0 \leq r < 10^{100}$
Coordonnée θ : Expression numérique
DEG: $|\theta| < 1440^\circ$
RAD: $|\theta| < 8\pi \text{ rad}$
GRA: $|\theta| < 1600 \text{ gra}$

Explication:

- 1) Transforme des coordonnées polaires (r, θ) en coordonnées rectangulaires (x, y) à l'aide des égalités suivantes: $x = r \cos \theta$
 $y = r \sin \theta$
- 2) La coordonnée x de (x, y) sera donnée comme sortie de la fonction et sa valeur sera assignée à la variable X, alors que la valeur de la coordonnée y sera assignée à la variable Y.
- 3) Une erreur se produit si $r < 0$.
- 4) Les valeurs de sortie correspondent à l'unité de l'angle déterminée (DEG, RAD ou GRA).

Exemple:

```
10 CLEAR
20 MODE 4
30 INPUT "r = ", R
40 INPUT "t (°) = ", T
50 X=REC(R, T)
60 PRINT "x="; X
70 PRINT "y="; Y
80 GOTO 30
```

* (x, y) sera affiché si (r, θ) est entré.

POL (coordonnée x, coordonnée y) (F)
 Expression numérique Expression numérique

Fonction:

Transforme des coordonnées rectangulaires (x,y) en coordonnées polaires (r,θ).

Paramètre:

Coordonnée x: Expression numérique }
 Coordonnée y: Expression numérique } $|x| + |y| > 0$

Explication:

- 1) Transforme des coordonnées rectangulaires (x,y) en coordonnées polaires (r,θ) à l'aide des égalités suivantes:

$$\begin{cases} r = \sqrt{x^2 + y^2} \\ \cos \theta = \frac{x}{\sqrt{x^2 + y^2}}, \sin \theta = \frac{y}{\sqrt{x^2 + y^2}} \end{cases}$$

- 2) La coordonnée r de (r,θ) sera donnée comme sortie de la fonction et sa valeur sera assignée à la variable X, alors que la valeur de la coordonnée θ sera assignée à la variable Y.
- 3) Le calcul de la coordonnée θ correspond à l'unité de l'angle (DEG, RAD ou GRA). La valeur de θ est dans les limites suivantes:
 DEG: $-180^\circ < \theta \leq 180^\circ$
 RAD: $-\pi \text{ rad} < \theta \leq \pi \text{ rad}$
 GRA: $-200 \text{ gra} < \theta \leq 200 \text{ gra}$
- 4) Une erreur se produit si $|x| + |y| = 0$.

Exemple:

```
10 CLEAR
20 MODE 4
30 INPUT "X=" , A
40 INPUT "Y=" , B
50 PRINT "r =" ; POL ( A , B )
60 PRINT "t =" ; DMS $ ( Y )
70 GOTO 30
```

* (r,θ) sera affiché si (x,y) est entré.

FACT Argument (F)
 Expression numérique

Fonction:

Fournit la valeur factorielle d'un argument donné.

Paramètre:

Argument: Expression numérique
 L'argument doit prendre une valeur entière se trouvant entre $(0 \leq \text{argument} \leq 69)$.

Explication:

- 1) Fournit la valeur factorielle (x!) de l'argument x.
- 2) Une erreur se produit si l'argument x contient des fractions.
- 3) En règle générale, l'argument est placé entre parenthèses mais celles-ci sont facultatives si l'argument est une variable ou une valeur numérique.

Exemple:

```
FACT 10 (10)
FACT 69 (69)
```

```
3628800
1.711224524E98
```

NPR

(Valeur de n , valeur de r)
 Expression numérique Expression numérique

Ⓔ

Fonction:Fournit les permutations (nPr) pour les valeurs données de n et r .**Paramètre:**

Valeur de n : Expression numérique } n et r sont des nombres
 Valeur de r : Expression numérique } entiers se trouvant entre
 $0 \leq r \leq n \leq 10^{10}$

Explication:

- 1) Cette fonction fournit les permutations $nPr (= \frac{n!}{(n-r)!})$
- 2) Une erreur se produit si n ou r contiennent des fractions.

Exemple:NPR (5 , 3) $\boxed{\text{EXE}}$ (${}_5P_3$)

60

NPR (5 , 0) $\boxed{\text{EXE}}$ (${}_5P_0$)

1

NCR

(Valeur de n , valeur de r)
 Expression numérique Expression numérique

Ⓔ

Fonction:Fournit les combinaisons (nCr) pour les valeurs données de n et r .**Paramètre:**

Valeur de n : Expression numérique } n et r sont des nombres
 Valeur de r : Expression numérique } entiers se trouvant entre
 $0 \leq r \leq n < 10^{10}$

Explication:

- 1) Cette fonction fournit les combinaisons $nCr (= \frac{n!}{r!(n-r)!})$.
- 2) Une erreur se produit si n ou r contiennent des fractions.

Exemple:NCR (5 , 3) $\boxed{\text{EXE}}$ (${}_5C_3$)

10

NCR (5 , 0) $\boxed{\text{EXE}}$ (${}_5C_0$)

1

EOXArgument
Expression numérique

Ⓕ

Fonction:

Fournit la valeur estimée de x pour une valeur donnée de y comme argument dans les statistiques de variable appariée de (x,y) .

Paramètre:

Argument: Expression numérique donnant la valeur de y .

Explication:

- 1) Fournit la valeur estimée de x pour une valeur donnée de y conformément à l'expression de régression linéaire $y = a + bx$ dans les statistiques de variable appariée de (x,y) .

$$EOX(y) = \frac{y - a}{b}$$

- 2) Les valeurs du terme constant a de la régression linéaire et du coefficient de régression linéaire b sont déterminées par les données statistiques.
- 3) La valeur de $EOX(y)$ sera incertaine quand $b = 0$. La valeur y introduite est dans les limites de $|y| < 10^{100}$ et la valeur estimée de x dans les limites de $|EOX(y)| < 10^{100}$ quand $b \neq 0$.
- 4) En règle générale, l'argument est placé entre parenthèses mais celles-ci sont facultatives si l'argument est une variable ou une valeur numérique.

EOYArgument
Expression numérique

Ⓕ

Fonction:

Fournit la valeur estimée de y pour une valeur donnée de x comme argument dans les statistiques de variable appariée de (x,y) .

Paramètre:

Argument: Expression numérique donnant la valeur de x .

Explication:

- 1) Fournit la valeur estimée de y pour une valeur donnée de x conformément à l'expression de régression linéaire $y = a + bx$ dans les statistiques de variable appariée de (x,y) .

$$EOY(x) = a + b x$$

- 2) Les valeurs du terme constant a de la régression linéaire et du coefficient de régression linéaire b sont déterminées par les données statistiques.
- 3) La valeur x introduite est dans les limites de $|x| < 10^{100}$ et la valeur estimée de y dans les limites de $|EOY(x)| < 10^{100}$.
- 4) En règle générale, l'argument est placé entre parenthèses mais celles-ci sont facultatives si l'argument est une variable ou une valeur numérique.

RAN

Ⓢ

Fonction:

Fournit un nombre au hasard entre 0 et 1.

Explication:

- 1) Fournit un nombre au hasard entre 0 et 1 pris dans une mantisse de 10 chiffres.
 $0 < \text{nombre au hasard} < 1$

Exemple:

Fournit un nombre à 1 chiffre au hasard entre 0 et 9.

$$\text{INT}(\text{RAN\#} * 10)$$

Fournit un nombre à 1 chiffre au hasard entre 1 et 6.

$$\text{INT}(\text{RAN\#} * 6) + 1$$

Fournit un nombre à 2 chiffres au hasard entre 10 et 99.

$$\text{INT}(\text{RAN\#} * 90) + 10$$

DEG (Degré [,Minute [,Seconde]])

Ⓢ

Expression numérique Expression numérique Expression numérique

Fonction:

Convertit une sexagésimale exprimée en degrés, minutes, secondes en une décimale.

Paramètre:

Degré: Expression numérique
 Minute: Expression numérique
 Seconde: Expression numérique

$$|\text{DEG}(\text{degree, minute, second})| < 10^{100}$$
Explication:

- 1) Convertit une sexagésimale exprimée en degrés, minutes, secondes en une décimale.

$$\text{DEG}(a, b, c) = a + \frac{b}{60} + \frac{c}{3600}$$

- 2) Les minutes et les secondes sont facultatives et seront alors considérées comme égales à 0.

- 3) Les parenthèses sont obligatoires.

Exemple:

DEG(12, 34, 56) Ⓢ

12.58222222

```
10 INPUT A,B,C
20 PRINT DEG(A,B,C)
30 END
```

DMS \$

(Argument)
Expression numérique

Ⓕ

Fonction:

Convertit un argument décimal donné en une chaîne de caractères en notation sexagésimale.

Paramètre:

Argument: Expression numérique
|Expression numérique| < 10¹⁰⁰

Explication:

- 1) Convertit un argument décimal donné en une chaîne de caractères en notation sexagésimale.
- 2) Bien que degrés, minutes et secondes soient affichés dans les limites de |expression numérique| < 10⁵, la valeur de l'expression numérique elle-même sera affichée même si elle est hors limites.
- 3) Le résultat sera donnée sous forme d'une chaîne de caractères.

Exemple:

```
DMS$(180/π) [EX]
DMS$(45.678) [EX]
DMS$(99999.999) [EX]
DMS$(100000.1) [EX]
```

```
57°17'44.81
45°40'40.8
99999°59'56.4
100000.1
```

```
10 INPUT A
20 $=DMS$(A)
30 PRINT $
40 END
```

HEX \$

(Argument)
Expression numérique

Ⓕ

Fonction:

Convertit un argument décimal donné en une chaîne de caractères en notation hexadécimale à 4 chiffres.

Paramètre:

Argument: Expression numérique
-32769 < expression numérique < 65536

Explication:

- 1) Convertit un argument décimal donné en une chaîne de caractères en notation hexadécimale à 4 chiffres.
- 2) La valeur de l'expression numérique donnée comme argument est traitée comme un nombre entier, les chiffres après la virgule étant ignorés.
- 3) Si la valeur de l'argument excède 32768, elle sera traitée après soustraction de 65536.

< Exemple >

40000 sera traité comme suit:
40000 - 65536 = -25536

Exemple:

```
HEX$(10000) [EX]
HEX$(65535) [EX]
HEX$(-1) [EX]
```

```
2710
FFFF
FFFF
```

```
10 INPUT "X=",X
20 PRINT "X=&H";HEX$(X)
30 GOTO 10
```

&H

Chaîne de caractères

Ⓐ

Fonction:

Cette fonction, lorsqu'elle est placée au début d'une chaîne de caractères hexadécimale, convertit cette dernière en une valeur décimale.

Paramètre:

Chaîne de caractères: Chaîne numérique hexadécimale.
(Jusqu'à 4 chiffres)
– 32768 ≤ &H chaîne de caractères ≤ 32767

Explication:

- 1) Cette fonction, lorsqu'elle est placée au début d'une chaîne de caractères hexadécimale, convertit cette dernière en une valeur entière décimale. Bien que &H soit considérée comme une fonction, elle ne l'est pas à strictement parler puisqu'elle fonctionne de manière opposée à HEX\$ et est en fait un identificateur de notation hexadécimale.
- 2) Le résultat de la conversion étant un entier décimal entre – 32768 et + 32767, &HFFFF, par exemple, n'indiquera pas 65535 mais – 1.
- 3) Le 0 placé en haut d'une notation hexadécimale et les espaces dans une chaîne de caractères en notation hexadécimale sont ignorés.

< Exemple >

&H0010 indique 16 d'un nombre décimal.
&HA□B indique 171 d'un nombre décimal.
&H□A indique 10 d'un nombre décimal.
(□ représente un espace)

- 4) Une erreur (Error 2) se produira si la chaîne de caractères à notation hexadécimale excède quatre chiffres ou si un caractère est écrit dans une notation autre qu'hexadécimale dans la chaîne.

< Exemple >

&H10000 → Error 2 (Cinq chiffres)
&HAG → Error 2 (G n'est pas hexadécimal)
&HA# → Error 2 (# n'est pas hexadécimal)

Exemple:

```

10 CLEAR
20 READ H$
30 IF H$="end" THEN 80
40 M$="&H"+H$
50 A=VAL(M$)
60 PRINT M$;"=";A;"□:□";
70 GOTO 10
80 PRINT H$::BEEP:END
100 DATA 10,100,1000,7FFF
110 DATA 8000,ABCD,FFFF
120 DATA end

```

Espace

* Convertit les hexadécimales en décimales.

NEW#

(M)

Fonction:

Efface les données en mémoire dans la DATA BANK.

Explication:

- 1) Efface toutes les données en mémoire.
- 2) Ne peut pas être exécuté quand un mot de passe est spécifié.
- 3) Ne peut être exécuté qu'en mode WRT.

Exemple:

MODE 1
NEW # EXE
MODE 2

LIST#

(M)

Fonction:

Affiche toutes les données en mémoire dans la DATA BANK.

Paramètre:

Sort toutes les données en mémoire dans la DATA BANK avec le numéro d'enregistrement.

Explication:

- 1) Affiche toutes les données en mémoire dans la DATA BANK à partir du numéro d'enregistrement le plus petit dans l'ordre de mise en mémoire.

- 2) Les contenus affichés sont le numéro d'enregistrement et la donnée en mémoire.
- 3) Les données en mémoire étant affichées automatiquement en séquence, appuyer sur la touche [EXE] pour arrêter l'affichage. Appuyer de nouveau sur la touche [EXE] pour reprendre l'affichage.
- 4) En mode PRT ON (MODE 7), les données en mémoire sont successivement affichées à grande vitesse et imprimées en même temps.
- 5) L'exécution n'est pas possible si un mot de passe a été spécifié. (Une erreur se produit).
- 6) Ne peut pas être utilisé dans un programme.
- 7) Ne peut pas être exécuté en mode MEMO IN (MODE 9).

Exemple:

LIST# EXE

— L I S T # —

```
1 510-01, TELEVISION, . $330
2 510-02, RADIO, . $80
3 510-03, TAPE RECORDER, . $100
4 510-04, STEREO, . $850
5 510-05, VIDEO RECORDER, . $750
6 510-06, DESKTOP CALCULATOR, . $50
7 510-07, PERSONAL COMPUTER, . $650
8 END
```

SAVE# ["Nom de fichier"]
 Chaîne de caractères

**Fonction:**

Met en mémoire sur cassette des données mémoire de la DATA BANK.

Paramètre:

Nom de fichier: $1 \leq \text{nombre de caractères de la chaîne} \leq 8$.
 Facultatif.

Explication:

- 1) Met en mémoire sur cassette des données mémoire de la DATA BANK.
- 2) Les données mémoire ne pouvant pas être mises en mémoire à l'aide de SAVE ou SAVE ALL, toujours charger les données mémoire à l'aide de SAVE#.
- 3) Si un mot de passe a été spécifié, la mise en mémoire est effectuée avec ce mot de passe. Par conséquent, le même mot de passe doit être spécifié quand le chargement est effectué à partir de la commande LOAD#.
- 4) Ne peut pas être exécutée en mode MEMO IN.

Exemple:

SAVE# EXE
 SAVE# "CASIO" EXE

LOAD# ["Nom de fichier"]L,M
 Chaîne de caractères

**Fonction:**

Charge les données mémoire dans la DATA BANK depuis une cassette.

Paramètre:

Nom de fichier: $1 \leq \text{nombre de caractères de la chaîne} \leq 8$.
 Facultatif.

M: (Si M est spécifié, des données mémoire supplémentaires peuvent être chargées.)

Explication:

- 1) Cas de LOAD# ["Nom de fichier"]
 - a) Charge les données mémoire dans la DATA BANK depuis une cassette après effaçage des données mémoire se trouvant présentement dans la DATA BANK.
 - b) Si le nom de fichier n'est pas spécifié, les premières données mémoire trouvées sur la cassette sont chargées d'abord.
 - c) Ne peut pas être exécuté en mode MEMO IN.
 - d) Ne peut pas être exécuté dans un programme.
- 2) Cas de LOAD# ["Nom de fichier"],M
 - a) Des données mémoire supplémentaires sont chargées dans la DATA BANK à la suite de celles s'y trouvant déjà.
 - b) à d) comme pour LOAD#.

Exemple:

LOAD# EXE
 LOAD# "CASIO" EXE

READ#

Nom de variable [, nom de variable]*

P

Fonction:

Lit les données mémoire dans la DATA BANK.

Paramètre:

Nom de variable: Variable numérique ou variable alphanumérique.
On peut aussi utiliser une variable de tableau.

Explication:

- 1) Lit en séquence les données en mémoire sur une variable.
- 2) Seules des données de type numérique peuvent être lues pour une variable numérique. Si on utilise des données de type alphanumérique une erreur (Error 2) se produit.
- 3) Quand les données nécessaires ont été lues par une instruction READ#, les données suivantes sont lues par l'instruction READ# suivante.
- 4) Si une ponctuation de type "," a été introduite dans les données mémoire dans la DATA BANK, elles sont lues dans l'ordre dans lequel elles ont été écrites.

<Exemple>**DONNEES**

No. 1 A, X, Y

No. 2 B, Z

No. 3 C



Séquence de lecture

A → X → Y → B → Z → C

- 5) Si les données à lire n'existent pas, une erreur (Error 4) se produit.
- 6) La séquence de lecture des données peut être modifiée par RESTORE# (voir page 212).

- 7) Si un espace a été introduit au début des données mémoire dans la DATA BANK, il est sauté.

<Exemple>

X, Y, Z

Cet espace est sauté.

- 8) Si une donnée est placée entre guillemets " ", la chaîne de caractères se trouvant entre les guillemets " " est lue.

Exemple:**<Données>**

No. 1 1, 2, 3

No. 2 4, 5, 6

No. 3 7, 8, 9

No. 4 10,

<Programme>

10 A=0

20 READ#\$

30 IF \$=" " THEN 60

40 A=A+VAL(\$)

50 GOTO 20

60 PRINT "Σx="; A

70 END

- * Lecture des données numériques dans la DATA BANK pour le calcul d'une somme.

RESTORE# ["Chaîne de caractères recherchée" [Ⓟ]
Expression numérique
[. { $\emptyset$ }] [. { Numéro de ligne
Numéro de zone de programme# }]]]

Function:

Recherche les données mémoire dans la DATA BANK et spécifie la séquence de lecture des données par READ#.

Paramètre:

Chaîne de caractères recherchée: Expression alphanumérique. Si on utilise une chaîne de caractères, la placer entre guillemets “ ”.

Numéro de ligne: Expression numérique.
0 < numéro de ligne < 10000

No. de zone de programme: Expression numérique.
 $0 \leq \text{No. de zone de programme} < 10$

Explication:

- 1) Recherche les données mémoire dans la DATA BANK et spécifie la séquence de lecture des données par l'instruction READ# suivante.
- 2) La relation entre un paramètre et la recherche de données est comme suit.

a) **RESTORE #**

Si la chaîne de caractères à rechercher et sa suite ne sont pas spécifiés, les données sont lues depuis le début par la READ# suivante.

b) **RESTORE #** "chaîne de caractères recherchée"

La donnée mémoire dont la chaîne de caractères recherchée se trouve au début est lue par la **READ#** suivante.

c) **RESTORE** # "chaîne de caractères recherchée", { 0 }

Si 0 est spécifié, comme en b) ci-dessus.

Si **l** est spécifié, la première donnée de la ligne qui comprend la donnée recherchée est lue par l'instruction **READ #** suivante.

- d) **RESTORE #** "chaîne de caractères recherchée" , $\left\{ \begin{Bmatrix} 0 \\ 1 \end{Bmatrix} \right\}$,
 $\left\{ \begin{array}{l} \text{numéro de ligne} \\ \text{No. de zone de programme \#} \end{array} \right\}$

Quand la recherche est exécutée, si la donnée appropriée n'existe pas, on saute à la ligne ou à la zone de programme spécifiée.

- * En b) et c), si la donnée appropriée n'existe pas, une erreur (Error4) se produit.
- * En d), si le numéro de ligne de rupture de séquence n'existe pas, ou si le programme n'existe pas dans la zone de programme, une erreur (Error 4) se produit.

- * Si un paramètre est affecté, la donnée appropriée est recherchée à partir de la donnée lue par l'instruction READ# suivante. Introduire RESTORE# : RESTORE# "chaîne de caractères recherchée" si l'on souhaite commencer la recherche à la première donnée.

Example:

< Données mémoire >

Enregistrement 1 Smith,03-347-4811, San Diego

Enregistrement 2 Jones, 075-351-1161, Princeton

Enregistrement 3 Williams, 06-314-2681, Cleveland

Enregistrement 4 Edwards, 045-211-0821, Cambridge

< Programme >

10 RESTORE#	}	Affiche les données mémoire se trouvant au début de la DATA BANK.
20 GOSUB 1000		
30 RESTORE#" J "	}	Affiche les données dont le premier caractère est J.
40 GOSUB 1000		
50 RESTORE#" CI " , 1	}	Recherche les données dont les deux premiers caractères sont CI et affiche la première donnée sur cette ligne.
60 GOSUB 1000		
70 RESTORE#" Aa " , 1 , 200	}	Passe à la ligne 200 s'il n'existe pas de donnée dont les deux premiers caractères sont Aa.
80 GOSUB 1000		
90 END		

```
200 BEEP:PRINT "Memo End"
210 END
1000 READ#$:PRINT$
1010 RETURN
```

} Sous-programme de lecture et d'affichage
des données mémoire.

< Exemple d'exécution >

RUN

EXE

EXE

EXE

Smith
Jones
Williams
Memo End

WRITE#

[Data [,Data]*]
expression expression

Ⓟ

Fonction:

Réécrit ou annule les données mémoire dans la DATA BANK.

Paramètre:

Données: Expression numérique ou expression alphanumérique.
Si on utilise une chaîne de caractères, la placer entre
guillemets " ".

Explication:

- 1) Ecrit les données dans la zone d'enregistrement présentement spécifiée par RESTORE#.
- 2) Les données nouvelles sont écrites sans relation aucune avec l'existence de données dans la zone d'enregistrement appropriée.
- 3) Si aucune donnée n'est spécifiée, les données en mémoire dans la zone d'enregistrement sont annulées.
- 4) Pour écrire plusieurs données dans la même zone d'enregistrement, on peut utiliser des "," pour la ponctuation.
- 5) Une fois les données nécessaires écrites à l'aide de la première instruction WRITE#, les données qui suivent sont écrites par l'instruction WRITE# suivante.
- 6) Pendant l'écriture des données mémoire, un pas sera nécessaire en plus du nombre de caractères.

Données mémoire ABC ~ XYZ

(26 caractères) + 1 pas = 27 pas

Exemple:

```
10 REM data write
20 RESTORE#
30 WRITE#"X,Y,Z"
40 GOSUB 1000
50 PRINT"_"
```

} Ecriture des nouvelles données
mémoire.

```

110 REM data change
120 RESTORE#
130 FOR J=1 TO 3
140 WRITE# STR$(J)
150 NEXT J
160 GOSUB 1000
170 PRINT"_"
210 REM data clear
220 RESTORE#
230 WRITE#
240 RESTORE#
250 READ#$
260 END

1000 REM display sub
1010 RESTORE#
1020 FOR J=1 TO 3
1030 READ#$:PRINT$;
1040 NEXT J
1050 RETURN

```

} Réécriture des données mémoire.
 } Données mémoire annulées.
 } Sous-programme d'affichage des données mémoire.

<Exemple d'exécution:>

MOD 1
 NEW# EXE
 MOD 2
 RUN EXE
 EXE
 EXE

} Efface toutes les données mémoire
 dans la DATA BANK.

```

XYZ
123
Error4 P0-250

```

↓
 Cette erreur se produit quand les données
 sont annulées et que la DATA BANK ne
 contient plus aucune donnée mémoire.

8

Fonction DATA BANK utile

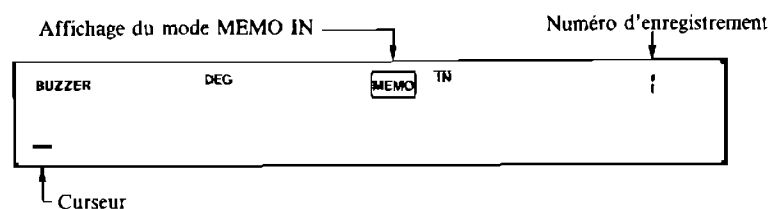
La DATA BANK (banque de données) peut s'occuper de votre agenda privé et traiter noms, numéros de téléphone, adresses, dates etc... Une donnée particulière peut être retrouvée rapidement dans l'ensemble des données en mémoire sans l'aide d'un programme spécial. De plus, grâce au programme BASIC, il est possible d'adapter cette fonction à une large gamme d'utilisations comme les emplois du temps, la réalisation de totaux etc. Les explications d'utilisation de la fonction DATA BANK sont accompagnées d'exemples concrets. Pour le détail concernant les commandes exclusives (LIST #, RESTORE #, READ #, WRITE #) de la DATA BANK, voir Chapitre 7 "Détail des commandes".

8-1 Pour spécifier le mode MEMO IN

Grâce à sa fonction DATA BANK, cet ordinateur peut être utilisé comme un "agenda électronique" dans lequel les données (données mémo) peuvent être écrites simplement et retrouvées par la simple opération de touches.

Pour utiliser la fonction DATA BANK il est nécessaire de commencer par introduire et mettre en mémoire un certain nombre de données. Pour introduire ces données dans la DATA BANK, spécifier le mode MEMO IN.

Ce mode MEMO IN est spécifié par  et l'écran affiche alors ce qui suit.



Le symbole "

Sur l'affichage ci-dessus, le curseur clignote en bas et à gauche de l'écran, aucune donnée n'a encore été introduite dans la DATA BANK et l'ordinateur est en état d'attente d'entrée par clavier pour l'Enregistrement 1. Si le numéro d'enregistrement est 5, cela signifie que quatre lignes de données mémo ont déjà été mises en mémoire dans la DATA BANK.

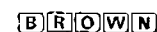
8-2 Introduction des données

Supposons que la DATA BANK soit utilisée comme un "répertoire téléphonique électronique" et introduisons les noms et numéros de téléphone de 10 personnes.

BROWN	03-021-1234	SMITH	0899-02-1007
ELLIS	011-041-7386	SULLIVAN	078-039-7132
FOX	06-021-6602	WATTS	0467-01-3569
JONES	052-031-6221	YOUNG	0425-01-0038
MILLS	03-063-2935	HOYT	03-054-4321

Introduire tout d'abord BROWN en mode MEMO IN en procédant comme suit. Séparer le nom du numéro de téléphone par une "," (virgule).

Opération











Affichage

BUZZER	DEG	MEMO IN	1
BROWN_			
BUZZER	DEG	MEMO IN	1
BROWN, _			
BUZZER	DEG	MEMO IN	1
BROWN, 03- _			
BUZZER	DEG	MEMO IN	1
BROWN, 03-021-1234_			
BUZZER	DEG	MEMO IN	1
BROWN, 03-021-1234			

Après avoir introduit les données, appuyer sur la touche , le curseur disparaît et la donnée BROWN est mémorisée dans la DATA BANK sous forme de donnée mémo d'enregistrement 1.

Appuyer une nouvelle fois sur la touche . L'affichage est libéré et le numéro d'enregistrement passe à 2. Le curseur clignote pour indiquer que l'ordinateur est en attente d'entrée par clavier.



BUZZER	DEG	MEMO IN	2
_			

Ensuite, entrer la donnée ELLIS en Enregistrement 2.

Opération:

ELLIS	BUZZER	DEG	MEMO	IN	2
7	BUZZER	DEG	MEMO	IN	2
011-	BUZZER	DEG	MEMO	IN	2
041-7386	BUZZER	DEG	MEMO	IN	2
EXE	BUZZER	DEG	MEMO	IN	2

En procédant de la même manière, entrer les noms et les numéros de téléphone de FOX à HOYT dans l'ordre. Ne pas oublier d'appuyer sur la touche [EXE] après chaque numéro de téléphone.

Notes:

1. Ce n'est pas sans raison que le nom et le numéro de téléphone doivent être séparés par une "," (virgule). La virgule est un symbole spécial indiquant la séparation entre deux données sur une même ligne dans la DATA BANK et comme ce symbole joue un rôle important dans le repérage ultérieur des données mémo, ne jamais oublier de placer la virgule.
2. Dans l'exemple précédent, après avoir mis un enregistrement en mémoire à l'aide de la touche [EXE], cette touche [EXE] était de nouveau enclenchée pour libérer l'affichage et passer au numéro d'affichage suivant. Toutefois, lorsque des données sont entrées de manière successive, il est inutile d'appuyer 2 fois de suite sur la touche [EXE] pour libérer l'affichage. Les données sont mises en mémoire à la fin de chaque enregistrement en appuyant une seule fois sur la touche [EXE]. Lorsque le premier caractère des données suivantes est entré, les

données précédentes disparaissent automatiquement de l'écran. Le numéro d'enregistrement ne change que lorsque la touche [EXE] est de nouveau enclenchée. Par exemple, les données FOX peuvent être entrées à la suite des données ELLIS de la manière suivante.

Opération:

[EXE] (Termine l'entrée des données ELLIS)	BUZZER	DEG	MEMO	IN	2
FOX 7	BUZZER	DEG	MEMO	IN	2
06-	BUZZER	DEG	MEMO	IN	2
021-6602	BUZZER	DEG	MEMO	IN	2
[EXE] (Le numéro d'enregistrement passe à 3)	BUZZER	DEG	MEMO	IN	3

Quand l'entrée des données pour les dix personnes est terminée, appuyer une nouvelle fois sur la touche [EXE] pour libérer l'écran.

[EXE] (Termine l'entrée des données HOYT)	BUZZER	DEG	MEMO	IN	10
[EXE] (Libère l'écran)	BUZZER	DEG	MEMO	IN	11

8-3 Affichage du contenu des données

Pour vérifier les données entrées, afficher le contenu des données mémorisées pour les dix personnes. Spécifier le mode RUN en appuyant sur .



BUZZER RUN DEG
Ready P0

Pour afficher tout le contenu de la DATA BANK, utiliser la commande LIST #. Les données de chacun des enregistrements seront successivement affichées à intervalle d'environ 1,5 s avec le numéro d'enregistrement.

Opération:



BUZZER RUN DEG
LIST #_

BUZZER RUN DEG
1 BROWN,03-021-1234

BUZZER RUN DEG
2 ELLIS,011-041-7386

BUZZER RUN DEG
3 FOX,06-021-6602

...

BUZZER RUN DEG
9 YOUNG,0425-01-0038

BUZZER RUN DEG
10 HOYT,03-054-4321

BUZZER RUN DEG
Ready P0

Pour arrêter provisoirement l'affichage et vérifier le contenu appuyer sur la touche . Pour reprendre l'affichage appuyer sur la touche . Toute erreur dans les données entrées peut ainsi être vérifiée.

* La commande LIST# peut aussi être exécutée en mode WRT (.

8-4 Correction des données

Supposons que le contenu de la DATA BANK ait été affiché à l'aide de la commande LIST # et que, après contrôle, il ait été déterminé que le numéro de téléphone de JONES en Enregistrement 4 est erroné.

Faux 052-031-6211 → Exact 052-031-6221

Dans ce cas, spécifier le mode MEMO IN et corriger la donnée en procédant comme suit.

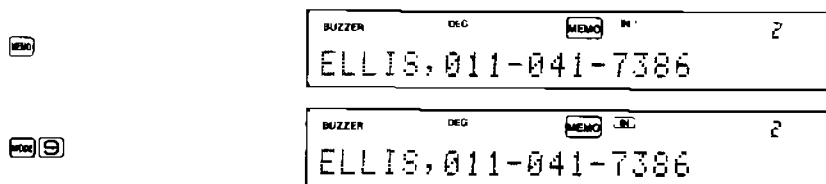
Opération:

	BUZZER DEG MEMO IN 1 1	Spécifie le mode MEMO IN.
	BUZZER DEG MEMO IN 1 BROWN,03-021-1234	Affiche la donnée mémo en Enregistrement 1.
	BUZZER DEG MEMO IN 2 ELLIS,011-041-7386	Avance à l'enregistrement suivant quand la touche est enclenchée.
	BUZZER DEG MEMO IN 3 FOX,06-021-6602	
	BUZZER DEG MEMO IN 4 JONES,052-031-6211	Affiche la donnée mémo en Enregistrement 4.
	BUZZER DEG MEMO IN EDIT 4 JONES,052-031-6211_	Appuyer sur la touche de déplacement du curseur pour spécifier le mode EDIT.
	BUZZER DEG MEMO IN EDIT 4 JONES,052-031-6211	Amener le curseur dans la position de l'erreur à corriger.
	BUZZER DEG MEMO IN EDIT 4 JONES,052-031-6221	Corriger la donnée.
	BUZZER DEG MEMO IN 4 JONES,052-031-6221	Appuyer sur la touche une fois la correction effectuée.

Si le mode MEMO IN est spécifié à l'aide des touches , l'enregistrement affiché est celui dont le numéro suit le dernier enregistrement mis en mémoire (numéro d'enregistrement 11 dans notre exemple). Appuyer alors sur la touche pour afficher l'Enregistrement 1. Le numéro d'enregistrement avancera d'une unité toutes les fois que la touche est enclenchée.

Quand on procède de cette manière, les données JONES se trouvant dans l'Enregistrement 4 sont affichées sans curseur. Appuyer sur l'une des touches de déplacement du curseur (ou) pour faire apparaître celui-ci, l'écran indiquant alors le symbole "EDIT". Cet état correspond au mode EDIT qui permet de corriger les erreurs dans les données mémo. A l'aide des touches de déplacement amener le curseur dans la position de l'erreur à corriger et après avoir fait les corrections nécessaires, appuyer sur la touche . Les données corrigées sont alors en mémoire. (Le symbole "EDIT" disparaît.)

Si on appuie sur les touches en mode RUN alors qu'une donnée MEMO est affichée, l'écran passe en mode MEMO IN avec cette ligne affichée.



Pour corriger la donnée affichée, il suffit d'appuyer sur la touche de déplacement du curseur jusqu'à ce que le mode EDIT soit spécifié.

En procédant de cette manière, il est facile de corriger les erreurs trouvées lors de la recherche des données mémorisées en mode RUN.

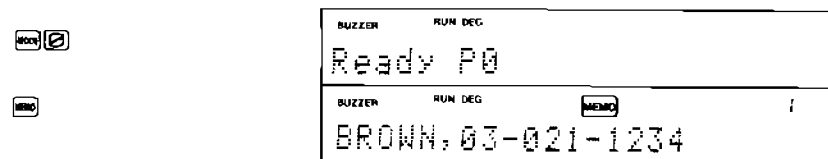
8-5 Repérage (recherche) des données

Avec la fonction DATA BANK, la recherche des données peut être effectuée directement à l'aide de la touche .

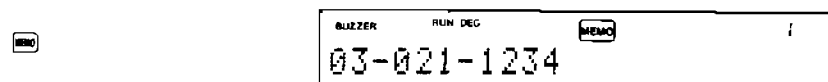
1) Recherche avec la touche

Quand on appuie sur la touche en mode RUN (), le nom BROWN et son numéro de téléphone dans l'Enregistrement 1 sont affichés. (Le symbole "MEMO" apparaît.)

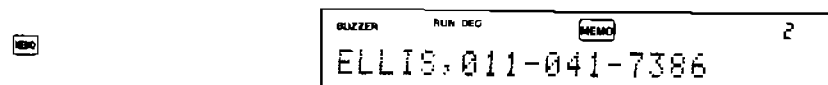
Opération:



Si on appuie de nouveau sur la touche , seul le numéro de téléphone est affiché.



Si on appuie encore sur la touche , les données ELLIS de l'Enregistrement 2 sont affichées.



Ainsi, toutes les fois qu'on appuie sur la touche , les données se trouvant dans un même enregistrement et séparées par une "," (virgule) sont affichées (les 24 premiers caractères dans le cas de données longues). Quand l'affichage de toutes les données d'un certain enregistrement est terminé, on passe à l'enregistrement suivant.

Quand toutes les données se trouvant dans la DATA BANK ont été affichées, l'écran est libéré et le symbole "MEMO" disparaît. Si on appuie de nouveau sur la touche **MEMO**, les données de l'Enregistrement 1 seront de nouveau affichées.

(Appuyer de manière répétée sur la touche **MEMO** pour afficher les données de l'Enregistrement 10.)

MEMO

MEMO

(Le symbole "MEMO" disparaît et le curseur clignote.)

MEMO

(Affichage des données de l'Enregistrement 1 de nouveau.)

BUZZER	RUN DEG	MEMO	10
HOYT, 03-054-4321			
BUZZER	RUN DEG	MEMO	10
03-054-4321			
BUZZER	RUN DEG	MEMO	
—			
BUZZER	RUN DEG	MEMO	1
BROWN, 03-021-1234			

2) Recherche rapide avec la touche **EXE**

La recherche et l'affichage des données mémo à l'aide de la touche **MEMO** peuvent être effectués enregistrement par enregistrement à l'aide de la touche **EXE**.

MOD **EXE**

MEMO

EXE

EXE

BUZZER	RUN DEG	MEMO	
Ready P0			
BUZZER	RUN DEG	MEMO	1
BROWN, 03-021-1234			
BUZZER	RUN DEG	MEMO	2
ELLIS, 011-041-7386			
BUZZER	RUN DEG	MEMO	3
FOX, 06-021-6602			

* Dans cet exemple, toutes les données sont affichées en même temps car le nombre de caractères ne dépasse pas la limite de 24 caractères. Si les données sont plus longues, seuls les 24 premiers caractères sont visibles sur l'écran.

Quand le dernier enregistrement mis en mémoire est affiché, il le reste même si on appuie sur la touche **EXE**.

(En appuyant plusieurs fois de suite sur la touche **EXE**, les données de l'Enregistrement 10 sont affichées.)

EXE (L'affichage n'est pas modifié.)

BUZZER	RUN DEG	MEMO	10
HOYT, 03-054-4321			
BUZZER	RUN DEG	MEMO	10
HOYT, 03-054-4321			

3) Recherche en arrière avec les touches **SHIFT** **EXE**

Pendant la recherche d'une donnée mémo avec la touche **MEMO** et la touche **EXE**, appuyer sur **SHIFT** **EXE** pour afficher l'enregistrement précédent.

Quand une recherche en arrière est effectuée enregistrement par enregistrement avec **SHIFT** **EXE** comme en recherche rapide avec la touche **EXE**, la commande revient aux données de l'Enregistrement 1. L'Enregistrement 1 est alors affiché de manière permanente même si on appuie de nouveau sur les touches **SHIFT** **EXE**. Cette opération peut être répétée jusqu'à affichage des données de l'Enregistrement 1.

(Données de l'Enregistrement 3 présentement affichées.)

SHIFT **EXE**

SHIFT **EXE** (Retour sur Enregistrement 1.)

SHIFT **EXE** (L'affichage reste inchangé.)

BUZZER	RUN DEG	MEMO	3
FOX, 06-021-6602			
BUZZER	RUN DEG	MEMO	2
ELLIS, 011-041-7386			
BUZZER	RUN DEG	MEMO	1
BROWN, 03-021-1234			
BUZZER	RUN DEG	MEMO	1
BROWN, 03-021-1234			

4) Recherche conditionnelle

Les données mémo sont successivement affichées dans l'ordre à partir de l'Enregistrement 1 quand la recherche est effectuée avec la touche **MEMO** ou la touche **EXE** mais si les données mémorisées sont en grand nombre, il faudra un certain temps avant que la donnée recherchée soit affichée. Si tel est le cas, il est pratique de recourir à une "recherche conditionnelle".

Par exemple, cherchons un nom commençant par la lettre "S". Appuyer sur

S **MEMO**.

BUZZER	RUN DEG	MEMO	
			5
SMITH, 0899-02-1007			
			7
SULLIVAN, 078-039-7132			
—			

Les données SMITH de l'Enregistrement 6 seront affichées. Si on appuie de nouveau sur **MEMO**, les données SULLIVAN de l'Enregistrement 7 sont affichées. Si on appuie de nouveau sur **MEMO**, l'écran est libéré et le curseur clignote. Cela signifie qu'il n'existe pas d'autre nom commençant par la lettre "S".

S'il existe plusieurs données pertinentes, elles seront affichées dans l'ordre croissant des numéros d'enregistrement.

La condition spécifiée n'est pas limitée à un seul caractère comme dans l'exemple précédent, mais peut être constituée d'une chaîne de caractères plus longue. Par exemple, si les 8 caractères de SULLIVAN sont utilisés comme condition, les données recherchées sont celles commençant par ces 8 caractères.

BUZZER	RUN DEG	MEMO	
SULLIVAN			
			7
SULLIVAN, 078-039-7132			
—			

5) Recherche conditionnelle supplémentaire

Supposons qu'après une recherche avec spécification de la condition "J", les données JONES de l'Enregistrement 4 soient affichées sur l'écran. Il est alors possible de procéder à une nouvelle recherche de données après l'Enregistrement 4 en spécifiant une condition supplémentaire.

Par exemple, recherche d'un numéro de téléphone commençant par l'indicatif 03 après l'Enregistrement 4. Appuyer sur **03** après l'Enregistrement 4. Appuyer sur **03** après l'Enregistrement 4. Appuyer sur **03** après l'Enregistrement 4.

(Supposons que les données JONES aient été affichées à l'aide de **03**.)

03 (Spécifie une condition supplémentaire.)

MEMO

MEMO

BUZZER	RUN DEG	MEMO	
			4
JONES, 052-031-6221			
			5
03-063-2935			
			10
03-054-4321			
—			

Si on appuie sur **03** comme ci-dessus, les données commençant par la condition "03" seront affichées après l'Enregistrement 4. S'il existe plusieurs données répondant à la condition "03", celles-ci seront successivement affichées toutes les fois que la touche **MEMO** sera enclenchée.

Si dans l'exemple précédent, on effectue non pas une recherche avec condition supplémentaire mais une recherche conditionnelle avec la condition "03", la première donnée affichée sera le numéro de téléphone de BROWN en Enregistrement 1.

03 (Recherche à condition unique.)

BUZZER	RUN DEG	MEMO	
			4
JONES, 052-031-6221			
			1
03-021-1234			

Donc, une recherche avec condition supplémentaire vous permet de trouver un enregistrement répondant à la première condition et, après modification de la condition, de rechercher les données répondant à cette condition dans les données suivantes. Cette fonction permet de cerner au plus près une recherche en modifiant la condition.

6) Confirmation avec la touche de déplacement du curseur

Si la recherche est effectuée en spécifiant la condition "03", seuls les numéros de téléphone seront affichés. La touche **→** de déplacement du curseur permet de confirmer le nom de la personne correspondant au numéro de téléphone.

   (Recherche avec condition.)

BUZZER	RUN DEG	MEMO	1
03-021-1234			
BUZZER	RUN DEG	MEMO	1
BROWN, 03-021-1234			

Toutes les fois que la touche  est enclenchée, le numéro de téléphone se déplace d'un espace vers la droite et le nom apparaît à gauche.

Pour afficher les données depuis le début, appuyer sur   au lieu d'appuyer 6 fois sur la touche .

BUZZER	RUN DEG	MEMO	1
03-021-1234			
BUZZER	RUN DEG	MEMO	1
BROWN, 03-021-1234			

Si le contenu d'un enregistrement contient plus de 24 caractères, il n'est pas possible de l'afficher en une seule fois. On peut utiliser la touche  pour décaler l'affichage vers la gauche et faire apparaître les caractères à droite jusqu'alors invisibles. Pour afficher les 24 derniers caractères de l'enregistrement, appuyer sur  .

8-6 Effacement des données

Comme pour leur correction (voir page 223), les données peuvent être effacées en mode EDIT.

1) Pour effacer une partie d'un enregistrement

Afficher les données concernées en mode MEMO IN et spécifier le mode EDIT en appuyant sur la touche de déplacement du curseur. Amener ensuite le curseur sous le caractère à effacer. Appuyer sur  pour effacer le caractère.

Quand tous les caractères souhaités ont été effacés, ne pas oublier d'appuyer sur la touche .

2) Pour effacer toutes les données d'un enregistrement

Afficher les données concernées en mode MEMO IN et spécifier le mode EDIT en appuyant sur la touche de déplacement du curseur. Libérer alors l'affichage en appuyant sur la touche . Appuyer ensuite sur la touche .

Toutes les données de cet enregistrement seront effacées et le numéro de l'enregistrement sera automatiquement affecté à l'enregistrement suivant.

Pour effacer toutes les données se trouvant dans la DATA BANK, exécuter la commande NEW # en mode WRT ( ).

Utiliser la commande NEW # avec discernement pour ne pas effacer des données importantes dans la DATA BANK.

8-7 Addition et insertion de données

Il est possible d'ajouter de nouvelles données à la suite du dernier enregistrement ou d'insérer un nouvel enregistrement entre des enregistrements déjà en mémoire.

1) Addition de données



BUZZER	DEG	MEMO IN	11
--------	-----	---------	----

Quand le mode MEMO IN est spécifié, l'ordinateur passe en état d'attente d'entrée par clavier et le numéro d'enregistrement est affiché. Dans l'exemple ci-dessus, le numéro d'enregistrement affiché est le numéro 11 et le curseur clignote. L'ordinateur est prêt pour l'entrée de données dans l'Enregistrement 11. Ceci signifie également que les données de 10 enregistrements ont déjà été mises en mémoire.

Les nouvelles données seront mises en mémoire dans l'Enregistrement 11 à l'aide de la touche .

2) Insertion de données

Pour insérer de nouvelles données entre des enregistrements déjà en mémoire, spécifier le mode MEMO IN et afficher l'enregistrement dans lequel ces données seront mises en mémoire. Entrer ensuite les données à insérer et appuyer sur . (Attention car si la seule touche  est enclenchée, les données seront mémorisées dans un enregistrement suivant le dernier enregistrement.)

Par exemple, supposons que nous souhaitons insérer les données HOYT entre les données FOX de l'Enregistrement 3 et les données JONES de l'Enregistrement 4. Afficher tout d'abord les données JONES en mode MEMO IN.

BUZZER	DEG	MEMO IN	4
JONES, 052-031-6221			

8-7 Addition et insertion de données

Entrer ensuite le premier caractère  de HOYT. L'affichage précédent est libéré et "H" est affiché. Entrer alors le reste des données HOYT.



BUZZER	DEG	MEMO IN	4
H			
BUZZER	DEG	MEMO IN	4
HOYT, 03-054-4321			

Finalement appuyer sur  .

BUZZER	DEG	MEMO IN	4
HOYT, 03-054-4321			

Les données HOYT sont désormais mémorisées en Enregistrement 4 et le numéro d'enregistrement des données JONES s'est décalé d'une unité. Appuyer sur la touche  et vérifier que les données JONES sont bien mémorisées en Enregistrement 5.



BUZZER	DEG	MEMO IN	5
JONES, 052-031-6221			

Mais, d'autre part, les données HOYT sont aussi mémorisées en Enregistrement 11.

(Affichage des données HOYT en Enregistrement 11.)

BUZZER	DEG	MEMO IN	11
HOYT, 03-054-4321			

Comme ces données ne sont plus nécessaires, les effacer en procédant comme suit.

 (Spécifie le mode EDIT.)

 (Libère l'affichage.)

 (Termine l'effacement. "EDIT" disparaît.)

BUZZER	DEG	MEMO IN	11
HOYT, 03-054-4321			
BUZZER	DEG	MEMO IN	11
BUZZER	DEG	MEMO IN	11

Les données HOYT sont désormais insérées entre FOX et JONES et le "répertoire téléphonique électronique" est classé par ordre alphabétique. La sortie des données avant et après classement est la suivante:

< Avant classement:>

```
1 BROWN, 03-021-1234
2 ELLIS, 011-041-7386
3 FOX, 06-021-6602
4 JONES, 052-031-6221
5 MILLS, 03-063-2935
6 SMITH, 0899-02-1007
7 SULLIVAN, 078-039-7132
8 WATTS, 0467-01-3569
9 YOUNG, 0425-01-0038
10 HOYT, 03-054-4321
```

<Après classement:>

```
1 BROWN, 03-021-1234
2 ELLIS, 011-041-7386
3 FOX, 06-021-6602
4 HOYT, 03-054-4321
5 JONES, 052-031-6221
6 MILLS, 03-063-2935
7 SMITH, 0899-02-1007
8 SULLIVAN, 078-039-7132
9 WATTS, 0467-01-3569
10 YOUNG, 0425-01-0038
```

8-8 Recherche à l'aide d'un programme

Tant que les données mémorisées ne se composent que de deux informations comme des noms et des numéros de téléphone, une recherche par opération manuelle à l'aide de la touche  comme décrit dans les sections précédentes ne pose pas de problèmes particuliers. Par contre, si le nombre des données est plus grand et si les informations mémorisées passent à trois, quatre ou plus, la longueur d'un enregistrement va dépasser 24 caractères. Il faudra alors déplacer l'affichage à l'aide de la touche  ou de la touche de déplacement du curseur pour appeler la donnée recherchée.

Il est facile de traiter une telle masse de données si la recherche de données est effectuée à l'aide d'un programme en BASIC.

La méthode de recherche que nous présentons est basée sur un programme utilisant les données que nous avons utilisées dans les sections précédentes (voir page 219). Cette méthode est aussi facilement applicable à des données comprenant trois informations ou plus.

• Généralités

Quand ce programme est exécuté, la question "Name?" (Nom?) est affichée. Si le nom de la personne dont on recherche le numéro de téléphone est entré, le numéro de téléphone sera affiché s'il existe dans la DATA BANK. S'il n'existe pas, "No Data!!" (Pas de donnée!!) sera affiché.

Ce programme n'étant prévu que pour la visualisation, spécifier le mode MEMO IN () pour entrer les noms ou les numéros de téléphone.

• Programme

```
10 DIM Z$(5)
20 RESTORE#
30 BEEP : INPUT "Name" , $
40 IF $="" THEN 30
50 FOR J=1 TO 5
60 Z$(J)= MID$(J*7-6,7)
70 NEXT J
80 F=0
90 RESTORE# Z$(1)+Z$(2)+Z$(3)+Z$(4)
    +Z$(5),1,140
```

```

100 READ# $,$
110 F=1
120 GOSUB 1000: FRINT $
130 GOTO 90
140 IF F=0 THEN PRINT "No Data !!"
150 GOTO 20
1000 FOR J=1 TO 3
1010 BEEP 1: BEEP 0
1020 NEXT J
1030 RETURN

```

197 octets

• Tableau des variables

Nom de la variable	Contenu
J	Variable de commande pour la boucle FOR ~ NEXT
F	Variable drapeau (F=1 si la personne en question existe. F=0 si la personne en question n'existe pas.)
Z\$(1) à Z\$(5)	Mémoisent les noms entrés.
\$	Pour la lecture des données.

• Exemple d'exécution

Opération:

```

MOD RUN
MILLS
EXT
EXT
NEWTON
EXT
EXT

```

```

Name?
MILLS_
03-063-2935
Name?
NEWTON_
No Data !!
Name?

```

8-9 Application aux calculs de tabulation

Dans la pratique, il est souvent nécessaire d'effectuer des calculs à partir de tableaux, tant de manière verticale qu'horizontale. Bien que ce type de calcul puisse être effectué à l'aide d'un tableau (voir page 105), il est possible de simplifier encore les calculs de tabulation en traitant la DATA BANK comme un grand tableau.

n \ m	Produit A	Produit B	Produit C	Produit D	Total horizontal
Agence X	5329	4280	3602	2310	
Agence Y	2682	6313	4203	1128	
Agence Z	5113	3229	5176	965	
Total vertical					

• Généralités

Les dimensions du tableau étant nécessaires pour l'exécution de ce programme, entrer le nombre d'informations horizontales (m) et le nombre d'informations verticales (n). Entrer les données une colonne après l'autre en commençant par le haut. "Calculation" (calcul) est affiché et les calculs commencent quand toutes les données ont été entrées. Quand les calculs sont terminés, les résultats sont affichés total vertical et total horizontal dans l'ordre.

• Programme

```

10 BEEP : INPUT "(m x n)",A,B: ERAS
  E F: DIM F(A+1,B+1)
20 FOR C=1 TO A
30 FOR D=1 TO B
40 PRINT "(";C;"x";D;")";
50 BEEP : INPUT F(C,D)
60 $="N"+ STR$(D)+"," + STR$(F(C,D))
70 IF A=C THEN $=$+" ,0"
80 WRITE# $
90 NEXT D
100 WRITE# "M"+ STR$(C)+",0"

```

```

110 NEXT C
120 PRINT "Calculation";
140 FOR C=1 TO B+1
150 F(A+1,C)=0
160 NEXT C
170 FOR D=1 TO A+1
180 F(D,B+1)=0
190 NEXT D
200 RESTORE#
210 FOR C=1 TO A
220 FOR D=1 TO B
230 READ# $,F(C,D)
240 F(A+1,D)=F(A+1,D)+F(C,D)
250 IF A=C THEN WRITE# F(A+1,D)
260 F(C,B+1)=F(C,B+1)+F(C,D)
270 NEXT D
280 READ# $: WRITE# F(C,B+1)
290 NEXT C
300 PRINT
310 RESTORE#
320 FOR C=1 TO A
330 RESTORE# "M"
340 READ# $,E
350 PRINT $;"=";E
360 NEXT C
370 RESTORE#
380 IF A>1 THEN RESTORE# "M"+ STR$(A-1)
390 READ# $,$
400 FOR C=1 TO B
410 READ# $,E,E
420 PRINT $;"=";E

```

```

430 NEXT C
440 BEEP : PRINT "OVER"
450 PRINT : END

```

500 octets

• Tableau des variables

Nom des variables	Contenu
A	Nombre d'informations horizontales dans le tableau
B	Nombre d'informations verticales dans le tableau
C	Variable de commande pour une boucle FOR ~ NEXT
D	Variable de commande pour une boucle FOR ~ NEXT
E	Pour la lecture des données
F	Pour le calcul du total vertical
F(1) à	Tableau pour calcul du total horizontal (Prépare le nombre d'informations verticales dans le tableau.)
\$	Pour la préparation à l'écriture des données dans la DATA BANK et pour la lecture des espaces à l'aide de READ# \$.

• Exemples d'exécution du programme

Calculer les totaux verticaux et horizontaux à l'aide du tableau de la page 237.

Opération:

MOD  RUN 

4  (Entre le nombre d'informations horizontales)

3  (Entre le nombre d'informations verticales)

5 3 2 9  (Entre la valeur dans la première colonne verticale)

2 6 8 2 

(m × n)?

?

(1 × 1)?

(1 × 2)?

(1 × 3)?

Entrer les données dans les 2ème, 3ème et 4ème colonnes dans l'ordre. Quand toutes les données sont entrées, "Calculation" est affiché. Quand les calculs sont terminés, les résultats suivants sont affichés.

965 **EXE**

(Affichage du total vertical de la 1ère colonne.)

EXE (Affichage du total vertical de la 2ème colonne.)**EXE** (Affichage du total vertical de la 3ème colonne.)**EXE** (Affichage du total vertical de la 4ème colonne.)**EXE** (Affichage du total horizontal de la 1ère ligne.)**EXE** (Affichage du total horizontal de la 2ème ligne.)**EXE** (Affichage du total horizontal de la 3ème ligne.)**EXE** (Les calculs sont terminés.)**EXE**

Calculation	
M1=	13124
M2=	13822
M3=	12981
M4=	4403
N1=	15521
N2=	14326
N3=	14483
OVER	
—	

* Avec ce programme, toutes les données étant mémorisées sous la forme de données mémo dans la DATA BANK, pour l'exécuter avec de nouvelles données, exécuter tout d'abord la commande NEW # en mode WRT.

8-10 Combinaison avec la fonction mémoire

Une méthode pratique d'utilisation de la fonction DATA BANK consiste à utiliser celle-ci conjointement avec la Mémoire Fonction. Mémoriser formules et équations dans la DATA BANK et appuyer sur la touche **MEM** pour appeler la formule ou l'équation souhaitée. Effectuer les calculs en mémorisant la formule retrouvée dans la Mémoire Fonction à l'aide de la touche **IN**.

Supposons que formules et équations suivantes soient mémorisées dans la DATA BANK.

1	QUADRATIC EQUATION (EQUATION QUADRATIQUE)
2	$X = (-B + \text{SQR}(B^2 - 4 * A * C)) / (2 * A) ; X = (-B - \text{SQR}(B^2 - 4 * A * C)) / (2 * A)$
3	HERON (FORMULE DE HERON)
4	$S * S = (A + B + C) / 2 ; S = \text{SQR}(S * (S - A) * (S - B) * (S - C))$

La formule permettant de calculer la racine carrée d'une équation quadratique et la formule de Héron sont mémorisées dans la DATA BANK.

Calculons maintenant la racine carrée de l'équation quadratique. Appuyer sur **Q** **MEM** pour chercher l'équation quadratique.

Q **MEM**

QUADRATIC EQUATION

Après vérification du nom, appuyer sur la touche **EXE** et la formule de calcul est affichée.

EXE $X = (-B + \text{SQR}(B^2 - 4 * A * C)) / (2)$

Mémoriser alors la formule de calcul dans la Mémoire Fonction.

IN

—

Tout est prêt pour les calculs.

Entrer 2 pour a, -3 pour b et -10 pour c par exemple.

CALC	B ?
-3 EXE	A ?
2 EXE	C ?
-10 EXE	X= 3.108495283
EXE	X=-1.608495283

Cette fonction est extrêmement pratique car les formules de calcul mémorisées dans la DATA BANK peuvent être appliquées telles quelles à la Mémoire Fonction.

Note:

Pour introduire le nom d'une formule et la formule de calcul dans la DATA BANK, ne pas les mémoriser dans un seul et même enregistrement. Mémoriser la formule de calcul dans l'enregistrement suivant l'enregistrement dans lequel le nom de la formule est mémorisé.

Appendice

Table du code caractère

	(space)	+	-	*	/	↑	!	"	#	\$	>	≥	=	≤	<	≠
Nombres	0	1	2	3	4	5	6	7	8	9	.	π	)	(	E	E
Lettres majuscules	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
	Q	R	S	T	U	V	W	X	Y	Z	γ	σ				
Lettre minuscules	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p
	q	r	s	t	u	v	w	x	y	z						
Symboles	?	,	:													
Symboles graphiques	○	□	△	@	X	÷	♠	←	♥	♦	+	μ	Ω	!	→	
	%	¥	□	□	□	□	□	□	□	□	□	□	□	□	□	□

* Caractères et symboles du tableau précédent sont alignés dans l'ordre, d'espace à ☐ (" ☐ " peut être affiché à l'aide de ☐☐ en mode étendu).

Calculs de fonctions

Nom des fonctions	Format	Fonction et gamme d'entrée	
Fonction trigonométrique	SIN (Expression numérique) * ci-après X	sin	$ X < 1440^\circ$ (8π rad, 1600 gra)
	COS (X) TAN (X)	cos tan	$ X < 1440^\circ$ (8π rad, 1600 gra) $ X < 1440^\circ$ (8π rad, 1600 gra) sauf quand $ X $ est un multiple impair de 90° ($\pi/2$ rad, 100 gra)
Fonction trigonométrique inverse	ASN (X)	$\sin^{-1}$	$ X \leq 1$, $-90^\circ \leq \text{ASN} \leq 90^\circ$ (rad: $-\pi/2 \leq \text{ASN} \leq \pi/2$, gra: $-100 \leq \text{ASN} \leq 100$)
	ACS (X)	$\cos^{-1}$	$ X \leq 1$, $0^\circ \leq \text{ACS} \leq 180^\circ$ (rad: $0 \leq \text{ACS} \leq \pi$, gra: $0 \leq \text{ACS} \leq 200$)
	ATN (X)	$\tan^{-1}$	$-90^\circ \leq \text{ATN} \leq 90^\circ$ (rad: $-\pi/2 \leq \text{ATN} \leq \pi/2$, gra: $-100 \leq \text{ATN} \leq 100$)
Fonction hyperbolique	HYP SIN (X)	sinh	$ X \leq 230,2585092$
	HYP COS (X)	cosh	$ X \leq 230,2585092$
	HYP TAN (X)	tanh	$ X < 10^{100}$
Fonction hyperbolique inverse	HYP ASN (X)	$\sinh^{-1}$	$ X < 5 \times 10^{99}$
	HYP ACS (X)	$\cosh^{-1}$	$1 \leq X < 5 \times 10^{99}$
	HYP ATN (X)	$\tanh^{-1}$	$ X < 1$
Racine carrée	SQR (X)	$\sqrt{x}$	$X \geq 0$
Racine cubique	CUR (X)	$\sqrt[3]{x}$	$ X < 10^{100}$
Puissance	X ↑ X	x^y	$x < 0 \rightarrow y$: nombre naturel

Nom des fonctions	Format	Fonction et gamme d'entrée	
Fonction exponentielle	EXP (X)	e^x	$-10^{100} < X \leq 230,2585092$
Logarithme commun	LOG (X)	$\log_{10} x$	$X > 0$
Logarithme naturel	LN (X)	$\log_e x$	$X > 0$
Intégrale	INT (X)	$[x]$	Donne l'intégrale maximum ne dépassant pas X (égal à la fonction de Gauss $[x]$)
Fraction	FRAC (X)	FRAC	Donne la partie décimale de X
Valeur absolue	ABS(X)	$ x $	Donne la valeur absolue de X
Signe	SGN (X)	$\text{sgn } x$	1 quand $X > 0$ 0 quand $X = 0$ -1 quand $X < 0$
Arrondissement	RND (X, Nombre de chiffres)*	RND (	Donne la valeur de X arrondie à l'unité spécifiée. $ \text{Nombre de chiffres} < 100$
Nombres au hasard	RAN #	RAN #	Fournit un nombre à 10 chiffres au hasard. $0 < \text{RAN\#} < 1$
π	π	π	Donne la valeur approximative du quotient de la circonférence d'un cercle par son diamètre.
Conversion décimale → sexagésimale	DMS\$ (X)*	DMS\$ (	Convertit un nombre décimal donné comme X en chaîne de caractères sexagésimale en degrés, minutes et secondes. $ X < 10^5$
Conversion sexagésimale → décimale	DEG (deg. [, min. [, sec.]])*	DEG (	$\text{DEG } (x, y, z) = x + y/60 + z/3600$. $ \text{DEG } (x, y, z) < 10^{100}$

Nom des fonctions	Format	Fonction et gamme d'entrée	
Conversion décimale → hexadécimale	HEX\$ (X)*	HFX\$ (	Convertit la valeur de X en chaîne de caractères hexadécimale de 4 chiffres $-32769 < X < 65536$
Conversion hexadécimale → décimale	&H Chaîne de caractères hexadécimale	&Hx	La chaîne de caractères contient un nombre hexadécimal dans les limites de 4 caractères.
Factorielle	FACT (X)	$x!$	$0 \leq X \leq 69$ (0 et nombre entier positif)
Permutation	NPR (n, r)*	nPr	$0 \leq r \leq n < 10^{10}$ (0 et nombre entier positif)
Combinaison	NCR (n, r)*	nCr	$0 \leq r \leq n < 10^{10}$ (0 et nombre entier positif)
Transformation des coordonnées rectangulaires en polaires	POL (X, Y)* X, Y : Expressions numériques	POL (	$ X < 10^{100}$, $ Y < 10^{100}$, $ X + Y \neq 0$ r est donné comme valeur de fonction pour affectation à la variable X alors que la valeur de θ est affectée à la variable Y.
Transformation des coordonnées polaires en rectangulaires.	REC (r, θ)* r, θ : Expressions numériques	REC (	$0 \leq r < 10^{100}$, $ \theta < 1440^\circ$ (8π rad, 1600 gra) Donne x comme valeur de fonction pour affectation à la variable X alors que la valeur de y est affectée à la variable Y.

Note:

Pour les fonctions repérées par un astérisque, les paramètres doivent être placés entre parenthèses.

*Certaines combinaisons ou permutations peuvent provoquer des erreurs dues à un dépassement de capacité pendant les calculs internes.

Code erreur/ signification	Cause	Contremesure
Error 1 Mémoire épuisée ou bloc du système saturé	- La capacité insuffisante de la zone libre ne permet pas d'écrire des programmes ou d'augmenter les variables. - La zone de calcul (bloc) ne peut fixer l'équation qui est excessivement complexe. - Impossible d'écrire les données dans la banque de données dont la capacité est insuffisante. - Neuf tableaux ou plus sont déclarés.	- Effacer les programmes inutiles à l'aide de la commande NEW ou réduire le nombre de variables. - Couper et simplifier l'équation. - Effacer le tableau.
Error 2 Erreur de syntaxe	- Erreur de format dans le programme ou l'équation. - Les formats du côté droit et du côté gauche de l'instruction affectée différent. (Par exemple caractère et nombre). - Tentative de lecture d'un caractère dans une variable numérique avec READ/READ#. - La chaîne de caractères dépasse 62 caractères.	- Corriger l'erreur dans le programme. - Changer la variable numérique pour une variable de caractère ou chercher le caractère (un espace est un caractère) dans l'instruction DATA. - Raccourcir la chaîne de caractères.
Error 3 Erreur mathématique	- Quand le résultat des calculs d'une équation est supérieur à 10^{100}. (Dépassement). - Quand les arguments sont hors de la gamme d'entrée des fonctions numériques. - Quand les résultats sont incertains ou impossibles. (Essayer de diviser par 0).	- Corriger l'équation ou les données. - Contrôler les données.

Code erreur/ signification	Cause	Contremesure
Error 4 Erreur non définie	- Aucune destination de saut pour les instructions GOTO ou GOSUB. - Il n'existe pas de données lisibles avec READ/READ# ou RESTORE#. - Le numéro de ligne spécifié avec RESTORE n'existe pas.	- Spécifier la bonne destination du saut. - Ecrire des données. - Corriger le numéro de ligne.
Error 5 Erreur d'argument	- Quand l'argument est hors de la gamme d'entrée des commandes et des fonctions nécessitant un argument. - L'indice dans le tableau est hors gamme d'entrée. - Tentative de spécification de deux tableaux avec le même nom mais des indices différents.	- Corriger l'erreur d'argument. - Changer l'indice. - Changer le nom du tableau.
Error 6 Erreur de variable	- Tentative d'utilisation d'une variable qui n'a pas été ajoutée. - Tentative d'utilisation du même nom de variable pour une variable numérique et une variable caractère. - Tentative d'utilisation d'un indice de nom de tableau qui n'a pas été déclaré.	- Augmenter le nombre des variables à l'aide d'une instruction DEFN. - Changer le nom de la variable pour la variable numérique et la variable caractère. - Déclarer d'abord le tableau ou corriger l'indice du nom de tableau.
Error 7 Erreur d'emboîtement	- Quand l'instruction RETURN est utilisé autrement que pour l'exécution d'un sous-programme. - Quand l'instruction FOR et l'instruction NEXT ne correspondent pas ou bien quand la variable de l'instruction NEXT ne correspond pas à celle de l'instruction FOR. - Quand l'emboîtement des sous-programmes (appel d'un sous-programme depuis un sous-programme) dépasse les huit niveaux.	- Faire correspondre correctement GOSUB ~ RETURN ou FOR ~ NEXT. - Corriger le sous-programme ou amener le niveau d'emboîtement de la boucle FOR au niveau adéquat.

Code erreur/ signification	Cause	Contremesure
Error 7 Erreur d'emboîtement	• Quand l'emboîtement d'une boucle FOR (insertion d'une boucle dans une boucle avec forme d'emboîtement) dépasse les quatre niveaux. • L'instruction CLEAR a été utilisée dans une boucle FOR ~ NEXT.	• Déplacer l'instruction CLEAR hors de l'instruction FOR ~ NEXT.
Error 8 Erreur de protection	• Quand un mot de passe a été spécifié et que: 1) Un mot de passe différent est entré, 2) Une commande interdite est exécutée, 3) Un programme est édité, 4) Des programmes sont chargés avec des mots de passe différents. 5) Entrée des données dans la banque de données. 6) Appel des données depuis la banque de données.	• Supprimer le mot de passe.
Error 9 Erreur option	• La commande SAVE ou PUT a été exécutée sans interface. • Quand le signal entré par la commande LOAD, LOAD# ou GET est erroné et ne peut pas être chargé. • Pas d'imprimante. • Quand l'imprimante n'est pas suffisamment chargée. • Le papier est coincé dans l'imprimante.	• Raccorder un magnétophone à cassette. • Réduire le volume d'écoute du magnétophone. • Régler la commande de tonalité du magnétophone en position moyenne. • Changer la cassette. • Nettoyer les têtes du magnétophone. • Charger l'imprimante. • Dégager le papier coincé dans l'imprimante.

Spécifications

Type:

FX-795P

Fonctions de calculs fondamentaux:

Nombres négatifs, exposants, addition, soustraction, multiplication et division entre parenthèses (avec fonction d'estimation d'ordre de priorité — logique algébrique vraie).

Fonctions incorporées:

Fonctions trigonométrique/trigonométrique inverse (unités d'angle en degrés/radians/grades), fonctions hyperbolique/hyperbolique inverse, fonctions logarithmique/exponentielle, racines carrées, racines cubiques, puissances, conversion en nombre entier, annulation de la partie entière, valeurs absolues, signes, désignation du nombre de chiffres significatifs, désignation du nombre de chiffres après la virgule, conversions décimale ↔ sexagésimal, conversions décimal ↔ hexadécimal, transformations des coordonnées rectangulaires ↔ polaires, factorielles, permutations, combinaisons, arrondissement, génération de nombres aléatoires, π .

Utilitaires incorporés:

Calcul matriciel, calcul de nombres complexes, solution numérique d'équations, intégration numérique et opérations binaires, décimales et hexadécimales.

Fonctions de calcul de statistiques:

Nombre de données, somme des x , somme des y , somme des x^2 , somme des y^2 , somme de $x.y$, moyenne de x , moyenne de y , moyenne des différences de x (2 types), moyenne des différences de y (2 types), terme constant de régression linéaire, coefficient de régression linéaire, coefficient de corrélation, valeur estimée de x , valeur estimée de y .

Commandes:

ERASE, INPUT, PRINT, GOTO, ON ~ GOTO, FOR ~ TO ~ STEP ~ NEXT, IF ~ THEN, GOSUB, ON ~ GOSUB, RETURN, READ, DATA, RESTORE, STOP, END, RUN, LIST, LIST ALL, LIST V, MODE, SET, CLEAR, NEW, NEW ALL, DIM, DEFN, PASS, REM, BEEP, LET, SAVE, SAVE ALL, LOAD, LOAD ALL, PUT, GET, VERIFY, NEW #, LIST #, LOAD #, SAVE #, READ #, WRITE #, RESTORE #, STAT, STAT CLEAR, STAT LIST.

Fonctions de programmation:

DM\$, KEY\$, CSR, LEN, MID\$, STR\$, VAL, HEX\$.

Limites de calcul:

$\pm 1 \times 10^{-99}$ à $\pm 9,999999999 \times 10^{99}$ et 0. Les opérations internes utilisent une mantisse de 12 chiffres.

Système de programme:

Système enregistré.

Langage de programme:

BASIC

Capacité de RAM (mémoire à accès direct):

15584 octets standard (zone système: 800 octets).

Nombre de zones de programme:

Maximum 10 (P0 à P9)

Nombre de variables:

26 variables fixes, variables de caractère exclusives et variables de tableau.

Nombre de piles:

Sous-programme: 8 niveaux

Boucle FOR~NEXT: 4 niveaux

Valeurs numériques: 6 niveaux

Opérateurs: 12 niveaux

Nombre de chiffres affichés:

Mantisse de 10 chiffres (y compris le signe moins) et exposant de 2 chiffres.

Contenu de l'affichage: BUZZER, EXT, [S], RUN, WRT, DEG,
RAD, GRA, DEFM, [MEMO], [IN], [EDIT],
TRACE ON, PRT ON, STOP

Éléments d'affichage:

Affichage à cristaux liquides à matrice de points 24 chiffres

Principaux composants:

C-MOS (semiconducteur oxyde-métal à symétrie complémentaire) VLSI
(circuit de logique intégré à multiplicateur) et autres

Alimentation électrique:

2 piles au lithium (CR2032) pour l'unité centrale de traitement

1 pile au lithium (CR1220) pour l'entretien de la mémoire

Puissance absorbée:

Maximum 0,07W

Durée de vie des piles:

1) Exécution de programme continue: environ 80 heures.

2) Affichage continu de 555555555 à 20°C (68°F): environ 180 heures ou
5 mois quand l'appareil est utilisé 1 heure par jour.

*Note: 1 heure comprend 10 minutes de condition 1) et 50 minutes de condition 2).

Piles de protection mémoire: environ 2 ans (avec piles principales en place)

Arrêt auto alimentation:

Environ 6 minutes

Gamme des températures ambiantes:

0°C à 40°C (32°F à 104°F)

Dimensions:

19,2H × 142W × 71D mm Pliée

9,6H × 142W × 142D mm Dépliée

Poids:

159 g (5,6 oz) y compris les piles

A		DIM		173
ABS	190	DMS\$		202
ACS	185	Données statistiques		35
Afficher exposant	21	E		
Argument	184	Elément de sortie		151
ASCII	19	Emboîtement		249
ASN	185	END		145
ATN	185	EOX		198
B		EOY		199
BASIC	89	ERASE		175
BEEP	170	Erreur		107
Boucle	159	&H		204
Bouton de remise à zéro	11	EXP		188
C		Exposant		3, 21
Calcul de nombres complexes	67	Expression		134
Calcul matriciel	53	Expression alphanumérique		134
CLEAR	144	Expression de comparaison		156
Code erreur	107	Expression de régression linéaire	40, 198	
Coefficients de corrélation	39	Expression numérique		134
Coefficient de régression linéaire	40, 198	Extension des variables		98, 171
Commandes de la DATA BANK	206	F		
Commande touche-unique	3	FACT		195
Contrôle de parité	143	Factoriel		26, 195
Coordonnées polaires	33, 193	Fonction de Gauss		25, 191
Coordonnées rectangulaires	33, 193	Fonctions caractères		180
Correction	138	FOR ~ TO ~ [STEP] NEXT		158
COS	184	FRAC		192
Courbe de régression	40	G		
CSR	152	GET		167
CUR	189	GOSUB		159
Curseur	5, 16	GOTO		153
D		GRA (grades)		8, 184
DATA	162	H		
DEFM	171	HEX\$		203
DEG	201	HYPACS		187
Degrés	7, 185			

HYPASN	187	Mode MEMO IN	8, 218
HYPATN	187	Mode PRINT	8, 117
HYPCOS	186	Mode RUN	7, 16, 93
HYP SIN	186	Mode SHIFT-IN	3, 9
HYPTAN	186	Mode TRACE	108
I		Mode WRT	7, 92
IF ~ THEN	156	Mot de passe	139
INPUT	148	Moyenne des différences	38
Instructions	90	Multi-instruction	147
JNT	191	N	
Intégration numérique	78	NCR	197
K		NEW [ALL]	136
KEY\$	150	NEW #	206
L		Nom de fichier	112, 141
LEN	180	Nombre au hasard	25, 200
LET	146	Nombre de chiffres après la virgule	34, 179
LIST	138	Nombre de chiffres significatifs	34, 179
LIST #	206	Nombre hexadécimal	5, 32
LIST V	138	Nombre d'octets	91, 103
LN	188	NPR	196
LOAD [ALL]	142	Numéro de ligne	90, 103
LOAD #	209	O	
LOG	188	ON ~ GOSUB	161
M		ON ~ GOTO	154
Mantisse	23	Opération manuelle	24
Message	49, 148	Ordre de priorité	23
Message erreur	107, 248	P	
Méthode de Newton	74	Paramètre	134
Méthode de Simpson	78	Partie décimale	31
MID\$	181	Partie entière	31
Mise au point	107	PASS	139
MODE	176	Périphériques	10, 109
Mode direct	8	Permutation	26, 196
Mode EDIT	223, 224	POL	194
Mode extension	8, 9	PRINT	151

Programme principal	104	Touches de déplacement du curseur	5
PUT	166	Touche effacer	6
		Touche égal (=)	4
R		Touche entrée de données statistiques	5
RAD (radians)	7, 184	Touche insérer	6, 19
RAN #	200	Touches fonctions	4, 27
READ	163	Touches numériques	3
READ #	210		
REC	193	U	
REM	147	Unité d'angle	7
RESTORE	165	Utilitaires incorporés	51
RESTORE #	212		
RETURN	161	V	
RND	192	VAL	182
RUN	137	Valeur estimée	41, 198
		Variables	22, 95
S		Variables de caractère	95, 148
SAVE [ALL]	141	Variables de caractère exclusives	95, 166
SAVE #	208	Variable de contrôle	158
SET	179	Variables de tableau	95, 173
Signal sonore	170	Variables numériques	95, 148
Seragésimale	25, 31, 201	Variables simples	95
SGN	190	VERIFY	143
SIN	184		
Solution numérique d'équations	74	W	
Sous-programme	104	WRITE #	215
SQR	189		
STAT	177	Z	
STAT CLEAR	177	Zone libre	91, 101
STAT LIST	178		
Statistiques à une variable	38		
Statistiques à variable appariée	38		
STOP	145		
STR\$	183		
Symbole BUZZER	16		
T			
TAN	184		
Terme constant de la régression linéaire			