

INFORMATIQUE

Apprendre
à
programmer

Langage BASIC

C. Lahuc

INFORMATIQUE

Apprendre
à
programmer

Langage BASIC

E. Lahuc

Avec la participation de J. Luisiano

Avant - Propos

Ce manuel s'adresse à toute personne débutante ou possédant quelques notions en informatique.

I Conseils d'utilisation à suivre:

1) Un chapitre forme un tout en lui-même.

Ne vous arrêtez jamais au milieu d'un paragraphe, ou sinon reprenez entièrement ce paragraphe au cours suivant.

2) Si vous éprouvez des difficultés :

revenez en arrière, assimilez convenablement chaque notion avant de passer à la suivante.

3) Ce manuel comporte de nombreuses répétitions nécessaires pour la mémorisation des points importants, pour provoquer un automatisme dans les réponses.

Le passage au chapitre suivant suppose la compréhension et la connaissance complète du précédent.

II Ordinateur et micro-ordinateur

Dans ce manuel nous parlerons indifféremment d'ordinateur ou de

2- micro-ordinateur : les règles de fonctionnement qui les régissent étant identiques. La miniaturisation de l'électronique moderne a permis la construction du micro-ordinateur.

III Quelques précisions supplémentaires :

Les travaux pratiques proposés dans ce manuel permettent une progression rapide dans l'apprentissage de la programmation.

Il fallait donc effectuer un choix parmi les ordinateurs pouvant être utilisés pour la réalisation des différents exercices.

Notre choix s'est porté sur un APPLE (APPLE 2 E ; APPLE 2 C ...). Ce choix n'est pas exhaustif. En fait le langage BASIC est le langage utilisé couramment dans la micro-informatique.

Seuls les chapitres de mise en marche de l'ordinateur, d'initialisation du support magnétique peuvent varier légèrement d'un micro-ordinateur à l'autre.

Sommaire

3.

Chapitre 1	Introduction .	page 5
2	Mise en marche - Premiers pas vers la programmation.	page 9
3	Approfondissement de l'instruction de sortie PRINT .	page 15
4	Premiers pas vers la programmation . Exécution différée des instructions.	page 35
5	Exemple d'initialisation d'une disquette. Sauvegarde d'un programme.	page 45
6	Recherche d'un programme.	page 49
7	des mémoires.	page 51
8	Instruction de sortie INPUT.	page 79
9	Amusons nous un peu .	page 97
10	Instructions IF... THEN ; GOTO ; IF... THEN GOTO ; END .	page 107
11	Instruction FOR ... TO ... NEXT ...	page 127
12	Instructions LEN ... MID \$...	page 143

I Définition matérielle d'un ordinateur:

Un micro-ordinateur se compose :

- de circuits électroniques (circuits intégrés ; microprocesseur) fonctionnant par impulsion électrique (unité élémentaire informatique appelée BIT).
- d'organes d'entrée fournissant les informations au microprocesseur.
- d'organes de sortie recevant les informations du microprocesseur.

1) d'unité élémentaire informatique:

En mémoire ne sont enregistrés que les chiffres 0 et 1 (0 : lampe éteinte ; 1 : lampe allumée).

Un BIT ne peut codifier que deux informations : il faut donc associer plusieurs BITS pour permettre une codification plus poussée.

La majorité des micro-ordinateurs domestiques utilisent une association de 8 BITS (Avec 8 BITS on peut codifier 255 informations différentes.)

2) des mémoires constituées de circuits intégrés:

a) des mémoires mortes : ROM

Les boîtiers ROM contiennent des informations écrites une fois pour toute par le constructeur de l'ordinateur. Elles ne s'effacent jamais même en coupant le courant d'alimentation.

L'utilisateur ne peut ni effacer ni modifier le contenu des ROM.

6.

b) des mémoires vives : RAM

Il s'agit de mémoires dans lesquelles on peut lire et écrire.

Les instructions écrites dans les RAM seront effacées lors de la coupure du courant d'alimentation du micro-ordinateur. Pour éviter que ces instructions soient effacées il faut, avant de couper le courant d'alimentation du micro-ordinateur, enregistrer ces instructions sur un support magnétique (disquette ou cassette).

3) le microprocesseur:

C'est le cœur de l'ordinateur. Il traite les informations fournies par les organes d'entrée. Il est capable d'additionner, soustraire des nombres dans un ordre donné ; de mémoriser les informations.

(Il existe plusieurs types de microprocesseurs se distinguant par leur vitesse d'exécution, la taille des données manipulées ...)

4) les organes d'entrée:

Ils fournissent les informations au microprocesseur, ils codent le langage alphanumérique en langage binaire (0 et 1) dont se sert le micro-ordinateur. Il s'agit du clavier de commande ...

5) les organes de sortie:

Ils décodent le langage binaire dont le micro-ordinateur s'est servi en langage alphanumérique. Ils permettent de voir vos informations et résultats des informations que vous avez fait agir entre elles. Ces informations sont orientées sur le moniteur (temporairement) ou sur l'imprimante (feuille de papier donc définitivement) ou sur une table traçante (graphiques) ou sur un support magnétique.

II Rôle d'un ordinateur, programmation:

7.

1) Rôle d'un ordinateur:

Un ordinateur permet de traiter les informations qu'on lui soumet (d'où le terme informatique) immédiatement ou en les mettant en mémoire pour les conserver. Ces informations peuvent agir les unes sur les autres selon le code que l'on a dicté (programme).

Les informations peuvent agir immédiatement ou peuvent être différées (mémoire), le support de la mémoire étant le disque ou la cassette.

Le clavier de commande permet de dialoguer entre vous et l'ordinateur selon un code préétabli (Basic par exemple).

2) Programme:

Un programme est une suite d'instructions écrites dans un langage assimilable par l'ordinateur et qui commande son fonctionnement d'une manière logique. Le but recherché est l'obtention d'un dessin, la résolution d'un problème, la mémorisation d'informations...

Le programme est écrit par le programmeur.

3) Langage de programmation: le Basic:

Le Basic que nous allons apprendre est un langage couramment utilisé par les ordinateurs individuels. Il y a de nombreuses variétés de Basic: chaque type d'ordinateur individuel en utilise une version légèrement différente.

Le Basic Applesoft que nous utilisons ici est une variété de Basic ayant d'énormes possibilités graphiques.

Apprendre un langage de programmation c'est comme apprendre un second

8. langage : de nouveaux mots (Anglais) et des règles syntaxiques de liaison à connaître . Pour que l'ordinateur réagisse convenablement il faut que les mots que vous composez soient tapés avec une très grande précision , aucune erreur n'est tolérée .

Mise en marche - Premiers pas vers la programmation

I Mise en marche de l'APPLE 2:

- 1) Mettez sous tension l'APPLE et l'écran.

Sur l'écran apparaît:

- 2) Appuyez sur quelques touches du clavier
Que se passe-t-il sur l'écran ?

- 3) Appuyez simultanément sur les touches et

Sur l'écran apparaît:

Vous pouvez dès lors converser avec votre APPLE 2.

II A la découverte de l'instruction PRINT et de la touche

PRINT signifie en Anglais

- 1) Exercice 1:

Tapez:

10.

PRINT " J ' APPRENDS A PROGRAMMER SUR UN
APPLE 2 "

Que se passe-t-il ?

Appuyez sur la touche



(RETURN)

Sur l'écran apparaît

2) Exercice 2:

Tapez:

PRINT " JE POURRAI BIENTOT EFFECTUER SEUL MON
PREMIER PROGRAMME "

Que se passe-t-il ?

Pour que l'instruction soit exécutée je dois appuyer sur



Sur l'écran apparaît

3) Exercice 3:

Tapez:

PRINT " LE CIEL EST VERT ET L'HERBE EST BLEUE "

Que se passe-t-il ?

Pour que l'instruction soit exécutée je dois appuyer sur :



Sur l'écran apparaît


Conclusion de l'exercice 3 :

d'ordinateur écrit ce qui se trouve dans les guillemets que la phrase donnée soit cohérente ou non.

4) Exercice 4 : Nous allons voir si vous savez lire.

Tapez rigoureusement ce qui suit sans rien modifier.

PRINT PROGRAMMER DEVIENT UN AMUSEMENT

Pour que l'instruction soit exécutée je dois appuyer sur 

Sur l'écran apparaît


Parce que

Pour l'ordinateur le chiffre 0 correspond à une valeur fautive.

5) Exercice 5 :

Tapez rigoureusement ce qui suit sans rien modifier.

PRINT L'ORDINATEUR EST TRES SUSCEPTIBLE "

Pour que l'instruction soit exécutée je dois appuyer sur 


Sur l'écran apparaît :

Parce que :

6) Exercice 6:

Tapez rigoureusement ce qui suit sans rien modifier

PRINT " L'APPLE 2 EST MOINS TOLERANT QUE MON
PROFESSEUR DE FRANCAIS .

Pour que l'instruction soit exécutée, je dois appuyer sur 

Sur l'écran apparaît :

Conclusion:

L'ordinateur n'accepte pas :

-
-

Mais il accepte :

Je veillerai cependant à toujours ouvrir et fermer les guillemets.

7) Exercice 7:

Tapez rigoureusement ce qui suit sans rien modifier

PRINT " VOYONS SI TOUS LES ORDRES DOIVENT ETRE
DONNES FIDELLEMENT "

Pour que l'instruction soit exécutée, je dois appuyer sur 

Sur l'écran apparaît :

Parce que:

SYNTAX ERROR : erreur de syntaxe .

8) Conclusion:


a) Pour imprimer une phrase ou un mot

- je tape PRINT
- j'ouvre les guillemets
- j'écris ma phrase (cohérente ou non, avec ou sans faute d'ortho-
-graphie, en Français ou dans une autre langue. L'ordinateur accepte
tout ce qui se trouve dans les guillemets)
- je ferme les guillemets



Remarques:


- Si j'oublie les guillemets ou si j'oublie seulement d'ouvrir
les guillemets le chiffre 8 apparaît.
- Lorsque je commets une faute d'orthographe sur l'instruction
PRINT l'ordinateur écrit SYNTAX ERROR et émet un
BIP.

b) Tous les ordres donnés à l'ordinateur (ici PRINT) pour effectuer un
certain travail doivent être transcrits fidèlement. Il ne faut ni commettre
d'erreur sur l'orthographe de l'instruction, ni oublier d'ouvrir et de fermer les
guillemets.

c) Lorsque je tape l'instruction PRINT l'ordinateur enregistre mon ordre
mais il attend que je lui ordonne de l'exécuter en appuyant sur la touche 
(return). Pour que l'ordinateur exécute un ordre il ne faut jamais

-14-

oubliez d'appuyer sur la touche  (return). Sans cette touche l'ordinateur ignore si vous avez terminé votre instruction. La touche  est indispensable pour prévenir l'ordinateur de la fin de l'instruction.

Exercez vous dès maintenant à utiliser l'instruction PRINT et la touche . Quand vous serez familiarisé avec ces premières instructions nous passerons au chapitre suivant. Vous pourrez prochainement initialiser votre disquette personnelle.

16.

1) Exercice 1:

Soyez toujours très précis dans vos réponses.

Sur quelles touches dois-je appuyer pour donner l'ordre à l'ordinateur d'imprimer sur l'écran la phrase suivante :

JE POURRAI BIENTOT EFFECTUER SEUL UN PROGRAMME

2) Exercice 2:

Soyez toujours très précis dans vos réponses.

Sur quelles touches dois-je appuyer pour donner l'ordre à l'ordinateur d'imprimer sur l'écran la phrase suivante :

JE SUIS UN ELEVE DU MONT-SAINT-JEAN

3) Exercice 3:

Soyez toujours très précis dans vos réponses.

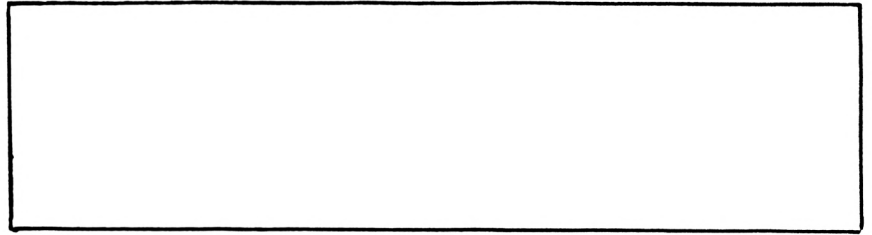
Sur quelles touches dois-je appuyer pour donner l'ordre à l'ordinateur d'imprimer sur l'écran la phrase suivante :

J'HABITE ANTIBES ; JE PASSE MES VACANCES D'HIVER A LA NEIGE

III Découverte de l'instruction HOME

Tapez HOME  (return)

Que se passe-t-il ?




Désormais pour j'utilise l'instruction

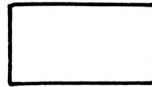
IV Impression d'un nombre, du résultat d'une opération:

1) Impression d'un nombre sur l'écran:


a) exemple:

• Tapez ? " 15 Ø " 

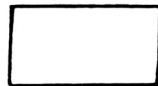
sur l'écran apparaît



En effet n'oublions pas que l'ordinateur imprime tout ce qui se trouve dans les guillemets .

• Tapez ? 15 Ø 

sur l'écran apparaît



b) Conclusion:

Pour demander à l'ordinateur d'imprimer un nombre tapez

<u>PRINT</u>	<u>nombre</u>
<u>ou</u>	
?	<u>nombre</u>

Il est inutile de mettre des guillemets autour du nombre

c) Exercice 1:

Soyez très précis dans vos réponses.

Sur quelles touches dois-je appuyer pour donner l'ordre à l'ordinateur d'imprimer sur l'écran le nombre suivant :

7321

(Attention: il y a deux réponses avec ou sans guillemets)

ou

d) Exercice 2:

Attention: sur l'ordinateur la virgule d'un nombre décimal est remplacée par un point. Ainsi au lieu de taper 13,51 on doit taper 13.51

Soyez toujours très précis dans vos réponses.

Sur quelles touches dois-je appuyer pour donner l'ordre à l'ordinateur d'imprimer sur l'écran le décimal suivant :

1375,722

ou

2) Touches concernant les opérations:

* multiplication

/ division

+ addition

- soustraction

^ élévation à une puissance

3) Impression d'une somme de deux termes :a) exemple :

• Tapez ? " 3 + 4 " ←]

Sur l'écran apparaît :

car l'ordinateur imprime ce qui se trouve dans les guillemets.

Avec les guillemets nous n'obtenons pas le résultat de $3+4$

• Tapez ? 3 + 4 ←]

Sur l'écran apparaît :

On obtient ainsi le résultat de cette somme de deux termes

b) exercice 1 :

Je désire voir affiché sur l'écran le résultat de :

$$732,521 + 9,56$$

Sur quelles touches dois-je appuyer ? Soyez très précis dans vos réponses.

c) exercice 2 :

Je désire voir affiché sur l'écran le résultat de :

$$2,5342 + 999,01$$

Sur quelles touches dois-je appuyer ? Soyez très précis dans vos réponses

4) Impression d'une différence de deux termes :a) exemple :

• Tapez ? " 12 - 5 " ←]

Sur l'écran apparaît

car l'ordinateur imprime ce qui se trouve dans les guillemets .

Avec les guillemets nous n'obtenons pas le résultat de $12 - 5$

• Tapez ? 12 - 5 ←]

Sur l'écran apparaît

On obtient ainsi le résultat de cette différence de deux termes .

b) exercice :

je désire voir affiché sur l'écran le résultat de :

$$7,27 - 3,5$$

Sur quelles touches dois-je appuyer ? Soyez très précis dans vos réponses .

5) Impression d'un produit de deux facteurs :a) exemple :

• Tapez ? " 17 * 9 " ←]

Sur l'écran apparaît

car l'ordinateur imprime ce qui se trouve dans les guillemets .

Avec les guillemets nous n'obtenons pas le résultat de 17×9

• Tapez ? 17 * 9 ←]

Sur l'écran apparaît

On obtient ainsi le résultat de ce produit de deux facteurs.

b) exercice:

je désire voir affiché sur l'écran le résultat de :

$$13,01 \times 4,2$$

Sur quelles touches dois-je appuyer ? Répondez très précis dans vos réponses.

6) Impression d'un quotient:

Attention: n'oubliez pas que la division par zéro n'existe pas.

a) exemple:

• Tapez ? " 921 / 3 " ←]

Sur l'écran apparaît

car l'ordinateur imprime ce qui se trouve dans les guillemets.

Avec les guillemets nous n'obtenons pas le résultat de $921:3$

• Tapez ? 921 / 3 ←]

Sur l'écran apparaît

On obtient ainsi le résultat du quotient de 921 par 3

b) exercice:

je désire voir affiché sur l'écran le résultat de :

22.

$$14,66424 : 3,72$$

Sur quelles touches dois-je appuyer? Soyez très précis dans vos réponses

Vérifiez le résultat donné par l'ordinateur en posant la division de 14,66424 par 3,72 (n'arrêtez pas la division avant le 3^e chiffre après la virgule)

résultat: $14,66424 : 3,72 =$

7) Comment utiliser les nombres négatifs:

Autour des décimales relatifs il n'est pas utile de placer des parenthèses.

Ainsi au lieu de taper:

$$\underline{\quad} \ ? \ \underline{\quad} \ 5.1 \ + \ (\underline{\quad} \ -7.2) \ \leftarrow \]$$

on se contente de taper:

$$\underline{\quad} \ ? \ \underline{\quad} \ 5.1 \ + \ \underline{\quad} \ -7.2 \ \leftarrow \]$$

Attention une expression mathématique doit contenir autant de parenthèses ouvertes que de parenthèses fermées.

a) exemple :

pour obtenir le résultat du produit de 27,5 par -3,77 vous devez

taper:

$$\underline{\quad} \ ? \ \underline{\quad} \ 27.5 \ * \ \underline{\quad} \ -3.77 \ \leftarrow \]$$

b) exercice :

je désire voir affiché sur l'écran le résultat de :

$$14,66424 : (-3,72)$$

Sur quelles touches dois-je appuyer? Soyez très précis dans vos réponses

8) Élévation à une puissance:

a) rappel mathématique:

$$5^4 = 5 \times 5 \times 5 \times 5 \quad ; \quad 7^8 = 7 \times 7 \times 7 \times 7 \times 7 \times 7 \times 7 \times 7$$

b) exemple:

Pour obtenir sur l'écran le résultat de 5^8 vous devez taper :

? 5 ^ 8 ←

c) exercice 1:

Je désire voir affiché sur l'écran le résultat de :

$$5^4$$

Sur quelles touches dois-je appuyer ? Soyez très précis dans vos réponses.

Tapez cette instruction sur le clavier, vous obtenez ainsi le résultat de 5^4

vous trouvez: $5^4 =$

d) exercice 2:

Je désire voir affiché sur l'écran le résultat de :

$$7^8$$

Sur quelles touches dois-je appuyer ? Soyez très précis dans vos réponses.

Tapez cette instruction sur le clavier, vous obtenez ainsi le résultat de 7^8

vous trouvez: $7^8 =$

24.

9) Priorités des opérations :

- ① \wedge puissances (de gauche à droite)
- ② $*$ / multiplications et divisions de gauche à droite
- ③ $+$ - additions et soustractions de gauche à droite

10) Exercice : Complétez le tableau suivant :

Calcul à effectuer	Réponse trouvée "à la main"	instruction donnée à l'ordinateur pour trouver la réponse	Réponse trouvée par l'ordinateur
$9 + 11 - 7$			
$8 \times 4,5$			
$4^2 + 1$			
$6 : 4 + 1$			
$5 - 4 : 2$			
$4 : 2 - 2$			
$6 \times (-2) + 6 : 3 + 8$			
$4 + (-6)$			
$(4^2)^3 + 1$			
$2 \times 2 \times 3 + 1$			
$2 \times 2 + 3 \times 3$			
$8 : 2 : 2 : 1$			
$8 \times 2 : 2 + 3 \times 2^2 \times 1$			
$120 : 3 \times 5$			
$15 - 3 : 3$			

11) Rôle des parenthèses :a) Priorité des parenthèses :

les parenthèses modifient l'ordre des opérations. L'ordinateur effectue d'abord les calculs se trouvant à l'intérieur des parenthèses. Il doit y avoir autant de parenthèses ouvertes que de parenthèses fermées.

b) exercice : Complétez le tableau suivant

Calcul à effectuer	Réponse trouvée "à la main"	instruction donnée à l'ordinateur pour trouver la réponse	Réponse trouvée par l'ordinateur
$44 : (2 + 2)$			
$(44 : 2) + 2$			
$3 + (-2 + 3 \times 2)$			
$(3 + (-2) + 3) \times 2$			
$100 : 200 \times (9 - 5)$			
$3 \times (5 - 7) - 2 \times 3$			
$3 \times 5 - (7 - 2) \times 3$			

V Impression d'une expression alphanumérique :1) Exercice 1 :

Effacez l'écran. Pour cela tapez

Tapez rigoureusement ce qui suit sans rien modifier.

? " LES NOTES DE MATHEMATIQUE DE STEPHANE AU PREMIER TRIMESTRE SONT 12,5 ; 9 ET 8,5 . SA MOYENNE EST DONC EGALE A : " ; $(12.5 + 9 + 8.5) / 3$ ←]

26.

Sur l'écran apparaît:

Parce que: d'ordinateur imprime la phrase se trouvant entre guillemets puis le résultat du calcul : $(12,5 + 9 + 8,5) : 3$ (car ce calcul ne se trouve pas entre guillemets).

2) Exercice 2:

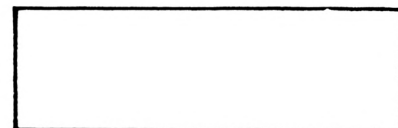
Tapez rigoureusement ce qui suit sans rien modifier:

? " J'AI ACHETE UN LIVRE A 26,75 FRANCS , UNE REGLE
A 5,35 FRANCS ET TROIS STYLOS A 4,25 FRANCS . JE
DISPOSAIS D'UNE SOMME DE 100 FRANCS , J'AI DONC
DEPENSE : " ; $100 - (26.75 + 5.35 + 3 * 4.25)$ ←]

Sur l'écran apparaît

3) Exercice 3:

Effacez l'écran . Pour cela tapez



Tapez rigoureusement ce qui suit sans rien modifier:

? " 3 + 4 = " ; 3 + 4 ←]

Sur l'écran apparaît :

4) Exercice 4 :

Tapez rigoureusement ce qui suit sans rien modifier

? " 13 x 17 = " ; 13 * 17 ←]

Sur l'écran apparaît

remarque : vous n'avez "posé" aucun calcul . L'ordinateur a effectué ce travail à votre place.

5) Exercice 5 :

Maintenant réfléchissez.

Je désire obtenir le résultat de $17 : 272$ sans poser la division.

Sur quelles touches dois-je taper , sachant que la présentation de ce calcul doit être identique à celle des exercices 3 et 4 ?

VI Instruction PRINT TAB :

Effacez l'écran . Pour cela tapez :

28.

1) Exercice 1:

Tapez rigoureusement ce qui suit:

? TAB (5) " JE NE SUIS PLUS EN VACANCES " ←]

Que voyez-vous apparaître sur l'écran ?

2) Exercice 2.

Tapez rigoureusement ce qui suit:

? TAB (1) " JE NE SUIS PLUS EN VACANCES " ←]

Que voyez-vous apparaître sur l'écran ?

3) Exercice 3:

Tapez rigoureusement ce qui suit:

? " JE NE SUIS PLUS EN VACANCES " ←]

Que voyez-vous apparaître sur l'écran ?

4) Exercice 4:

Vous souhaitez donner l'ordre à l'ordinateur d'imprimer votre prénom sur l'écran, mais vous désirez laisser 10 espaces vides devant celui-ci.

Quelle instruction devez-vous taper ?

5) Exercice 5:

Sur une ligne horizontale on peut écrire 40 caractères.

Vous souhaitez donner l'ordre à l'ordinateur d'imprimer votre prénom dans la partie centrale de l'écran (autant de vides à gauche qu'à droite).

Quelle instruction devez-vous taper ?

VII Quelques problèmes de syntaxe:

1) Exercice 1:

Effacez l'écran. Pour cela tapez

Tapez rigoureusement ce qui suit :

? " J ' E T A I S " , " E N V A C A N C E S " ←]

Sur l'écran apparaît

2) Exercice 2:

Tapez rigoureusement ce qui suit :

? " J ' E T A I S " ; " E N V A C A N C E S " ←]

30.

Sur l'écran apparaît:

3) Exercice 3:

Tapez rigoureusement ce qui suit.

? "J'ETAIS", "EN"; "VACANCES" ←]

Sur l'écran apparaît:

4) Exercice 4:

Tapez rigoureusement ce qui suit:

? 3, 7, 11, 15 ←]

Sur l'écran apparaît:

5) Exercice 5:

Tapez rigoureusement ce qui suit:

? 3 ; 7 ; 11 ; 15 ←]

Sur l'écran apparaît

6) Conclusions.

Dans une instruction PRINT l'utilisation:

- de la virgule permet un affichage par colonne des résultats .
- du point virgule ne permet aucune séparation entre les expressions imprimées sur l'écran.

VIII Applications:

31.

Effacez l'écran. Pour cela tapez

1) Exercice 1:

Tapez rigoureusement ce qui suit:

? TAB (3) "NOM: "; TAB (31) "PRENOM: " ←]

Sur l'écran apparaît:

2) Exercice 2:

Tapez rigoureusement ce qui suit:

? "DATE DE NAISSANCE: "; TAB(23) "LIEU DE NAISSANCE:"
←]

Sur l'écran apparaît:

3) Exercice 3:

Tapez rigoureusement ce qui suit:

? 123 ; TAB (6) 567 ; TAB (11) 91Ø11 ←]

Sur l'écran apparaît

IX Récapitulatif:

Avant d'aborder le chapitre suivant, un tour d'horizon rapide des points essentiels étudiés dans ce chapitre semble nécessaire.

1) PRINT et ?:

d'instruction PRINT peut être remplacée par le point d'interrogation ?

PRINT et ? ont la même signification imprimer.

2) Pour imprimer un nombre:

a) Il n'est pas nécessaire d'écrire ce nombre entre guillemets

exemple:

Instruction	Affichage sur l'écran
? 7321 ←]	7321

b) la virgule d'un nombre décimal est remplacée par un point (comme sur une calculatrice).

exemple: ? 13.71 ←]

3) Pour imprimer le résultat d'une opération:

L'ordinateur sait aussi calculer comme une "simple" calculatrice.

Il est indispensable de ne pas mettre de guillemets.

exemple:


Instruction	Affichage sur l'écran
? 7 + 2 ←]	9

4) Pour imprimer une expression alphanumérique:

Une seule instruction suffit pour donner l'ordre à l'ordinateur d'afficher sur une "même ligne" une phrase et le résultat d'un calcul.

la phrase est écrite entre guillemets, le calcul sans guillemets. la phrase et le calcul sont séparés par un point virgule pour permettre l'affichage successif de la phrase et du résultat du calcul sans saut de ligne, sans passage à la ligne.

exemple:

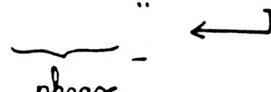
Instruction	<code>? "LA SOMME DES CHIFFRES EST: "; 1+2+3+4+5+6+7+8+9</code> 
Affichage	LA SOMME DES CHIFFRES EST: 45

5) Pour imprimer à un endroit précis sur l'écran:

exemples

a) `? TAB (7) " phrase "` 

Cette instruction permet d'afficher 6 blancs ($7-1=6$) avant d'écrire l'expression qui se trouve entre guillemets.

b) `? TAB (43) " phrase "` 

Sur une même ligne horizontale on peut imprimer 40 caractères.

Cette instruction permet donc de sauter une ligne puis d'afficher 2 blancs avant d'écrire l'expression qui se trouve entre guillemets.

Premiers pas vers la programmation.
Exécution différée des instructions.

I Exécution immédiate, exécution différée des instructions, essai de définition d'un programme :

1) Objetif des précédents chapitres :

Dans les deux premiers chapitres l'ordinateur exécutait immédiatement l'instruction donnée.

Par exemple en tapant sur le clavier :

? " JE SUIS EN COURS D'INFORMATIQUE " ←]

l'ordinateur exécute immédiatement cet ordre, il obéit sans délai à cet ordre en imprimant sur l'écran :

JE SUIS EN COURS D'INFORMATIQUE

2) Objetif de ce chapitre et des chapitres suivants :

Parfois il est très utile de ne pas avoir une exécution immédiate d'une instruction. Il faut donc pouvoir mémoriser (mettre en mémoire) les instructions pour retarder leur exécution. On obtient ainsi une exécution différée des instructions.

3) Tentative de définition d'un programme :

Un programme est constitué d'une liste d'instructions envoyées, par l'intermédiaire du clavier, à l'ordinateur dans un ordre donné. Ces instructions sont mises en mémoire dans l'ordinateur pour obtenir une exécution différée.

4) Comment obtenir une exécution différée d'instructions:

a) Il est indispensable de s'assurer que la mémoire de l'ordinateur est vide (en d'autres termes qu'il ne reste pas dans la mémoire des instructions d'un précédent programme).

Pour effacer la mémoire centrale de l'ordinateur tapez :

NEW ←]

b) Pour savoir dans quel ordre les instructions doivent être exécutées par l'ordinateur chaque instruction doit être précédée d'un numéro de ligne (ce numéro est un entier naturel).

Attention: l'ordinateur exécutera les instructions dans l'ordre croissant de leur numéro de ligne.

Veillez à ne pas écrire deux fois le même numéro de ligne. Après avoir tapé la première instruction précédée de son numéro de ligne tapez sur ←] pour envoyer cette instruction à l'ordinateur.

Suis tapez la seconde instruction (n'oubliez pas le numéro de ligne).

.....

Enfin tapez la dernière instruction puis ←]

c) Lorsque toutes les instructions sont tapées pour demander l'exécution de ces instructions tapez :

RUN ←]

5) Avantages de l'exécution différée des instructions:

a) Il est possible de rajouter des instructions : il suffit de rajouter une instruction avec un numéro de ligne permettant de faire exécuter cette instruction au moment opportun.

b) Il est possible de corriger une instruction : il suffit de taper de nouveau le numéro de ligne de l'instruction à corriger et d'apporter la correction. Le numéro de ligne étant le même la précédente instruction est effacée, la nouvelle est automatiquement mémorisée à sa place.

c) Il est possible de faire exécuter plusieurs fois de suite un programme. Pour cela il suffit de taper de nouveau :

RUN ←]

RUN signifie : je demande à l'ordinateur de revenir à la première ligne d'instruction de ce programme, puis d'exécuter ces instructions par ordre de numéro croissant.

6) Comment revoir les instructions :

Il est possible de revoir les instructions d'un programme, afin d'apporter (par exemple) des modifications à ce programme.

Pour cela, il suffit de taper :

LIST ←]

7) Comment conserver un programme :

Tout programme constitué en suivant les conseils précédents est automatiquement effacé lors de la coupure du courant d'alimentation.

Il est impossible de conserver un programme en l'absence d'un support magnétique (la disquette pour l'APPLE 2).

Essayons de nous familiariser avec chacune des instructions étudiées dans ce chapitre : NEW ; RUN ; LIST ...

Pour cela passons au paragraphe suivant.

38.

II Etude d'un programme:

1) Exemple: lisez attentivement ce qui suit :

NEW ←]

1Ø HOME ←]

2Ø ? TAB (3) "NOM: "; TAB (15) "PRENOM: "; TAB (29)

"CLASSE: " ←]

3Ø ? ←]

4Ø ? TAB (3) " DATE DE NAISSANCE: " ←]

5Ø ? ←]

6Ø ? TAB (3) " ADRESSE DES PARENTS: " ←]

2) Interprétation du programme:

Essayons de suivre "pas à pas" le travail effectué par l'ordinateur.

Reprenons le programme précédent.

- Pour demander à l'ordinateur d'exécuter les instructions de ce programme je tape :

- Sur l'écran apparaît :

<u>Instructions</u>	<u>Travail effectué par l'ordinateur</u>
NEW ←]	le programme précédent s'efface de la mémoire centrale de l'ordinateur.
1 Ø HOME ←]	l'écran s'efface, l'ordinateur passe à l'instruction suivante.
2 Ø ? TAB(3) "NOM:" ; TAB(15) "PRENOM:" ; TAB(29) "CLASSE:" ←]	l'ordinateur écrit au 3 ^e intervalle NOM: puis au 15 ^e intervalle PRENOM: puis au 29 ^e intervalle CLASSE: . Il passe ensuite à l'instruction suivante.
3 Ø ? ←]	l'ordinateur saute une ligne (il imprime... rien) . Il passe à l'instruction suivante.
4 Ø ? TAB(3) "DATE DE NAISSANCE:" ←]	au 3 ^e intervalle il écrit : DATE DE NAISSANCE: il passe à l'instruction suivante.
5 Ø ? ←]	l'ordinateur saute une ligne puis passe à l'instruction suivante.
6 Ø ? TAB(3) "ADRESSE DES PARENTS:" ←]	au 3 ^e intervalle il écrit : ADRESSE DES PARENTS:

3) Exercice 4:

- Tapez LIST ←]

Sur l'écran apparaît :

40.

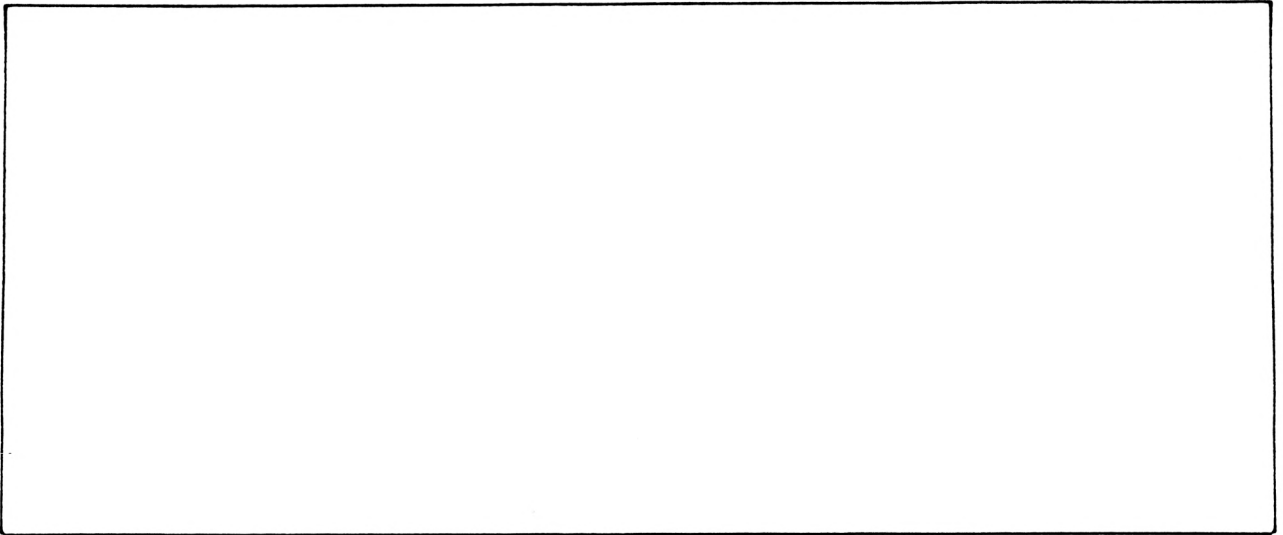
• Tapez :

52 ? TAB (3) " LIEU DE NAISSANCE : " ←]

• Tapez :

RUN ←]

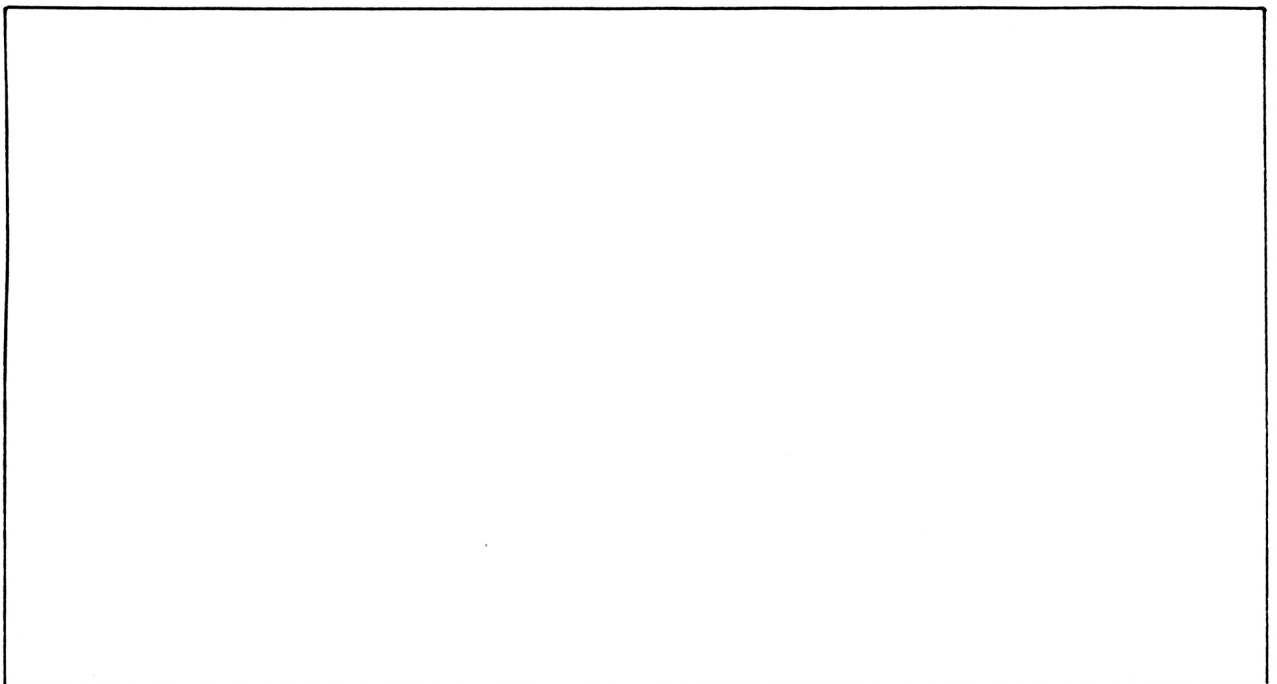
Sur l'écran apparaît :



• Tapez :

LIST ←]

Sur l'écran apparaît :



4) Exercice 2:

- Je désire voir une ligne sautée entre l'inscription :

LIEU DE NAISSANCE :

et l'inscription :

ADRESSE DES PARENTS :

Pour cela je dois rajouter la ligne d'instruction :

- Tapez cette ligne d'instruction .
- Pour demander à l'ordinateur d'exécuter les instructions du programme je tape:

- Pour revoir toutes les instructions du programme je tape:

5) Exercice 3:

(Si vous avez des difficultés pour effectuer cet exercice lisez de nouveau le paragraphe I 5) b))

- Je désire remplacer l'inscription :

LIEU DE NAISSANCE :

par l'inscription :

PROFESSION DES PARENTS :

Pour cela je dois taper la ligne d'instruction:

- Tapez cette ligne d'instruction .
- Pour demander à l'ordinateur d'exécuter les instructions du programme

42-

je tape :

• Pour revoir toutes les instructions du programme je tape :

6) Exercice 4 :

• Je désire supprimer l'inscription :

PROFESSION DES PARENTS :

Pour cela je dois taper :

• Tapez RUN ←]

Vous voyez apparaître sur l'écran deux lignes sautées entre l'inscription :

DATE DE NAISSANCE :

et l'inscription :

ADRESSE DES PARENTS :

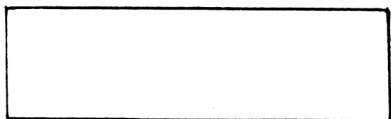
Pour retrouver une seule ligne sautée entre ces deux inscriptions (c'est à dire pour revenir au programme du paragraphe II 2) je dois taper :

7) Exercice 5 :

• Tapez HOME ←]

Que se passe-t-il ?

des instructions du programme sont-elles effacées ? Pour s'en assurer
essayons de revoir toutes les instructions du programme en tapant :



Conclusion :

d'instruction HOME efface l'écran . Mais l'instruction HOME
n'efface pas les instructions d'un programme .

- Réfléchissez . Comment effacer définitivement le programme de la
mémoire de l'ordinateur sans avoir besoin de couper le courant d'alimentation ?
(Aidez vous si nécessaire du paragraphe I 4) a))

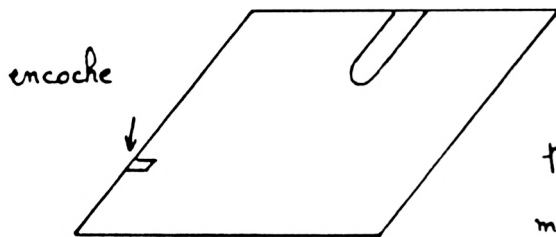
Pour effacer définitivement le programme de la mémoire de l'ordinateur
je doit taper :

Exemple d'initialisation d'une disquette.
Sauvegarde d'un programme.

de procédé permettant d'initialiser une disquette varie d'une marque d'ordinateur à l'autre .

I Exemple d'initialisation d'une disquette sur un APPLE 2 e:

1) représentation d'une disquette:



Si vous obtenez l'encoche vous protégez votre disquette contre des modifications maladroites.

2) Méthode à suivre pour initialiser votre disquette sur un APPLE 2 e:

- ① Eteindre l'ordinateur.
- ② Introduire une disquette déjà initialisée dans l'APPLE.
- ③ Mettre l'ordinateur sous tension.
- ④ Enlever la disquette déjà initialisée.
- ⑤ Introduire à la place la disquette à initialiser.
- ⑥ Effectuer un programme. Ce programme s'affichera sur l'écran chaque fois que vous mettez l'APPLE sous tension avec votre disquette.

Par exemple:

NEW ←]

5 HOME ←]

10 ? TAB(3) " CETTE DISQUETTE APPARTIENT A : "

46.

```
15? ← ]  
20? TAB(3) " DUPONT PASCAL " ← ]  
INIT HELLO ← ]
```

Le programme est sauvegardé sous le nom de HELLO. La dernière instruction est indispensable.

Éteignez l'ordinateur, attendez quelques secondes, mettez de nouveau l'APPLE sous tension : le programme HELLO est exécuté.

Votre disquette est initialisée.

3) Remarque :

L'initialisation d'une disquette est différente sur un APPLE 2 C.

II Affichage de tous les programmes enregistrés sur une disquette :

Tapez :

```
CATALOG ← ]
```

Le nom du programme HELLO (seul programme actuellement sur la disquette) apparaît sur l'écran.

III Sauvegarde des programmes sur une disquette initialisée :

1) Reprenez le programme du chapitre précédent. Vous pouvez le conserver sur votre disquette initialisée, pour cela :

① tapez de nouveau le programme.

② choisissez un nom de programme, par exemple :

RENSEIGNEMENT

③ tapez :

SAVE RENSEIGNEMENT ←]

Votre programme est conservé sous le nom RENSEIGNEMENT

2) Tapez :

CATALOG ←]

Vous voyez apparaître sur l'écran deux noms de programmes :

HELLO et RENSEIGNEMENT

3) . Pour "rappeler" , pour remettre en mémoire le programme

RENSEIGNEMENT tapez :

LOAD RENSEIGNEMENT ←]

. Pour revoir la liste des instructions de ce programme tapez :

LIST ←]

. Pour demander à l'ordinateur d'exécuter ce programme tapez :

RUN ←]

4) Pour effacer , pour enlever le programme RENSEIGNEMENT de la mémoire de l'ordinateur tapez :

DELETE RENSEIGNEMENT ←]

5) . Tapez :

CATALOG ←]

Sur l'écran apparaît uniquement le nom du programme HELLO

48.

- "chargez" le programme HELLO dans la mémoire de l'ordinateur, par cela tapez :

LOAD HELLO ←

- Pour revoir la liste des instructions de ce programme tapez :

LIST ←

- Vous pouvez rajouter une instruction à ce programme. Par exemple :

15 ? ←

20 ? " J'APPRENDS A PROGRAMMER " ←

- N'oubliez pas si vous souhaitez mémoriser ces instructions dans le programme HELLO de taper :

SAVE HELLO ←

IV En bref :

- d'instruction SAVE permet de conserver un programme.
- d'instruction CATALOG permet de faire apparaître les noms de tous les programmes conservés sur une disquette.
- d'instruction DELETE permet d'effacer définitivement un programme.
- d'instruction LOAD permet de "charger" un programme dans la mémoire de l'ordinateur.

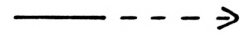
Plusieurs programmes seront proposés dans ce manuel. A vous de conserver ceux qui vous intéressent.

Attention deux programmes différents doivent porter des noms différents.

Recherche d'un programme

49-

Réalisez le programme permettant de voir apparaître en haut de l'écran la représentation ci-après.



8 video

REPRESENTATION

DE QUELQUES

FIGURES

PLANES

1 ligne sautée

TRAPEZE ISOCELE

1 ligne sautée

1 vide

CARRÉ

1 ligne sautée

RECTANGLE

PARALLELOGRAMME

TRAPEZE ISOCELE

3 video

4 video

Coran du moniteur

Ce chapitre ne nécessite pas l'utilisation d'une disquette (ou support magnétique)

A. Introduction:

Lorsque vous décidez de conserver des documents vous les rangez (ordonnez) dans des pochettes, tiroirs, casiers...

L'ordinateur possède lui aussi des "tiroirs" appelés mémoires prêts à recevoir des renseignements importants. Vous pouvez ainsi conserver, mémoriser des valeurs ou des renseignements qui seront utilisés ultérieurement.

Les mémoires ne peuvent pas rester "anonymes", elles doivent obligatoirement porter des noms différents permettant de les différencier et de les utiliser.

B. Mémorisation des nombres, des résultats d'opérations:

I Comment distinguer les mémoires numériques:

1. Quelques noms de mémoires numériques:

Une mémoire numérique peut être désignée par un nom quelconque (ayant un sens ou non) pourvu que ce nom commence par une lettre.

exemples de noms de mémoires numériques:

A ; B ; C ; X ; Y ; Z (une lettre quelconque de l'alphabet)

$A\emptyset; A1; A2; \dots; A9; B\emptyset; B1; \dots; B9; \dots; Z\emptyset; Z1; \dots; Z9$ (une lettre de l'alphabet suivie d'un chiffre $\emptyset; 1; 2; 3; 4; 5; 6; 7; 8$ ou 9).

$AA; AB; AC; \dots; AZ; BA; BB; BC; \dots; BZ; \dots; ZA; ZB; ZC; \dots; ZZ$
(un mot constitué de deux lettres quelconques de l'alphabet)

SOMME; POINTS; JOUEUR; \dots (n'importe quel mot commençant par une lettre, ayant un sens ou non).

2) Quelques précisions supplémentaires:

Le Basic ne prend en considération que les deux premiers "caractères" de chaque nom de mémoire. Ainsi pour l'ordinateur les mots:

ECOLE; ECOLIER; ECOLOGIE; ECRIVAIN \dots

se rapportent tous à la même mémoire.

3) Quelques erreurs à éviter:

Certains noms de mémoires ne sont pas autorisés car ils comprennent un mot possédant une signification très précise. Par exemple le mot PRINT.

Ainsi un nom de mémoire ne peut pas commencer par PRINT.
Par exemple la mémoire PRINTEMPS n'existe pas.

II Différentes possibilités d'utilisation des mémoires numériques:

1) Conservier, mémoriser:

Conservier est le rôle essentiel d'une mémoire.

a) Conserver les nombres:

• Tapez $A = 76$ ←]
 Que se passe-t-il ?

la valeur 76 n'est pas imprimée sur l'écran, elle est mémorisée, envoyée dans le " tiroir " A

Interprétation:

<u>Avant l'instruction</u>	<u>Après l'instruction</u>
<input type="text"/> mémoire A vide	<input type="text" value="76"/> mémoire A

• Tapez $? A$ ←]

Sur l'écran apparaît:

• Tapez $? "A"$ ←]

Sur l'écran apparaît:

• Tapez $? "A = "; A$ ←]

Sur l'écran apparaît:

• Tapez $A = 361$ ←]

Il ne se passe rien sur l'écran. la valeur 361 est envoyée dans la mémoire A à la place de la valeur 76. le nombre 76 est definitivement effacé.

Interprétation:

<u>Avant l'instruction</u>	<u>Après l'instruction</u>
<div style="border: 1px solid black; width: 40px; height: 20px; margin: 0 auto; display: flex; align-items: center; justify-content: center;">76</div> mémoire A	<div style="border: 1px solid black; width: 40px; height: 20px; margin: 0 auto; display: flex; align-items: center; justify-content: center;">361</div> mémoire A

Vérification:

• Tapez ? A ←]

Sur l'écran apparaît:

• Tapez ? "A=" ; A ←]

Sur l'écran apparaît:

b) Conserver les résultats d'opérations

• Tapez B = 36 + 39.5 ←]

Que se passe-t-il ?

L'ordinateur a effectué la somme $36 + 39,5$; il a trouvé $75,5$.

Il a ensuite envoyé, mémorisé le résultat $75,5$ en mémoire B

Interprétation:

<u>Avant l'instruction</u>	<u>Après l'instruction</u>	
	<u>calcul effectué</u>	<u>mémorisation</u>
<div style="border: 1px solid black; width: 60px; height: 30px; margin: 0 auto;"></div> mémoire B vide	$36 + 39,5 = 75,5$	<div style="border: 1px solid black; width: 60px; height: 30px; margin: 0 auto; display: flex; align-items: center; justify-content: center;">75.5</div> mémoire B

• Tapez ? B ←]

Sur l'écran apparaît :

• Tapez ? "B" ←]

Sur l'écran apparaît :

• Tapez ? "B = " ; B ←]

Sur l'écran apparaît :

• Tapez ? A ←]

Sur l'écran apparaît :

Vous pouvez constater que 364 est toujours mémorisé en mémoire A.

• Tapez ? C ←]

Sur l'écran apparaît :

Effectivement nous n'avons conservé aucun nombre en mémoire C.

• Tapez D = 36 * 10 - 37 + 3 * 2 / 4 ←]

Que se passe-t-il ?

L'ordinateur a effectué $36 \times 10 - 37 + 3 \times 2 : 4$; il a trouvé 324,5.

56.

Il a ensuite envoyé, mémorisé 324,5 en mémoire D

Interprétation :

<u>Avant l'instruction</u>	<u>Après l'instruction</u>	
	<u>Calcul effectué</u>	<u>mémorisation</u>
<div style="border: 1px solid black; width: 60px; height: 20px; margin: 0 auto;"></div> mémoire D vide	$36 \times 10 - 37 + 3 \times 2 : 4 = 324,5$	<div style="border: 1px solid black; width: 60px; height: 20px; margin: 0 auto; text-align: center;">324,5</div> mémoire D

• Tapez ? D ←]

Sur l'écran apparaît :

Vérification du calcul "à la main" :

$36 \times 10 =$

$3 \times 2 =$

$(3 \times 2) : 4 = \dots : 4 =$

$36 \times 10 - 37 = \dots - 37 =$

$36 \times 10 - 37 + 3 \times 2 : 4 =$

c) Ajouter, soustraire, multiplier, diviser, faire agir les mémoires entre elles :

• Tapez ? A ←]

Sur l'écran apparaît :

• Tapez ? B ←]

Sur l'écran apparaît :

• Tapez ? A+B ←]

Sur l'écran apparaît :

Que se passe-t-il ?

d'ordinateur a additionné le contenu des deux mémoires A et B . Il a imprimé alors le résultat de cette opération .

Attention : il n'y a pas conservation de ce résultat .

• Tapez E = A + B ←]

Que se passe-t-il ?

d'ordinateur a additionné le contenu des deux mémoires A et B . Il a conservé, mémorisé le résultat en mémoire E .

Interprétation :

<u>Avant l'instruction</u>	<u>Après l'instruction</u>	
	<u>Calcul effectué</u>	<u>Mémorisation</u>
<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 5px; text-align: center;">361 A</div> <div style="border: 1px solid black; padding: 5px; text-align: center;">75,5 B</div> <div style="border: 1px solid black; padding: 5px; text-align: center;"> E vide</div> </div>	$361 + 75,5 = 436,5$	<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 5px; text-align: center;">361 A</div> <div style="border: 1px solid black; padding: 5px; text-align: center;">75,5 B</div> <div style="border: 1px solid black; padding: 5px; text-align: center;">436,5 E</div> </div>

• Tapez ? E ←]

Sur l'écran apparaît :

• Tapez ? D * C ←]

Sur l'écran apparaît

58.

Interprétation de ce résultat :

L'ordinateur a effectué le produit du contenu de la mémoire D par le contenu de la mémoire C.

Contenu de D : 324,5

Contenu de C : \emptyset (aucun nombre n'a jamais été conservé en C)

$$324,5 \times 0 =$$

L'ordinateur a imprimé le résultat de cette opération sur l'écran.

Attention : Il n'y a pas conservation de ce résultat.

• Tapez ? A - B ←]

Sur l'écran apparaît :



Interprétation de ce résultat :

Contenu de la mémoire A : 361

Contenu de la mémoire B : 75,5

$$361 - 75,5 =$$

L'ordinateur a imprimé le résultat de cette opération sur l'écran.

Attention : Il n'y a pas conservation de ce résultat.

• Tapez F = A - B ←]

Que se passe-t-il ?

L'ordinateur a soustrait le contenu de la mémoire B au contenu de la mémoire A.

Il a conservé, mémorisé le résultat en mémoire F.

Interprétation:

<u>Avant l'instruction</u>	<u>Après l'instruction</u>	
	<u>Calcul effectué</u>	<u>mémorisation</u>
<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 5px; text-align: center;">361 A</div> <div style="border: 1px solid black; padding: 5px; text-align: center;">75,5 B</div> <div style="border: 1px solid black; padding: 5px; text-align: center;"> F vide</div> </div>	$361 - 75,5 = 285,5$	<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 5px; text-align: center;">361 A</div> <div style="border: 1px solid black; padding: 5px; text-align: center;">75,5 B</div> <div style="border: 1px solid black; padding: 5px; text-align: center;">285,5 F</div> </div>

• Tapez ? F ←]

Sur l'écran apparaît:

• Tapez G = A - 61 ←]

Que se passe-t-il?

l'ordinateur a enlevé 61 au contenu de la mémoire A. Il a conservé, mémorisé le résultat en mémoire G.

Interprétation:

<u>Avant l'instruction</u>	<u>Après l'instruction</u>	
	<u>Calcul effectué</u>	<u>mémorisation</u>
<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 5px; text-align: center;">361 A</div> <div style="border: 1px solid black; padding: 5px; text-align: center;"> G vide</div> </div>	$361 - 61 = 300$	<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 5px; text-align: center;">361 A</div> <div style="border: 1px solid black; padding: 5px; text-align: center;">300 G</div> </div>

• Tapez ? G ←]

Sur l'écran apparaît:

• Tapez ? 2 * G ←]

Sur l'écran apparaît:

Interprétation de ce résultat:

Contenu de la mémoire G: 300

$$2 \times 300 =$$

d'ordinateur a imprimé le résultat de cette opération sur l'écran.

Attention: il n'y a pas conservation de ce résultat.

• Tapez H = F * G ←]

Que se passe-t-il ?

d'ordinateur a multiplié le contenu de la mémoire F par le contenu de la mémoire G.

Il a conservé, mémorisé le résultat en mémoire H

Interprétation:

<u>Avant l'instruction</u>	<u>Après l'instruction</u>	
	<u>Calcul effectué</u>	<u>Mémorisation</u>
<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 5px; text-align: center;">285,5 F</div> <div style="border: 1px solid black; padding: 5px; text-align: center;">300 G</div> <div style="border: 1px solid black; padding: 5px; text-align: center;"> H vide</div> </div>	$285,5 \times 300 = 85650$	<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 5px; text-align: center;">285,5 F</div> <div style="border: 1px solid black; padding: 5px; text-align: center;">300 G</div> <div style="border: 1px solid black; padding: 5px; text-align: center;">85650 H</div> </div>

• Tapez ? H ←]

Sur l'écran apparaît

• Tapez ? G / 3 ←]

Sur l'écran apparaît :

Interprétation de ce résultat :

contenu de la mémoire G : 300

$$300 : 3 = 100$$

l'ordinateur a imprimé le résultat de cette opération sur l'écran

Attention : Il n'y a pas conservation de ce résultat.

• Tapez L = G / 3 ←]

Que se passe-t-il ?

l'ordinateur a divisé le contenu de la mémoire G par 3.

Il a conservé, mémorisé le résultat en mémoire L.

Interprétation :

Avant l'instruction	Après l'instruction	
	<u>Calcul effectué</u>	<u>Mémorisation</u>
<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="text-align: center;"> <div style="border: 1px solid black; padding: 5px; width: 40px; height: 40px; display: flex; align-items: center; justify-content: center;">300</div> <p>G</p> </div> <div style="text-align: center;"> <div style="border: 1px solid black; width: 40px; height: 40px; margin: 0 auto;"></div> <p>L vide</p> </div> </div>	$300 : 3 = 100$	<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="text-align: center;"> <div style="border: 1px solid black; padding: 5px; width: 40px; height: 40px; display: flex; align-items: center; justify-content: center;">300</div> <p>G</p> </div> <div style="text-align: center;"> <div style="border: 1px solid black; padding: 5px; width: 40px; height: 40px; display: flex; align-items: center; justify-content: center;">100</div> <p>L</p> </div> </div>

• Tapez ? L ←]

Sur l'écran apparaît :

• Tapez $K = A / L$ ←]

Que se passe-t-il ?

d'ordinateur a divisé le contenu de la mémoire A par le contenu de la mémoire L.

(Attention: la division par zéro est impossible).

Il a conservé, mémorisé le résultat en mémoire K.

Interprétation:

<u>Avant l'instruction</u>	<u>Après l'instruction</u>	
	<u>Calcul effectué</u>	<u>Mémorisation</u>
<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="text-align: center;"> <div style="border: 1px solid black; padding: 5px; width: 40px; height: 40px; margin: 0 auto;">361</div> <p>A</p> </div> <div style="text-align: center;"> <div style="border: 1px solid black; padding: 5px; width: 40px; height: 40px; margin: 0 auto;">100</div> <p>L</p> </div> <div style="text-align: center;"> <div style="border: 1px solid black; padding: 5px; width: 40px; height: 40px; margin: 0 auto;"></div> <p>K vide</p> </div> </div>	$361 : 100 = 3,61$	<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="text-align: center;"> <div style="border: 1px solid black; padding: 5px; width: 40px; height: 40px; margin: 0 auto;">361</div> <p>A</p> </div> <div style="text-align: center;"> <div style="border: 1px solid black; padding: 5px; width: 40px; height: 40px; margin: 0 auto;">100</div> <p>L</p> </div> <div style="text-align: center;"> <div style="border: 1px solid black; padding: 5px; width: 40px; height: 40px; margin: 0 auto;">3,61</div> <p>K</p> </div> </div>

2) Effacer le contenu des mémoires:

a) première possibilité:

En coupant le courant d'alimentation.

Attention: En l'absence d'un programme conservé sur une disquette (ou support magnétique) au moyen de l'instruction SAVE: les contenus de vos mémoires s'effacent lors de la coupure du courant d'alimentation.

b) deuxième possibilité:

• Tapez $M = 99$ ←]

Interprétation:

<u>Avant l'instruction</u>	<u>Après l'instruction</u>
<div style="border: 1px solid black; width: 60px; height: 40px; margin: 0 auto;"></div> <p>M vide</p>	<div style="border: 1px solid black; padding: 5px; width: 60px; height: 40px; margin: 0 auto;">99</div> <p>M</p>

- Pour effacer le contenu de la mémoire M

Tapez M = ∅ ←]

Interprétation:

<u>Avant l'instruction</u>	<u>Après l'instruction</u>
<div style="border: 1px solid black; width: 40px; height: 40px; display: flex; align-items: center; justify-content: center; margin: 0 auto;">99</div> <p>M</p>	<div style="border: 1px solid black; width: 40px; height: 40px; display: flex; align-items: center; justify-content: center; margin: 0 auto;">∅</div> <p>M</p>

- Vérification:

Tapez ? M ←]

Sur l'écran apparaît :

- c) Troisième possibilité : en modifiant le contenu de la mémoire.

Cette possibilité a déjà été utilisée dans l'un des paragraphes précédents.

- Tapez J = 27 ←]

Interprétation :

<u>Avant l'instruction</u>	<u>Après l'instruction</u>
<div style="border: 1px solid black; width: 40px; height: 40px; display: flex; align-items: center; justify-content: center; margin: 0 auto;"> </div> <p>J vide</p>	<div style="border: 1px solid black; width: 40px; height: 40px; display: flex; align-items: center; justify-content: center; margin: 0 auto;">27</div> <p>J</p>

Vous pouvez effacer le contenu de la mémoire J en envoyant une autre valeur à conserver à la place de 27.

Par exemple le nombre 27 ne vous intéresse plus mais vous souhaitez mémoriser le nombre 103.

- Tapez J = 103 ←]

Interprétation :

<u>Avant l'instruction</u>	<u>Après l'instruction</u>
<div style="border: 1px solid black; width: 30px; height: 30px; display: flex; align-items: center; justify-content: center; margin: 0 auto;">27</div> J	<div style="border: 1px solid black; width: 30px; height: 30px; display: flex; align-items: center; justify-content: center; margin: 0 auto;">103</div> J

Vous verrez lors des chapitres suivants de très nombreuses utilisations de ces mémoires numériques dans les programmes.

C Mémorisation des expressions alphanumériques :

I Comment distinguer les mémoires alphanumériques :

1) Quelques noms de mémoires alphanumériques :

Une mémoire alphanumérique peut être désignée par un nom quelconque (ayant un sens ou non) pourvu que ce nom commence par une lettre et finisse par le caractère dollar \$

exemples de noms de mémoires alphanumériques :

A\$; B\$; C\$; ; X\$; Y\$; Z\$ (une lettre quelconque de l'alphabet suivie du caractère dollar \$).

A0\$; A1\$; A2\$; ; A9\$; B0\$; ; B9\$; ; Z0\$; ; Z9\$ (une lettre quelconque de l'alphabet suivie d'un chiffre 0, 1, 2, 3, 4, 5, 6, 7, 8 ou 9 puis du caractère dollar \$)

AA\$; AB\$; AC\$; ; AZ\$; BA\$; BB\$; BC\$; ; BZ\$; ZA\$; ; ZZ\$ (un mot constitué de deux lettres quelconques de l'alphabet finissant par le caractère dollar \$).

SOMME\$; POINTS\$; JOUEUR\$

(N'importe quel mot ayant un sens ou non suivi du caractère dollar \$).

2) Quelques précisions supplémentaires:

Le Basic ne prend en considération que les deux premiers caractères de chaque nom de mémoire et le dernier caractère dollar \$.

Ainsi pour l'ordinateur les mots:

ECOLE \$; ECOLIER \$; ECOLOGIE \$; ECRIVAIN \$...

se rapportent tous à la même mémoire alphanumérique.

3) Quelques erreurs à éviter:

Certains noms de mémoires ne sont pas autorisés car ils débutent par un mot possédant une signification précise : par exemple les mots PRINT ; SAVE ; LIST ...

Ainsi un nom de mémoire alphanumérique ne peut pas commencer par PRINT et finir par le caractère dollar \$:

Par exemple la mémoire PRINTEMPS \$ n'existe pas.

De même un nom de mémoire alphanumérique ne peut pas commencer par SAVE et finir par le caractère dollar \$:

la mémoire SAVEUR \$ n'existe pas.

Un nom de mémoire alphanumérique ne peut pas débiter par LIST et finir par le caractère dollar \$:

la mémoire LISTE \$ n'existe pas.

.....

II Différentes possibilités d'utilisation des mémoires alphanumériques:

1) Conserver, mémoriser:

Une mémoire alphanumérique permet de conserver des mots, des phrases, des nombres ou des expressions alphanumériques.

a) Conserver:

• Tapez A \$ = " BLEU " ←]

Que se passe-t-il?

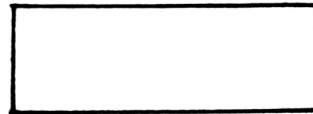
Le mot BLEU n'est pas imprimé sur l'écran; il est mémorisé, envoyé dans la mémoire A \$.

Interprétation:

<u>Avant l'instruction</u>	<u>Après l'instruction</u>
<div style="border: 1px solid black; width: 100px; height: 30px; margin: 0 auto;"></div> mémoire A \$ vide	<div style="border: 1px solid black; padding: 5px; width: 100px; height: 30px; margin: 0 auto;">BLEU</div> mémoire A \$

• Tapez ? A \$ ←]

Sur l'écran apparaît



• Tapez B \$ = " GRIS " ←]

Que se passe-t-il?

Le mot GRIS n'est pas imprimé sur l'écran; il est mémorisé, envoyé dans la mémoire B \$.

Interprétation:

<u>Avant l'instruction</u>	<u>Après l'instruction</u>
<div style="border: 1px solid black; width: 100px; height: 30px; margin: 0 auto;"></div> mémoire B \$ vide	<div style="border: 1px solid black; padding: 5px; width: 100px; height: 30px; margin: 0 auto;">GRIS</div> mémoire B \$

• Tapez ? B \$ ←]

Sur l'écran apparaît :

• Tapez C\$ = "LE CIEL EST : " ←]

Que se passe-t-il ?

la phrase LE CIEL EST : n'est pas imprimée sur l'écran. Elle est mémorisée, envoyée dans la mémoire C\$.

Interprétation :

<u>Avant l'instruction</u>	<u>Après l'instruction</u>
<div style="border: 1px solid black; width: 100%; height: 100%;"></div> <p>mémoire C\$ vide</p>	<div style="border: 1px solid black; width: 100%; height: 100%; padding: 5px;"> LE CIEL EST : </div> <p>mémoire C\$</p>

• Tapez ? C\$ ←]

Sur l'écran apparaît :

• Tapez D\$ = "1Ø" ←]

Que se passe-t-il ?

le nombre 1Ø n'est pas imprimé sur l'écran. Il est mémorisé, envoyé dans la mémoire D\$.

Interprétation :

<u>Avant l'instruction</u>	<u>Après l'instruction</u>
<div style="border: 1px solid black; width: 100%; height: 100%;"></div> <p>mémoire D\$ vide</p>	<div style="border: 1px solid black; width: 100%; height: 100%; padding: 5px;"> 1Ø </div> <p>mémoire D\$</p>

• Tapez E \$ = " 5 " ←]

Que se passe-t-il ?

Le chiffre 5 n'est pas imprimé sur l'écran. Il est mémorisé, envoyé dans la mémoire E \$.

Interprétation:

<u>Avant l'instruction</u>	<u>Après l'instruction</u>
<div style="border: 1px solid black; width: 40px; height: 30px; margin: 0 auto;"></div> mémoire E \$ vide	<div style="border: 1px solid black; width: 40px; height: 30px; display: flex; align-items: center; justify-content: center;">5</div> mémoire E \$

• Tapez ? D \$ ←]

Sur l'écran apparaît:

• Tapez ? E \$ ←]

Sur l'écran apparaît:

Conclusion: Vous pouvez conserver des nombres dans les mémoires alphanumériques.

b) "Ajouter" les mémoires alphanumériques:

Il est possible de faire agir les mémoires alphanumériques entre elles en les ajoutant.

• Tapez ? C \$ + A \$ ←]

Sur l'écran apparaît:

• Tapez ? C\$ + B\$ ←]

Sur l'écran apparaît :

Attention :

• Il n'est pas possible de soustraire, de multiplier, de diviser des mémoires alphanumériques.

• Il n'est même pas possible de taper :

? D\$ - E\$ ←]

Pourtant dans les mémoires D\$ et E\$ se trouvent des nombres.

• Tapez ? D\$ + E\$ ←]

Sur l'écran apparaît :

• Tapez ? E\$ + D\$ ←]

Sur l'écran apparaît :

2) Effacer le contenu des mémoires :

a) première possibilité :

En coupant le courant d'alimentation.

Attention : En l'absence d'un programme conservé sur une disquette (ou support magnétique) au moyen de l'instruction SAVE : les contenus de vos mémoires alphanumériques s'effacent lors de la coupure du courant d'alimentation.

70.

b) deuxième possibilité:

• Tapez M\$ = "EFFACER" ←]

Interprétation :

<u>Avant l'instruction</u>	<u>Après l'instruction</u>
<div style="border: 1px solid black; width: 100px; height: 20px; margin: 0 auto;"></div> <p>M\$ vide</p>	<div style="border: 1px solid black; padding: 5px; margin: 0 auto; text-align: center;">EFFACER</div> <p>M\$</p>

• Pour effacer le contenu de la mémoire M\$

Tapez M\$ = " " ←]

Interprétation :

<u>Avant l'instruction</u>	<u>Après l'instruction</u>
<div style="border: 1px solid black; padding: 5px; margin: 0 auto; text-align: center;">EFFACER</div> <p>M\$</p>	<div style="border: 1px solid black; width: 100px; height: 20px; margin: 0 auto;"></div> <p>M\$ vide</p>

• Vérification:

Tapez ? M\$ ←]

Sur l'écran apparaît

c) troisième possibilité : en modifiant le contenu de la mémoire.

• Tapez J\$ = "GAGNER" ←]

Interprétation :

<u>Avant l'instruction</u>	<u>Après l'instruction</u>
<div style="border: 1px solid black; width: 100px; height: 20px; margin: 0 auto;"></div> <p>J\$ vide</p>	<div style="border: 1px solid black; padding: 5px; margin: 0 auto; text-align: center;">GAGNER</div> <p>J\$</p>

Vous pouvez effacer le contenu de J\$ en envoyant un autre mot ou expression alphanumérique à la place de GAGNER.

Par exemple le verbe GAGNER ne vous intéresse plus mais vous souhaitez mémoriser le verbe PERDRE.

• Tapez J\$ = "PERDRE" ←]

Interprétation:

<u>Avant l'instruction</u>	<u>Après l'instruction</u>
<div style="border: 1px solid black; padding: 5px; display: inline-block;">GAGNER</div> J\$	<div style="border: 1px solid black; padding: 5px; display: inline-block;">PERDRE</div> J\$

Attention:

Il faut toujours écrire les expressions alphanumériques à mémoriser entre guillemets.

D En bref:

N'oublions pas que le programmeur ne peut agir que sur les mémoires vives.

I Dans les mémoires numériques : il est possible :

1) De conserver :

a) des nombres.

b) des résultats d'opérations : Addition, Soustraction, Multiplication, Division, Élévation à une puissance ...

2) D'effacer.

- Les mémoires numériques peuvent "agir" entre elles. Il est possible de les additionner, les soustraire, les multiplier, les diviser...
- Il est possible de modifier le contenu de chacune des mémoires numériques.

II Dans les mémoires alphanumériques : il est possible :

1) De conserver : des nombres, des mots ou des expressions alphanumériques.

2) D'effacer.

- Les mémoires alphanumériques peuvent "agir" entre elles, mais on peut uniquement les "ajouter".
- Il est possible de modifier le contenu de chacune des mémoires alphanumériques.

E. Evitons les pièges :

1) . Tapez POMME = 15 ←
la valeur 15 est mémorisée, envoyée dans la mémoire POMME.

. Tapez ? POMME ←

Sur l'écran apparaît :

• Tapez POIRE = 20 ←]

la valeur 20 est mémorisée, envoyée dans la mémoire POIRE.

• Tapez ? POIRE ←]

Sur l'écran apparaît :

• Tapez ? POMME ←]

Sur l'écran apparaît :

• Réfléchissons : Comment interpréter ce résultat ?

d'ordinateur ne prend en considération que les deux premiers "caractères" de chaque nom de mémoire.

POMME et POIRE désignent donc la même mémoire.

Reprenons une à une les instructions précédentes.

<u>Instruction</u>	<u>Avant l'instruction</u>	<u>Après l'instruction</u>
POMME = 15 ←]	<input type="text"/> mémoire POMME vide	<input type="text" value="15"/> POMME
POIRE = 20 ←]	<input type="text" value="15"/> POIRE ou POMME (car POIRE et POMME désignent la même mémoire)	<input type="text" value="20"/> POIRE ou POMME

2) . Tapez ARBRE = 200 ←]

le nombre 200 est mémorisé, envoyé dans la mémoire ARBRE .

. Tapez ? ARBRE ←]

Sur l'écran apparaît :

. Tapez ART = 150 ←]

le nombre 150 est mémorisé, envoyé dans la mémoire ART (donc dans la mémoire ARBRE à la place de 200).

. Tapez ? ART ←]

Sur l'écran apparaît :

. Tapez ? ARBRE ←]

Sur l'écran apparaît :

. Tapez ARBRE = ART + ARBRE ←]

Interprétation de cette dernière instruction :

<u>Avant l'instruction</u>	<u>Opération effectuée</u>	<u>Après l'instruction</u>
<input type="text" value="150"/> ARBRE ou ART	$150 + 150 = 300$	<input type="text" value="300"/> ARBRE ou ART

. Tapez ARBRE = ART / ARBRE ←]

Interprétation de cette dernière instruction:

<u>Avant l'instruction</u>	<u>Opération effectuée</u>	<u>Après l'instruction</u>
<div style="border: 1px solid black; width: 100px; height: 40px; margin: 0 auto; display: flex; align-items: center; justify-content: center;">300</div> <p>ARBRE ou ART</p>	$300 : 300 = 1$	<div style="border: 1px solid black; width: 100px; height: 40px; margin: 0 auto; display: flex; align-items: center; justify-content: center;">1</div> <p>ARBRE ou ART</p>

• Tapez ? ARBRE ←

Sur l'écran apparaît:

• Tapez ? ART ←

Sur l'écran apparaît:

Vous avez définitivement perdu les valeurs initiales 200 et 150 mémorisées respectivement dans les mémoires ARBRE et ART.

F Échange des contenus de deux mémoires:

I Problème:

• Tapez A = 30 ←

Interprétation:

<u>Avant l'instruction</u>	<u>Après l'instruction</u>
<div style="border: 1px solid black; width: 100px; height: 40px; margin: 0 auto;"></div> <p>A vide</p>	<div style="border: 1px solid black; width: 100px; height: 40px; margin: 0 auto; display: flex; align-items: center; justify-content: center;">30</div> <p>A</p>

76.

• Tapez B = 60 ←]

Interprétation:

<u>Avant l'instruction</u>	<u>Après l'instruction</u>
<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 5px; text-align: center;">30 A</div> <div style="border: 1px solid black; padding: 5px; text-align: center;">B vide</div> </div>	<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 5px; text-align: center;">30 A</div> <div style="border: 1px solid black; padding: 5px; text-align: center;">60 B</div> </div>

On souhaiterait échanger les contenus des mémoires A et B. Pour cela utilisons une mémoire auxiliaire C.

II Résolution du problème:

1) Première étape:

Envoyez le contenu de A dans la mémoire C. Pour cela:

Tapez C = A ←]

Interprétation:

<u>Avant l'instruction</u>	<u>Après l'instruction</u>
<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 5px; text-align: center;">30 A</div> <div style="border: 1px solid black; padding: 5px; text-align: center;">60 B</div> <div style="border: 1px solid black; padding: 5px; text-align: center;">C vide</div> </div>	<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 5px; text-align: center;">30 A</div> <div style="border: 1px solid black; padding: 5px; text-align: center;">60 B</div> <div style="border: 1px solid black; padding: 5px; text-align: center;">30 C</div> </div>

2) Deuxième étape:

Envoyez le contenu de B dans la mémoire A. Pour cela:

Tapez A = B ←]

Interprétation:

<u>Avant l'instruction</u>	<u>Après l'instruction</u>
<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 5px; text-align: center;">30 A</div> <div style="border: 1px solid black; padding: 5px; text-align: center;">60 B</div> <div style="border: 1px solid black; padding: 5px; text-align: center;">30 C</div> </div>	<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 5px; text-align: center;">60 A</div> <div style="border: 1px solid black; padding: 5px; text-align: center;">60 B</div> <div style="border: 1px solid black; padding: 5px; text-align: center;">30 C</div> </div>

3) Troisième étape:

Envoyez le contenu de C dans la mémoire B. Pour cela:

Tapez B = C ←]

Interprétation:

<u>Avant l'instruction</u>			<u>Après l'instruction</u>		
6∅ A	6∅ B	3∅ C	6∅ A	3∅ B	3∅ C

4) Vérification:

• Tapez ? A ←]

Sur l'écran apparaît:

• Tapez ? B ←]

Sur l'écran apparaît:

5) Résumé des différentes instructions utilisées:

A = 3∅ ←]

B = 6∅ ←]

C = A ←]

A = B ←]

B = C ←]

Exercez vous dès maintenant à "manipuler" convenablement les mémoires.

78.

Vous verrez bientôt le rôle essentiel joué par les mémoires dans les programmes.

Dans le prochain chapitre nous étudierons une nouvelle instruction : l'instruction `INPUT`. Cette dernière ne peut être utilisée que dans les programmes.

N'oubliez pas que l'utilisation d'un support magnétique est indispensable pour conserver vos programmes.

L'instruction INPUT permet à l'ordinateur d'attendre une donnée numérique ou alphanumérique en provenance du clavier.

Pour "donner la main" à l'opérateur qui doit frapper les données au clavier, un point d'interrogation (?) apparaît sur l'écran. Dès lors l'ordinateur attend une information.

Cette information est enregistrée, mémorisée dans une mémoire soit numérique, soit alphanumérique.

I Instruction INPUT:

1) Exemple: lisez attentivement ce qui suit :

```

NEW ← ]
5 HOME ← ]
10 INPUT C ← ]
15 ? C ← ]
    
```

2) Interprétation du programme:

Essayons de suivre "pas à pas" le travail effectué par l'ordinateur.

Reprenons le programme précédent.

• Pour demander à l'ordinateur d'exécuter les instructions de ce programme,

je tape:

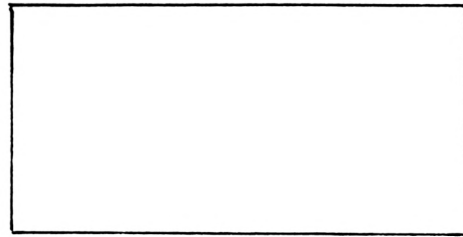
• Sur l'écran apparaît

• d'ordinateur est en attente, il est arrêté à l'instruction 10. Bien qu'il achève l'exécution du programme vous devez taper une donnée numérique.

Ce nombre est enregistré et envoyé dans la mémoire C dès que vous appuyez sur la touche ←]

Essayez : tapez un nombre puis appuyez sur la touche ←]

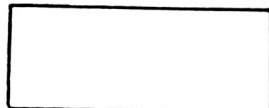
• Sur l'écran apparaît:



• Interprétation:

<u>Instructions</u>	<u>Travail effectué par l'ordinateur.</u>
NEW ←]	le programme précédent s'efface de la mémoire centrale de l'ordinateur.
SHOME ←]	l'écran s'efface, l'ordinateur passe à l'instruction suivante.
10 INPUT C ←]	l'ordinateur <u>attend</u> une donnée. Vous <u>devez taper</u> votre donnée <u>numérique</u> . Elle est enregistrée et mémorisée dans la mémoire C dès que vous appuyez sur la touche ←] .
15 ? C ←]	l'ordinateur imprime la valeur numérique mémorisée en mémoire C

• Pour demander à l'ordinateur d'exécuter une seconde fois ce programme je tape:



3) Exemple : lisez attentivement ce qui suit :

NEW ←]

5HOME ←]

1Ø INPUT R\$ ←]

15 ? R\$ ←]

4) Interprétation de ce programme :

Essayons de suivre " pas à pas " le travail effectué par l'ordinateur.

Reprenons le programme précédent.

• Pour demander à l'ordinateur d'exécuter les instructions de ce programme ,

je tape :

• Sur l'écran apparaît :

• l'ordinateur est en attente , il est arrêté à l'instruction 1Ø . Pour qu'il achève l'exécution du programme vous devez taper une donnée alphanumérique.

Cette phrase, cette expression alphanumérique est enregistrée et envoyée dans la mémoire R\$ dès que vous appuyez sur la touche ←] .

Essayez :

Tapez un mot , une phrase ou une expression alphanumérique puis appuyez sur la touche ←] .

• Sur l'écran apparaît :

• Interprétation:

<u>Instructions</u>	<u>Travail effectué par l'ordinateur</u>
NEW ←]	le programme précédent s'efface de la mémoire centrale de l'ordinateur.
5 HOME ←]	l'écran s'efface, l'ordinateur passe à l'instruction suivante.
10 INPUT R \$ ←]	l'ordinateur <u>attend</u> une donnée alphanumérique. Vous devez taper votre donnée alphanumérique. Elle est enregistrée et mémorisée dans la mémoire R \$ dès que vous appuyez sur la touche ←]
15 ? R \$ ←]	l'ordinateur imprime l'information mémorisée en mémoire R \$

• Pour demander à l'ordinateur d'exécuter une seconde fois ce programme

je tape:

5) Exemple: lisez attentivement ce qui suit :

NEW ←]

5 HOME ←]

10 INPUT A, B, C \$ ←]

15 ? A ←]

20 ? C \$ ←]

25 ? B ←]

6) Interprétation de ce programme:

Essayons de suivre " pas à pas " le travail effectué par l'ordinateur.

Tapons le programme précédent

- Pour demander à l'ordinateur d'exécuter les instructions de ce programme, je tape :

- Sur l'écran apparaît :

- L'ordinateur est en attente, il est arrêté à l'instruction \neq . Pour qu'il poursuive l'exécution du programme vous devez taper une donnée numérique.

Ce nombre est enregistré et envoyé dans la mémoire A dès que vous appuyez sur la touche $\leftarrow \square$.

Essayez : tapez un nombre puis appuyez sur la touche $\leftarrow \square$.

- Sur l'écran apparaît :

- L'ordinateur est de nouveau en attente, il n'a pas achevé l'exécution de l'instruction \neq . Pour qu'il poursuive l'exécution du programme vous devez taper une donnée numérique.

Ce nombre est enregistré et envoyé dans la mémoire B dès que vous appuyez sur la touche $\leftarrow \square$.

Essayez : tapez un nombre puis appuyez sur la touche $\leftarrow \square$.

- Sur l'écran apparaît :

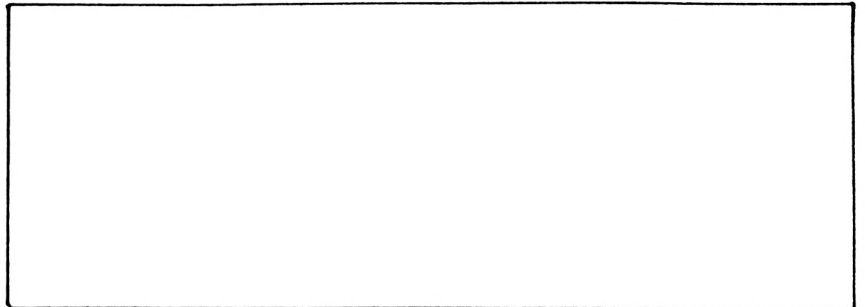
- L'ordinateur est en attente pour la dernière fois, il est arrêté à l'instruction

10. Pour qu'il achève l'exécution du programme vous devez taper une donnée alphanumérique.

Cette phrase, cette expression alphanumérique est enregistrée et envoyée dans la mémoire C \$ dès que vous appuyez sur la touche $\leftarrow \rfloor$.

Essayez : tapez un mot, une phrase ou une expression alphanumérique puis appuyez sur la touche $\leftarrow \rfloor$.

• Sur l'écran apparaît :



• Interprétation :

<u>Instructions</u>	<u>Travail effectué par l'ordinateur</u>
NEW $\leftarrow \rfloor$	le programme précédent s'efface de la mémoire centrale de l'ordinateur.
SHOME $\leftarrow \rfloor$	l'écran s'efface, l'ordinateur passe à l'instruction suivante.
10 INPUT A,B,C \$ $\leftarrow \rfloor$	d'ordinateur <u>attend</u> des données. Vous <u>devez taper</u> : <ul style="list-style-type: none"> • une donnée numérique envoyée dans la mémoire A dès que vous appuyez sur la touche $\leftarrow \rfloor$ • une seconde donnée numérique envoyée dans la mémoire B dès que vous appuyez sur la touche $\leftarrow \rfloor$ • une donnée alphanumérique envoyée dans la mémoire C \$ dès que vous appuyez sur la touche $\leftarrow \rfloor$
15 ? A $\leftarrow \rfloor$	d'ordinateur imprime la donnée mémorisée en mémoire A
20 ? C \$ $\leftarrow \rfloor$	d'ordinateur imprime la phrase mémorisée en mémoire C \$
25 ? B $\leftarrow \rfloor$	d'ordinateur imprime le nombre mémorisé en mémoire B

7) quelques variantes du programme précédent:

a) . lisez attentivement ce qui suit:

NEW ←]

5 HOME ←]

10 INPUT A, B, C\$ ←]

15 ? A, C\$, B ←]

. Tapez le programme précédent.

Pour demander à l'ordinateur d'exécuter les instructions de ce programme,

je tape :

. Sur l'écran apparaît:

. Tapez un nombre, puis appuyez sur la touche ←]

Le nombre est mémorisé en mémoire A.

. Sur l'écran apparaît:

. Tapez un nombre, puis appuyez sur la touche ←]

Le nombre est mémorisé en mémoire B.

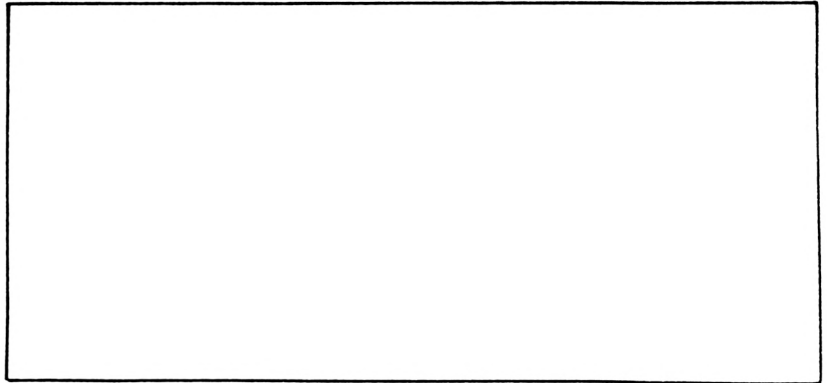
. Sur l'écran apparaît:

. Tapez une phrase ou une expression alphanumérique, puis appuyez sur la touche

←] .

Cette phrase, cette expression est mémorisée en mémoire C\$

• Sur l'écran apparaît:



• Que constatez-vous?

Seul l'affichage sur l'écran est différent. Cette différence provient de l'instruction 15. N'oubliez pas en effet que l'introduction de la virgule entre les mémoires A, B et C\$ provoque un affichage par colonne.

b). lisez attentivement ce qui suit :

NEW ←]

5 HOME ←]

10 INPUT A, B, C\$ ←]

15 ? A ; B ; C\$ ←]

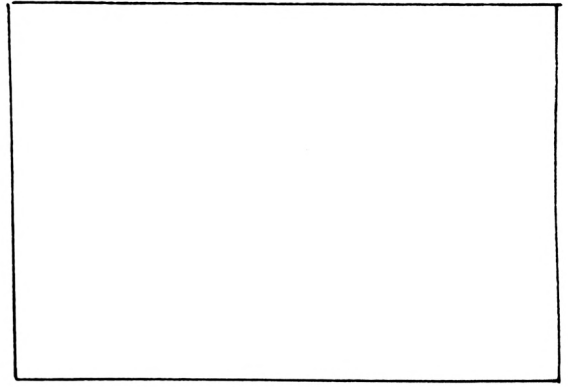
• Tapez le programme précédent.

• Demandez à l'ordinateur d'exécuter les instructions de ce programme.

Effectuez seul chacune des opérations nécessaires pour obtenir une exécution complète du programme ci-dessus.

Si vous éprouvez encore quelques hésitations revenez sur les exemples précédents, vous devez y parvenir sans problème.

- d'écran se présente de la façon suivante :



- Que constatez vous ?

d'affichage sur l'écran est de nouveau différent. Cette différence provient toujours de l'instruction 15. N'oubliez pas en effet que l'introduction du point-virgule entre les mémoires A ; B et C \$ provoque un affichage successif sans aucune séparation des données mémorisées en A ; B et C \$.

- Attention :

la syntaxe est très importante en informatique.

Dans l'instruction 10 :

```
10 INPUT A , B , C $
```

les mémoires A , B et C \$ sont séparées par des virgules.

Vous n'avez pas le droit de remplacer cette instruction par l'instruction :

```
10 INPUT A ; B ; C $
```

le point-virgule n'est pas autorisé dans une instruction INPUT.

Pourtant il peut être employé sans problème dans une instruction PRINT

- 8) Exemple : lisez attentivement ce qui suit :

```
NEW ← ]
```

```
5 HOME ← ]
```

```
10 INPUT " LE MOIS DERNIER J'AI GAGNE : " ; E ← ]
```

```
15 ? " LE MOIS DERNIER VOUS AVEZ GAGNE : " ; E ← ]
```

9) Interprétation de ce programme:

Suivons "pas à pas" le travail effectué par l'ordinateur.

Reprenons le programme précédent.

- Pour demander à l'ordinateur d'exécuter les instructions de ce programme,

je tape :

- Sur l'écran apparaît :

- L'ordinateur est en attente, il est arrêté à l'instruction t_0 . Pour qu'il achève l'exécution du programme vous devez taper une donnée numérique. Ce nombre est envoyé, mémorisé dans la mémoire E dès que vous appuyez sur la touche \leftarrow .

Attention: vous n'avez pas le droit de taper :

7580 F (ou 7580 FR\$ ou 7580 FRANCS).

En effet n'oubliez pas que E est le nom d'une mémoire numérique et non pas alphanumérique.

Essayez : tapez un nombre puis appuyez sur la touche \leftarrow .

- L'écran se présente de la façon suivante :

- Demandez à l'ordinateur d'exécuter une nouvelle fois ce programme. Pour cela tapez :

• Interprétation:

<u>Instructions</u>	<u>Travail effectué par l'ordinateur.</u>
NEW ←]	le programme précédent s'efface de la mémoire centrale de l'ordinateur.
5 HOME ←]	l'écran s'efface, l'ordinateur passe à l'instruction suivante.
10 INPUT "LE MOIS DERNIER J'AI GAGNE: "; E ←]	l'ordinateur imprime l'expression se trouvant entre guillemets. Puis il <u>attend</u> une donnée <u>numérique</u> . Vous devez taper votre donnée numérique. Elle est enregistrée et mémorisée dans la mémoire E dès que vous appuyez sur la touche ←] .
15 ? " LE MOIS DERNIER VOUS AVEZ GAGNE: "; E ←]	l'ordinateur imprime l'expression se trouvant entre guillemets. Puis il imprime l'information mémorisée en mémoire E.

• Remarques:

a) Nous avons observé dans le paragraphe 7) b) l'importance de la virgule pour séparer les noms de mémoires dans une instruction INPUT.

Mais dans une instruction INPUT toute expression entre guillemets doit être séparée du nom de la mémoire par un point-virgule.

b) Observez bien l'instruction 15 ci-dessus : l'expression entre guillemets est séparée du nom de la mémoire par un point-virgule. Lorsque l'ordinateur exécute cette instruction il imprime successivement sans aucune séparation l'expression se trouvant entre guillemets et le nombre mémorisé en mémoire E.

90-

10) Variante du programme précédent :

- lisez attentivement ce qui suit :

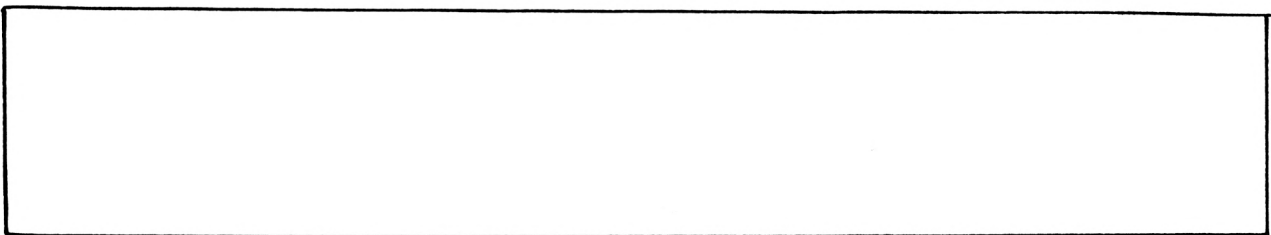
NEW

SHOME

10 INPUT " LE MOIS DERNIER J'AI GAGNE : "; F\$

15 ? " LE MOIS DERNIER VOUS AVEZ GAGNE : "; F\$

- lisez le programme précédent.
- Demandez à l'ordinateur d'exécuter les instructions de ce programme.
Effectuez seul chacune des opérations nécessaires pour obtenir une exécution complète du programme ci-dessus.
Si vous éprouvez toujours des difficultés, n'hésitez pas : revenez sur les exemples précédents.
- d'écran se présente de la façon suivante :



- Que constatez-vous ?

Dans le programme précédent l'ordinateur attendait une donnée numérique. Vous n'avez pas le droit de taper :

7580 F (ou 7580 FRS ou 7580 FRANCS)

puisque E désignait une mémoire numérique.

Dans le programme ci-dessus la mémoire numérique E a été remplacée par la mémoire alphanumérique F\$

Par conséquent lorsque l'ordinateur attendait une donnée vous pouviez taper:

7580 ou 7580 F ou 7580 FRS ou 7580 FRANCS.

Peut être n'avez vous pas remarqué cette différence entre les deux programmes.

Vous pouvez recommencer plusieurs fois l'exécution de ce programme si vous le souhaitez.

II Ébauche de programmes:

1) Programme 1:

Vous devez faire preuve d'un peu d'initiative.

Voici le tableau d'interprétation d'un programme pour lequel vous ne connaissez pas les instructions. A vous de mettre en application votre savoir.

Vous devez reconstituer la première colonne de ce tableau, et par conséquent retrouver les instructions du programme correspondant.

<u>Instructions</u>	<u>Travail effectué par l'ordinateur</u>
	le programme précédent s'efface de la mémoire centrale de l'ordinateur.
	d'écran s'efface, l'ordinateur passe à l'instruction suivante.
	d'ordinateur imprime la phrase se trouvant entre guillemets. Cette phrase est: QUEL EST VOTRE PRENOM? Puis il attend une réponse. Vous devez taper une donnée alphanumérique. Elle est mémorisée envoyée en mémoire L\$ dès que vous appuyez sur la touche ←]
	d'ordinateur imprime la phrase suivante VOUS VOUS APPELEZ: Puis il imprime l'information mémorisée en mémoire L\$.

• Quelques conseils utiles:

Vérifiez si vos instructions débutent par un numéro de ligne.

Attention: il ne doit pas y avoir de numéro de ligne devant NEW.

Pensez à l'importance de la syntaxe (virgule, point-virgule); n'oubliez pas de guillemets.

• Tapez votre programme puis demandez à l'ordinateur d'exécuter votre programme.

• deux cas peuvent se produire:

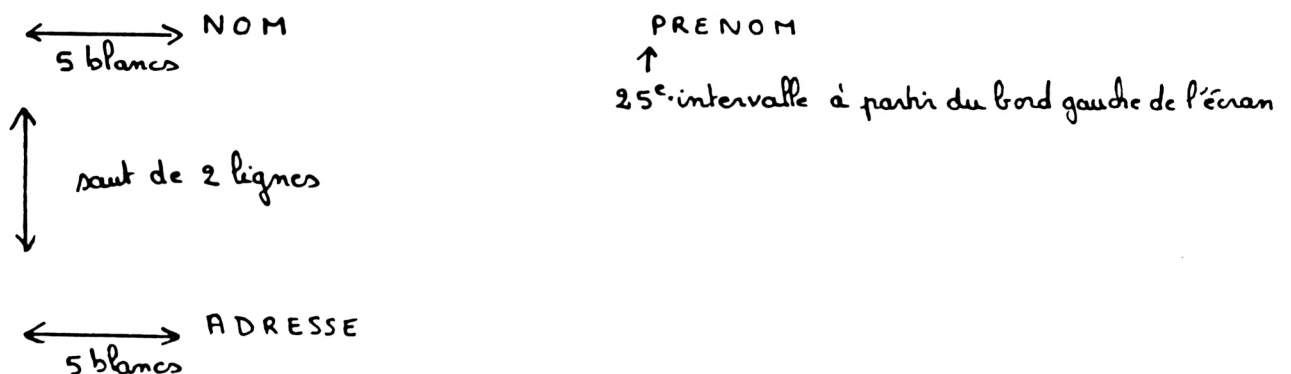
a) Votre programme se déroule tout à fait normalement.

C'est très bien : vous pouvez étudier le paragraphe suivant.

b) Votre programme comporte une ou plusieurs erreurs. Reprenez les paragraphes précédents, n'oubliez pas que vous avez des programmes analogues à celui-ci tout au long du chapitre.

2) Programme 2:

Voici un programme permettant d'introduire de façon explicite le nom, prénom, adresse d'un individu et de les écrire sur le modèle suivant:



• Instructions du programme:

```

NEW          ← ]
5 HOME      ← ]
10 INPUT " DONNEZ VOTRE NOM: "; A$ ← ]
15 INPUT " DONNEZ VOTRE PRENOM: "; B$ ← ]
20 INPUT " DONNEZ VOTRE ADRESSE: "; C$ ← ]
30 HOME ← ]
35 ? TAB(6) A$ ; TAB(25) B$ ← ]
40 ? ← ]
45 ? ← ]
50 ? TAB(6) C$ ← ]

```

• Tapez le programme précédent puis demandez à l'ordinateur d'exécuter les instructions de ce programme.

• Que constatez-vous ?

a) d'écran s'efface.

b) d'ordinateur imprime

DONNEZ VOTRE NOM :

Puis l'ordinateur attend une donnée alphanumérique, donnez votre réponse.

Si par exemple vous tapez DUPONT puis ←] : l'ordinateur passe immédiatement à l'instruction suivante après avoir mémorisé DUPONT en A\$.

c) Il imprime:

DONNEZ VOTRE PRENOM :

Puis il attend une nouvelle donnée alphanumérique. Donnez votre réponse.

Si par exemple vous tapez MICHEL puis ←] : l'ordinateur passe immédiatement à l'instruction suivante après avoir mémorisé MICHEL

dans la mémoire B\$.

d) l'imprime :

DONNEZ VOTRE ADRESSE :

Puis il attend une dernière donnée alphanumérique. Donnez votre réponse.

Si par exemple vous tapez 5 RUE ARISTIDE BRIAND puis ← :

l'ordinateur passe immédiatement à l'instruction suivante après avoir mémorisé 5 RUE ARISTIDE BRIAND dans la mémoire C\$.

e) l'écran s'efface (pas les contenus des mémoires)

f) l'écran apparaît de la façon suivante :

```

. . . . . DUPONT . . . . . MICHEL
. . . . .
. . . . .
. . . . . 5 RUE ARISTIDE BRIAND

```

Les pointillés ci-dessus permettent de symboliser les espaces vides ainsi que les sauts de ligne.

. Demandez à l'ordinateur d'exécuter de nouveau ce programme.

3) Programme 3 :

Déterminez un programme permettant d'introduire de façon explicite les renseignements suivants : nom, prénom, date de naissance, adresse complète avec ville et code postal, profession d'un individu et de les écrire selon le modèle suivant :

←→ NOM
5 blancs

↑
saut d'une ligne
↓

PRENOM
↑
25^e intervalle à partir du bord gauche de l'écran

←→ DATE DE NAISSANCE
5 blancs

↑
saut de deux lignes
↓

←→ ADRESSE
5 blancs

←→ VILLE
5 blancs

←→ code postal
5 blancs

↑
saut de deux lignes
↓

←→ PROFESSION
5 blancs

Un conseil pratique: vous pouvez vous aider du programme précédent.

Vous pouvez désormais essayer de trouver d'autres idées de programmes.

N'hésitez pas à les réaliser.

Ce chapitre ne nécessite l'utilisation d'aucun support magnétique. Il permet de se familiariser avec certaines touches ou instructions encore inconnues.

At. Utilisation des touches $>$; $<$; $>=$;
 $<=$; $< >$; $=$

I Quelques précisions indispensables :

<u>Symbole</u>	<u>Signification</u>
$>$	strictement supérieur à (on n'accepte pas l'égalité).
$<$	strictement inférieur à (on n'accepte pas l'égalité).
$>=$	supérieur <u>ou</u> égal à (ce symbole est noté \geq en mathématiques).
$<=$	inférieur <u>ou</u> égal à (ce symbole est noté \leq en mathématiques).
$< >$	n'est pas égal à.
$=$	est égal à

98.
II Vrai ou Faux ?

1) Exercice 1:

- Dans les deux cadres ci-dessous vous devez écrire VRAIE si la réponse vous paraît exacte, FAUSSE si la réponse vous paraît inexacte.

d'écriture $6 > 2$ est

d'écriture $55 > 78$ est

• Tapez $\begin{array}{c} ? \\ _ _ _ _ \end{array} 6 > 2 \leftarrow$

Sur l'écran apparaît:

• Tapez $\begin{array}{c} ? \\ _ _ _ _ \end{array} 55 > 78 \leftarrow$

Sur l'écran apparaît:

• Conclusion:

Pour l'ordinateur le chiffre 1 correspond à la réponse VRAIE

Pour l'ordinateur le chiffre 0 correspond à la réponse FAUSSE

2) Exercice 2:

• Tapez $\begin{array}{c} _ _ _ _ \\ _ _ _ _ \end{array} \text{JEAN} = 5 \leftarrow$

le chiffre 5 est mémorisé dans la mémoire JEAN.

• Tapez $\begin{array}{c} _ _ _ _ \\ _ _ _ _ \end{array} \text{JULES} = 9 \leftarrow$

le chiffre 9 est mémorisé dans la mémoire JULES.

• Tapez $\begin{array}{c} ? \\ _ _ _ _ \end{array} \text{JULES} < \text{JEAN} \leftarrow$

• Sur l'écran apparaît:

En effet la valeur mémorisée en mémoire JULES n'est pas strictement inférieure au nombre mémorisé en mémoire JEAN

• Tapez ? J E A N < J U L E S ←]

• Sur l'écran apparaît:

En effet il est exact d'écrire $5 < 9$.

3) Exercice 3:

• Tapez ? 4 > 2 ←]

Sur l'écran apparaît:

• Tapez ? 3 + (4 > 2) ←]

Sur l'écran apparaît:

En effet l'ordinateur effectue d'abord le calcul dans les parenthèses.

Comme $4 > 2$ est une écriture vraie le résultat du "calcul" entre parenthèses est égal à 1. De plus:

$$3 + 1 = 4$$

• Tapez ? 5 < > 4 ←]

Sur l'écran apparaît:

• Tapez ? (5 < > 4) * 2 ←]

Sur l'écran apparaît:

5 est différent de 4. Donc la valeur 1 est attribuée à l'expression entre parenthèses. De plus:

$$1 \times 2 = 2.$$

D'où le résultat imprimé sur l'écran.

4) Exercice 4:

• Tapez $T = 4 < > 45.7 \leftarrow$

4 est différent du nombre décimal 45,7. La valeur 1 est donc attribuée à l'expression: $4 < > 45.7$

d'ordinateur mémorise ce chiffre 1 dans la mémoire T.

• Tapez $? T \leftarrow$

Sur l'écran apparaît

5) Exercice 5:

• Tapez $R = 1 - (4 > = 5) \leftarrow$

4 n'est pas supérieur ou égal à 5. La valeur \emptyset est donc attribuée à l'expression entre parenthèses. De plus

$$1 - \emptyset = 1$$

d'ordinateur mémorise le chiffre 1 dans la mémoire R.

• Tapez $? R \leftarrow$

Sur l'écran apparaît:

• Tapez $V = 7.5 < (15/2) \leftarrow$

d'ordinateur divise 15 par 2. Il trouve 7,5.

7,5 n'est pas strictement inférieur à 7,5. Donc la valeur \emptyset est mémorisée dans la mémoire V.

• Zapez ? R = V ←]

Sur l'écran apparaît:

la valeur mémorisée en mémoire R n'est pas égale à la valeur mémorisée en mémoire V.

• Zapez ? R = T ←]

Sur l'écran apparaît:

la valeur mémorisée en mémoire R est égale à la valeur mémorisée en mémoire T.

B Utilisation des mots NOT ; OR ; AND dans les instructions.

I NOT:

1) Exercice 1:

• Zapez ? NOT 1 ←]

Sur l'écran apparaît:

En effet une réponse qui n'est pas vraie est fausse
 \downarrow \downarrow \downarrow \downarrow
 NOT 1 = \emptyset

• Tapez ? NOT Ø ←

Sur l'écran apparaît:

En effet une réponse qui n'est pas fausse est vraie
 NOT Ø = 1

2) Exercice 2:

• Précision importante:

Pour l'ordinateur tous les nombres différents de zéro correspondent à une valeur VRAIE. Seul le chiffre zéro correspond à la valeur FAUSSE.

• Tapez ? NOT 75 ←

Sur l'écran apparaît:

En tenant compte de l'indication ci-dessus : 75 est différent de zéro. 75 est donc considéré comme une valeur VRAIE.

<u>Instruction</u>	? NOT 75
<u>Traduction de l'instruction</u>	imprimez n'est pas vrai
<u>Réponse donnée par l'ordinateur</u>	Ø (FAUX)

3) Exercice 3:

• Tapez ? NOT (45 > 75) ←

• Sur l'écran apparaît:

la valeur \emptyset est attribuée à l'expression $45 > 75$ (car 45 n'est pas strictement supérieur à 75).

<u>Instruction</u>	?	NOT	(45 > 75)
<u>Traduction de l'instruction</u>	↓ imprimez	↓ n'est pas	↓ faux
<u>Réponse donnée par l'ordinateur</u>	1	(VRAI)	

II AND (et):

• Tapez ? 1 AND 1 ←]

Sur l'écran apparaît:

• Tapez ? ∅ AND 1 ←]

Sur l'écran apparaît:

• Tapez ? 1 AND ∅ ←]

Sur l'écran apparaît:

• Tapez ? ∅ AND ∅ ←]

Sur l'écran apparaît:

Conclusion:

d'ordinateur considère la "réponse" comme VRAIE dans le seul cas où les deux expressions séparées par AND sont raies simultanément.

III OR (ou):

• Tapez ? 1 OR 1 ←]

Sur l'écran apparaît

• Tapez ? 0 OR 1 ←]

Sur l'écran apparaît

• Tapez ? 1 OR 0 ←]

Sur l'écran apparaît

• Tapez ? 0 OR 0 ←]

Sur l'écran apparaît:

Conclusion:

d'ordinateur considère la "réponse" comme FAUSSE dans le seul cas où les deux expressions séparées par OR sont fausses simultanément.

C. Utilisation de $>$; $<$; $>=$; $<=$; $=$
 $<$ $>$; NOT ; AND ; OR.

Complétez le tableau suivant :

<u>Instructions</u>	Répondez dans cette colonne par <u>VRAI</u> ou <u>FAUX</u>	Réponse donnée par l'ordinateur
? $3 > 2$ \leftarrow]		
? $(3 > 2) \text{ AND } \emptyset$ \leftarrow]		
? NOT \emptyset \leftarrow]		
? $4 = 5$ \leftarrow]		
? $(\text{NOT } \emptyset) \text{ AND } (4 = 5)$ \leftarrow]		
? $4 < > 5$ \leftarrow]		
? $4 = 5$ \leftarrow]		
? $(4 < > 5) \text{ OR } (4 = 5)$ \leftarrow]		
? $\emptyset \text{ AND } 1$ \leftarrow]		
? $1 \text{ OR } (\emptyset \text{ AND } 1)$ \leftarrow]		
? $3 > 4$ \leftarrow]		
? $54 < 33$ \leftarrow]		
? NOT \emptyset \leftarrow]		
? $(3 > 4) \text{ OR } (54 < 33)$ \leftarrow]		
? $(3 > 4) \text{ OR } (54 < 33) \text{ AND NOT } \emptyset$ \leftarrow]		
? $23 \text{ OR } \emptyset$ \leftarrow]		
? NOT $(23 \text{ OR } \emptyset)$ \leftarrow]		

D. Priorités des "opérations"

- ① () parenthèses
- ② NOT ; attribution du signe - avec nombres négatifs
- ③ \wedge puissances (de gauche à droite)
- ④ * ; / multiplications et divisions de gauche à droite.
- ⑤ + ; - additions et soustractions de gauche à droite.
- ⑥ > ; < ; > = ; < = ; < > ; =
- ⑦ AND
- ⑧ OR

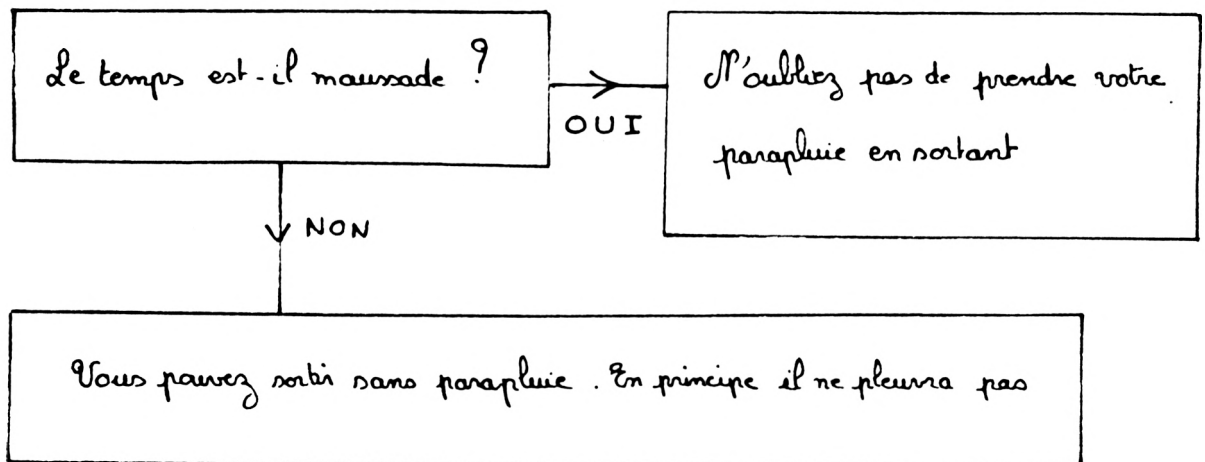
Instructions IF... THEN ;
GOTO ; IF... THEN GOTO ; END

Chacune de ces instructions n'est utilisable que dans les programmes.
(Il en était de même de l'instruction INPUT).

I Instruction IF.... THEN (Si..... Alors):

1) Problème:

Si le temps est maussade alors je prends mon parapluie.



2) Programme: lisez attentivement ce qui suit:

```

NEW ← ]
5 HOME ← ]
10 INPUT " LE TEMPS EST-IL MAUSSADE ? . REPONDEZ
    PAR OUI OU NON : " ; A$ ← ]
15 ? ← ]
20 IF A$ = "OUI" OR A$ = "oui" THEN PRINT "N'OUBLIEZ
    PAS DE PRENDRE VOTRE PARAPLUIE EN SORTANT" ← ]
  
```

```

25 IF A$ = "NON" OR A$ = "non" THEN PRINT "VOUS
    POUVEZ SORTIR SANS PARAPLUIE. EN PRINCIPE IL NE
    PLEUVRA PAS." ← ]

```

3) . Tapez le programme précédent

. Demandez à l'ordinateur d'exécuter les instructions de ce programme. Pour cela

tapez :

Sur l'écran apparaît :

d'ordinateur s'arrête, il attend une réponse. Vous devez taper OUI ou oui ou NON ou non.

Tapez par exemple OUI (ou bien oui). Dès que vous appuyez sur la touche ←] cette réponse est mémorisée dans la mémoire A\$.

Sur l'écran apparaît

d'exécution du programme est terminée.

. Demandez à l'ordinateur d'exécuter de nouveau ce programme. Pour cela

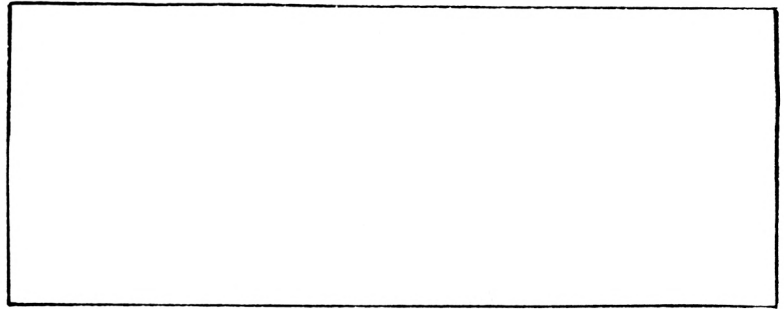
tapez :

Sur l'écran apparaît :

L'ordinateur s'arrête, il attend une donnée.

Tapez NON (ou non). Dès que vous appuyez sur la touche ← cette réponse est mémorisée dans la mémoire A\$.

Sur l'écran apparaît :



l'exécution du programme est terminée.

4) Etude du programme précédent.

Reprenons le paragraphe 2) et suivons "pas à pas" le travail effectué par l'ordinateur.

a) NEW ←]

nous savons maintenant que cette instruction permet d'effacer tout ancien programme de la mémoire centrale de l'ordinateur.

b) En tapant un numéro de ligne devant chacune des autres instructions nous en avons retardé l'exécution.

c) En tapant

RUN ←]

l'ordinateur exécute les instructions du programme dans l'ordre croissant des numéros. Il revient donc à l'instruction 5.

5 HOME

l'écran s'efface puis l'ordinateur passe à l'instruction suivante 10.

10 INPUT "LE TEMPS EST-IL MAUSSADE?". REPONDEZ
PAR OUI OU NON : " ; A\$

L'ordinateur imprime la phrase se trouvant entre guillemets. Puis il s'arrête, il attend une information. Votre réponse est limitée : vous avez le choix entre OUI ou NON.

Vous devez taper votre réponse puis appuyer sur la touche ←.

Votre réponse est alors mémorisée dans la mémoire A\$.

L'ordinateur passe à l'instruction suivante 15.

15 ?

L'ordinateur saute une ligne puis il passe à l'instruction suivante 20.

20 IF A\$ = "OUI" OR A\$ = "oui" THEN PRINT
" N'OUBLIEZ PAS DE PRENDRE VOTRE PARAPLUIE
EN SORTANT "

Deux cas peuvent se produire :

• 1^e cas :

L'information mémorisée dans la mémoire A\$ est OUI ou oui.

Dans ce cas l'ordinateur imprime la phrase se trouvant entre guillemets :

N'OUBLIEZ PAS DE PRENDRE VOTRE PARAPLUIE EN
SORTANT

Puis l'ordinateur passe à l'instruction suivante 25.

• 2^e cas :

L'information mémorisée dans la mémoire A\$ n'est ni OUI ni oui.

Dans ce cas l'ordinateur passe directement à l'instruction 25.

25 IF A\$ = "NON" OR A\$ = "non" THEN PRINT "VOUS

POUVEZ SORTIR SANS PARAPLUIE . EN PRINCIPE IL NE

PLEUVRA PAS "

Deux cas peuvent se produire :

• 1^{er} cas:

L'information mémorisée dans la mémoire A § est NON ou non . Dans ce cas l'ordinateur imprime la phrase se trouvant entre guillemets :

VOUS POUVEZ SORTIR SANS PARAPLUIE . EN PRINCIPE
IL NE PLEUVRA PAS

Puis l'ordinateur s'arrête car il n'y a plus d'instruction dans le programme .

• 2^{er} cas:

L'information mémorisée dans la mémoire A § n'est ni NON ni non .

Dans ce cas l'ordinateur passe directement à l'instruction suivante .

Comme il n'y en a pas , il s'arrête .

Dans ce programme deux cas seulement sont envisagés :

- le cas où la réponse est OUI ou oui .
- le cas où la réponse est NON ou non .

II Instruction IF ... THEN GOTO (Si ... Alors aller) ;

Instruction END (fin) ; Instruction GOTO (aller) :

L'instruction IF ... THEN GOTO est la forme la plus utilisée de l'instruction si ... alors .

Reprenons le problème du paragraphe I 1).

Voici une variante du programme précédent :

112-

1) Programme : lisez attentivement ce qui suit :

```
NEW ← ]  
5 HOME ← ]  
10 INPUT " LE TEMPS EST -IL MAUSSADE ? . REPONDEZ  
PAR OUI OU NON : " ; A$ ← ]  
15 ? ← ]  
20 IF A$ = "OUI" OR A$ = "oui" THEN GOTO 55  
← ]  
25 IF A$ = "NON" OR A$ = "non" THEN GOTO 45  
← ]  
30 ? " LA REPONSE DONNEE A LA QUESTION  
PRECEDENTE N'EST PAS CORRECTE . VOUS POUVEZ  
UNIQUEMENT REPONDRE PAR OUI OU NON " ← ]  
35 ? ← ]  
40 GOTO 10 ← ]  
45 ? " VOUS POUVEZ SORTIR SANS PARAPLUIE . EN  
PRINCIPE IL NE PLEUVRA PAS " ← ]  
50 END ← ]  
55 ? " N'OUBLIEZ PAS DE PRENDRE VOTRE PARAPLUIE  
EN SORTANT " ← ]
```

2) . Tapez le programme précédent :

vous avez deux possibilités à votre disposition :

1^e possibilité : taper entièrement les instructions ci-dessus .

2^e possibilité : taper les instructions à partir de l'instruction 20 .

En effet les "lignes" 5 ; 10 et 15 sont identiques à celles du précédent programme . De plus dans ce cas l'instruction NEW est inutile

puisque les instructions 20 et 25 du programme du paragraphe I 2) sont remplacées par de nouvelles instructions, elles sont donc automatiquement effacées de la mémoire centrale de l'ordinateur.

- Demandez à l'ordinateur d'exécuter les instructions de ce programme.

Pour cela tapez

Sur l'écran apparaît :

l'ordinateur s'arrête, il attend une réponse. Vous devez taper OUI ou OUI ou NON ou non.

Tapez par exemple OUI (ou bien oui). Dès que vous appuyez sur la touche ← cette réponse est mémorisée dans la mémoire A5.

Sur l'écran apparaît :

l'exécution du programme est terminée.

- Demandez à l'ordinateur d'exécuter de nouveau ce programme. Pour cela

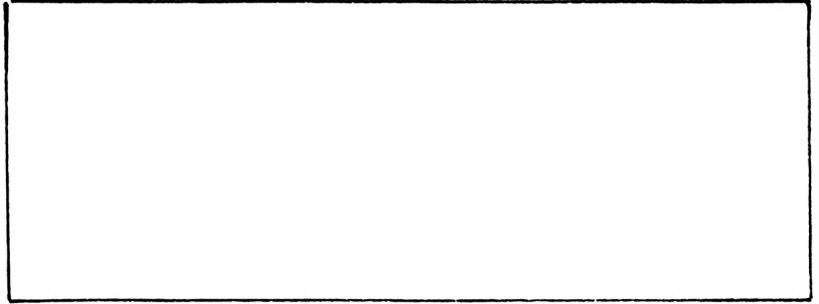
tapez :

Sur l'écran apparaît :

d'ordinateur s'arrête , il attend une donnée .

Tapez NON (ou non) . Dès que vous appuyez sur la touche ←]
cette réponse est mémorisée dans la mémoire A\$.

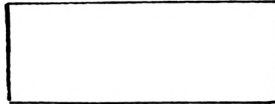
Sur l'écran apparaît :



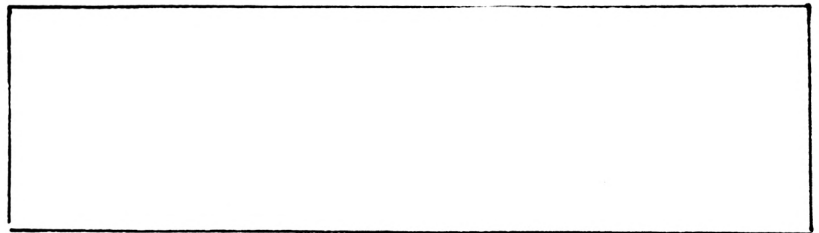
d'exécution du programme est terminée .

• Demandez à l'ordinateur d'exécuter une dernière fois ce programme .

Pour cela tapez



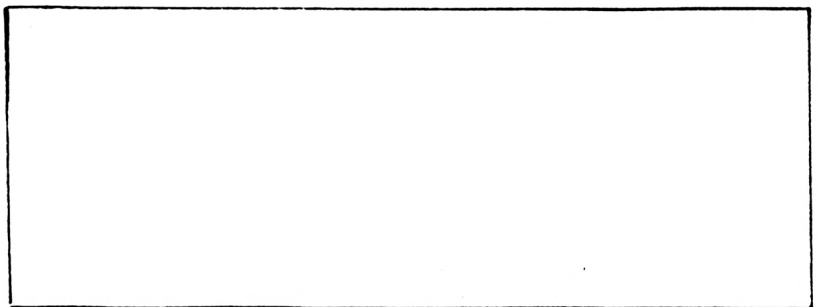
Sur l'écran apparaît



d'ordinateur s'arrête , il attend une donnée .

Tapez OUI (l'erreur portant sur la première lettre du mot OUI
est commise volontairement) . Dès que vous appuyez sur la touche ←]
cette réponse est mémorisée dans la mémoire A\$.

Sur l'écran apparaît :

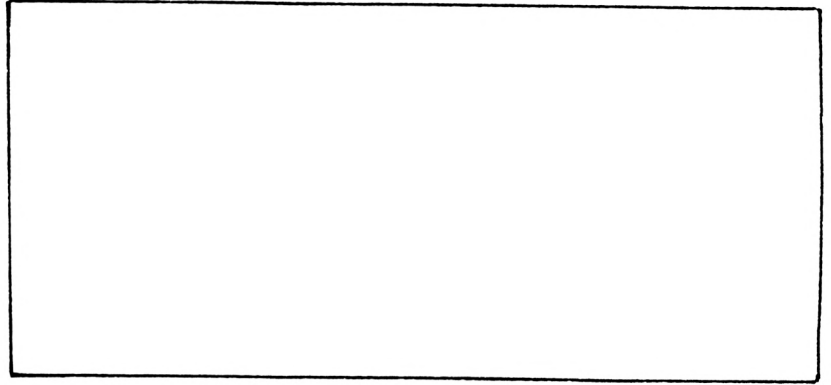


l'ordinateur s'arrête de nouveau , il attend une donnée .

Tapez l'une des quatre réponses OUI ; oui ; NON ; non (A vous
de choisir votre réponse) . Dès que vous appuyez sur la touche ←]

cette réponse est mémorisée dans la mémoire A §.

Sur l'écran apparaît:



d'exécution du programme est terminée.

3) Etude du programme précédent:

Reprenons le paragraphe II 1) et suivons "pas à pas" le travail effectué par l'ordinateur.

a) NEW ←]

nous savons maintenant que cette instruction permet d'effacer tout ancien programme de la mémoire centrale de l'ordinateur.

b) En tapant un numéro de ligne devant chacune des autres instructions nous en retardons l'exécution.

c) En tapant

RUN ←]

l'ordinateur exécute les instructions du programme dans l'ordre croissant des numéros. Il revient donc à l'instruction 5.

S HOME ←

l'écran s'efface puis l'ordinateur passe à l'instruction suivante 10

10 INPUT "LE TEMPS EST-IL MAUSSADE ?. REPONDEZ
PAR OUI OU NON: "; A\$

L'ordinateur imprime la phrase se trouvant entre guillemets. Puis il s'arrête, il attend une information. Votre réponse est limitée : vous avez le choix entre OUI ou NON.

Vous devez taper votre réponse puis appuyer sur la touche \leftarrow .

Votre réponse est alors mémorisée dans la mémoire A\$.

L'ordinateur passe à l'instruction suivante 15.

15 ?

L'ordinateur saute une ligne puis il passe à l'instruction suivante 20.

20 IF A\$ = "OUI" OR A\$ = "oui" THEN GOTO 55

Deux cas peuvent se produire :

• 1^e cas :

L'information mémorisée dans la mémoire A\$ est OUI ou oui.

Dans ce cas l'ordinateur passe à l'instruction 55 (il saute les instructions 25; 30; 35; 40; 45; 50 comme si elles n'existaient pas).

55 ? "N'OUBLIEZ PAS DE PRENDRE VOTRE PARAPLUIE

EN SORTANT"

L'ordinateur imprime alors sur l'écran :

N'OUBLIEZ PAS DE PRENDRE VOTRE PARAPLUIE EN
SORTANT

L'ordinateur s'arrête, l'exécution du programme est terminée.

• 2^e cas :

L'information mémorisée dans la mémoire A\$ n'est ni OUI ni oui

Dans ce cas l'ordinateur passe directement à l'instruction suivante:
l'instruction 25.

25 IF A\$ = "NON" OR A\$ = "non" THEN GOTO 45

Deux cas peuvent se produire :

• 1^{er} cas :

d'information mémorisée dans la mémoire A\$ est NON ou non.

Dans ce cas l'ordinateur passe à l'instruction 45 (il saute les instructions 30, 40 et 50 comme si elles n'existaient pas).

45 ? " VOUS POUVEZ SORTIR SANS PARAPLUIE . EN
PRINCIPE IL NE PLEUVRA PAS "

l'ordinateur imprime alors sur l'écran :

VOUS POUVEZ SORTIR SANS PARAPLUIE . EN PRINCIPE
IL NE PLEUVRA PAS

l'ordinateur passe à l'instruction suivante : l'instruction 50

50 END

l'ordinateur s'arrête , l'exécution du programme est terminée .

• 2^{er} cas :

d'information mémorisée dans la mémoire A\$ n'est ni NON ni non.

Dans ce cas l'ordinateur passe directement à l'instruction suivante:
l'instruction 30.

30 ? " LA REPONSE DONNEE A LA QUESTION PRECEDENTE
N'EST PAS CORRECTE . VOUS POUVEZ UNIQUEMENT
REPOUDRE PAR OUI OU NON "

l'ordinateur imprime alors sur l'écran :

LA REPONSE DONNEE A LA QUESTION PRECEDENTE N'EST PAS CORRECTE . VOUS POUVEZ UNIQUEMENT REPONDRE PAR OUI OU NON

d'ordinateur passe à l'instruction suivante : l'instruction 35

35 ?

d'ordinateur saute une ligne puis il passe à l'instruction suivante 40

40 GOTO 10

d'ordinateur revient en arrière , il retourne à l'instruction 10

le travail effectué par l'ordinateur est alors identique à celui que nous venons d'expliquer . Si vous éprouvez encore quelques difficultés reprenez les explications à partir de l'instruction 10.

III Exemples de programmes utilisant les instructions précédentes :

1) Exemple 1:

a) Problème:

Un éditeur doit effectuer un programme permettant d'ordonner deux nombres décimaux par ordre croissant.

b) Programme: lisez attentivement ce qui suit :

NEW ←]

5 HOME ←]

10 ? " CE PROGRAMME PERMET D'ORDONNER PAR ORDRE CROISSANT DEUX NOMBRES DECIMAUX " ←]


```

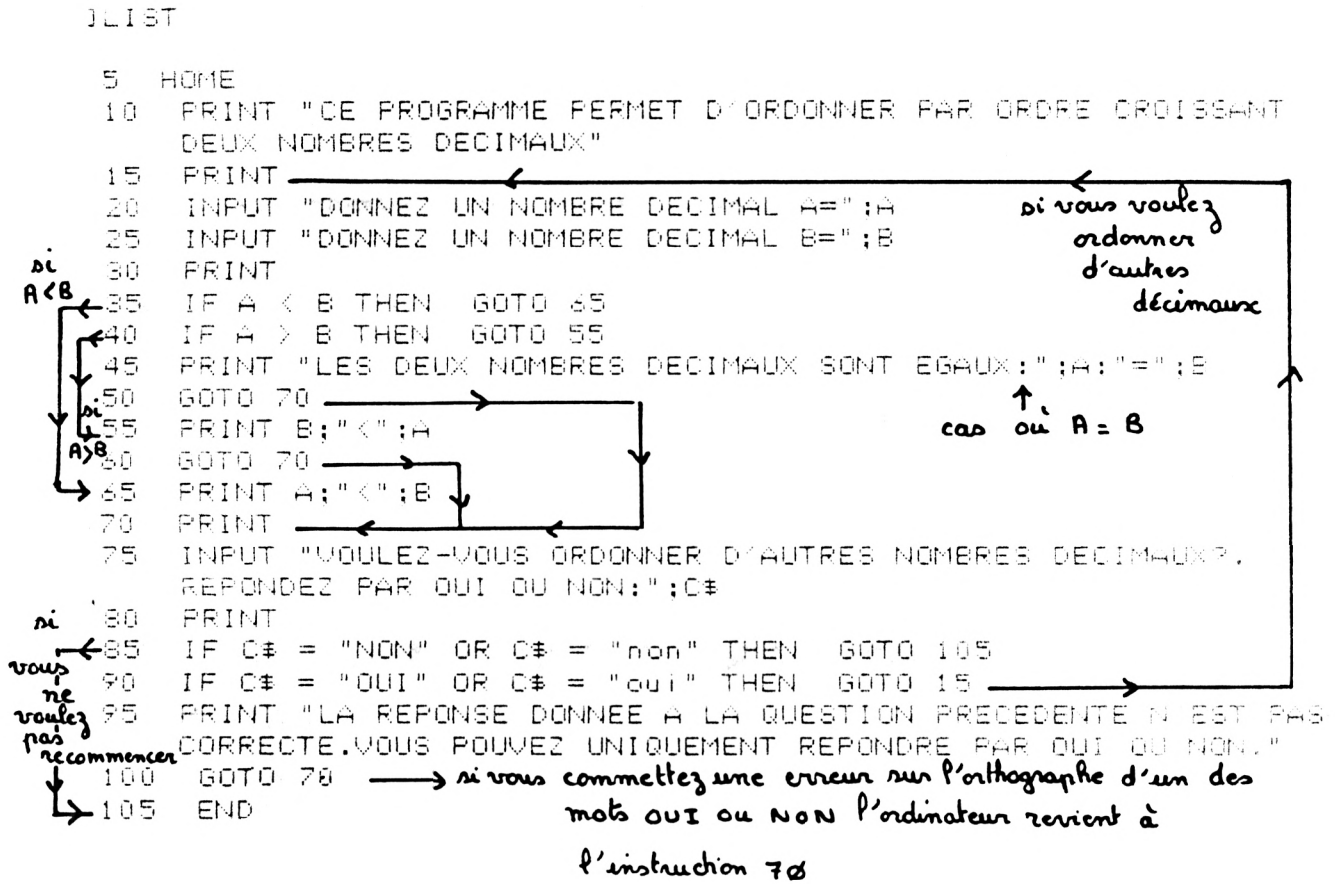
15 ? ← ]
20 INPUT " DONNEZ UN NOMBRE DECIMAL A = "; A ← ]
25 INPUT " DONNEZ UN NOMBRE DECIMAL B = "; B ← ]
30 ? ← ]
35 IF A < B THEN GOTO 65 ← ]
40 IF A > B THEN GOTO 55 ← ]
45 ? " LES DEUX NOMBRES DECIMAUX SONT EGaux : "; A ;
    " = "; B ← ]
50 GOTO 70 ← ]
55 ? B ; " < " ; A ← ]
60 GOTO 70 ← ]
65 ? A ; " < " ; B ← ]
70 ? ← ]
75 INPUT " VOULEZ - VOUS ORDONNER D'AUTRES NOMBRES
    DECIMAUX ? . REPONDEZ PAR OUI OU NON : "; C$ ← ]
80 ? ← ]
85 IF C$ = "NON" OR C$ = "non" THEN GOTO 105 ← ]
90 IF C$ = "OUI" OR C$ = "oui" THEN GOTO 15 ← ]
95 ? " LA REPONSE DONNEE A LA QUESTION PRECEDENTE
    N'EST PAS CORRECTE . VOUS POUVEZ UNIQUEMENT
    REPONDRE PAR OUI OU NON." ← ]
100 GOTO 70 ← ]
105 END ← ]

```

c) Copiez le programme ci-dessus puis demandez à l'ordinateur d'exécuter les instructions de ce programme .

d) Suivez pas à pas le travail effectué par l'ordinateur.

Pour vous aider voici le listing (liste complète des instructions) du programme accompagné de quelques explications indispensables.



Voici un exemple d'exécution de ce programme obtenu sur une imprimante.

Cet exemple envisage chaque réponse possible :

- * le cas où $A < B$.
- * le cas où vous souhaitez ordonner de nouveau d'autres décimaux.
- * le cas où $A > B$.
- * le cas où $A = B$.
- * le cas où vous ne souhaitez plus ordonner d'autres décimaux mais où vous commettez une erreur sur l'orthographe de la réponse NON.
- * le cas où vous signalez convenablement à l'ordinateur que vous n'avez plus de décimaux à ordonner.

FIN

CE PROGRAMME PERMET D'ORDONNER PAR ORDRE CROISSANT DEUX NOMBRES DECIMAUX

DONNEZ UN NOMBRE DECIMAL A=33.4
 DONNEZ UN NOMBRE DECIMAL B=33.39

$33.39 < 33.4$

VOULEZ-VOUS ORDONNER D'AUTRES NOMBRES DECIMAUX?.REPONDEZ PAR OUI OU NON:OUI

DONNEZ UN NOMBRE DECIMAL A=33.39
 DONNEZ UN NOMBRE DECIMAL B=33.4

$33.39 < 33.4$

VOULEZ-VOUS ORDONNER D'AUTRES NOMBRES DECIMAUX?.REPONDEZ PAR OUI OU NON:OUI

DONNEZ UN NOMBRE DECIMAL A=34
 DONNEZ UN NOMBRE DECIMAL B=34.0

LES DEUX NOMBRES DECIMAUX SONT EGaux: $34 = 34$

VOULEZ-VOUS ORDONNER D'AUTRES NOMBRES DECIMAUX?.REPONDEZ PAR OUI OU NON:NON

LA REPONSE DONNEE A LA QUESTION PRECEDENTE N'EST PAS CORRECTE.
 VOUS POUVEZ UNIQUEMENT REPONDRE PAR OUI OU NON.

VOULEZ-VOUS ORDONNER D'AUTRES NOMBRES DECIMAUX?.REPONDEZ PAR OUI OU NON:NON

2) Exercice:

Déterminez un programme permettant d'ordonner deux nombres décimaux par ordre décroissant.

Vous pouvez vous aider du programme précédent.

3) Exemple 2:

a) Problème:

Un élève doit citer tous les nombres de 3 en 3 à partir de 12 jusqu'à 99. Enfin tous les nombres décimaux de 3 en 3 à partir de 11 jusqu'à 95, et tous les décimaux de 3 en 3 à partir de 19 jusqu'à 97.

b) Programme: lisez attentivement ce qui suit :

```

NEW ← ]
5 HOME ← ]
10 INPUT "QUEL EST LE PREMIER DECIMAL A CITER?";
    B ← ]
15 INPUT "QUEL EST LE DERNIER DECIMAL A CITER
    EVENTUELLEMENT?"; C ← ]
20 ? ← ]
30 ? B, ← ]
35 B = B + 3 ← ]
40 IF B < = C THEN GOTO 30 ← ]
45 ? ← ]
50 INPUT "DEVEZ-VOUS RECOMMENCER? . REPONDEZ
    PAR OUI OU NON : "; D$ ← ]
55 IF D$ = "OUI" OR D$ = "oui" THEN GOTO 10 ← ]
60 IF D$ = "NON" OR D$ = "non" THEN GOTO 75 ← ]
65 ? "LA REPONSE A LA QUESTION PRECEDENTE N'EST
    PAS CORRECTE . VOUS DEVEZ REPONDRE PAR OUI OU
    NON." ← ]
70 GOTO 45 ← ]
75 END ← ]

```

c) Tapez le programme ci-dessus puis demandez à l'ordinateur d'exécuter les instructions de ce programme.

d) Suivez pas à pas le travail effectué par l'ordinateur.

Pour vous aider voici le listing de ce programme accompagné de quelques

DLIST

```

5 HOME
10 INPUT "QUEL EST LE PREMIER DECIMAL A CITER?":B
15 INPUT "QUEL EST LE DERNIER DECIMAL A CITER EVENTUELLEMENT?":C
20 PRINT
30 PRINT B,
35 B = B + 3
40 IF B < = C THEN GOTO 30
45 PRINT
50 INPUT "DEVEZ-VOUS RECOMMENCER?.REPONDEZ PAR OUI OU NON:":D#
55 IF D# = "OUI" OR D# = "oui" THEN GOTO 10
60 IF D# = "NON" OR D# = "non" THEN GOTO 75
65 PRINT "LA REPONSE A LA QUESTION PRECEDENTE N'EST PAS CORRECTE.
VOUS DEVEZ REPONDRE PAR OUI OU NON."
70 GOTO 45
75 END
    
```

← la virgule permet un affichage des résultats par colonne

Avant de donner un exemple d'exécution de ce programme obtenu sur une imprimante, quelques remarques s'imposent :

- Ce programme permet à l'élève d'obtenir une réponse à chacune des questions posées.
- Un affichage par colonne est très pratique, il permet à l'élève de pouvoir copier convenablement les nombres donnés par l'ordinateur.
- Vous pouvez essayer de taper le même programme en enlevant la virgule à la fin de l'instruction 30. L'élève n'a alors plus le temps de copier les réponses affichées sur l'écran (sauf s'il dispose d'une imprimante).

Voici un exemple d'exécution de ce programme obtenu sur une imprimante :

IRUN

QUEL EST LE PREMIER DECIMAL A CITER?12
 QUEL EST LE DERNIER DECIMAL A CITER EVENTUELLEMENT?99

12	15	18	21	24
27	30	33	36	39
42	45	48	51	54
57	60	63	66	69
72	75	78	81	84
87	90	93	96	99

DEVEZ-VOUS RECOMMENCER?.REPONDEZ PAR OUI OU NON:OUI
 QUEL EST LE PREMIER DECIMAL A CITER?11
 QUEL EST LE DERNIER DECIMAL A CITER EVENTUELLEMENT?95

11	14	17	20	23
26	29	32	35	38
41	44	47	50	53
56	59	62	65	68
71	74	77	80	83
86	89	92	95	

DEVEZ-VOUS RECOMMENCER?.REPONDEZ PAR OUI OU NON:pui
 LA REPONSE A LA QUESTION PRECEDENTE N'EST PAS CORRECTE.VOUS DEVEZ REPONDRE
 PAR OUI OU NON

DEVEZ-VOUS RECOMMENCER?.REPONDEZ PAR OUI OU NON:oui
 QUEL EST LE PREMIER DECIMAL A CITER?19
 QUEL EST LE DERNIER DECIMAL A CITER EVENTUELLEMENT?97

19	22	25	28	31
34	37	40	43	46
49	52	55	58	61
64	67	70	73	76
79	82	85	88	91
94	97			

DEVEZ-VOUS RECOMMENCER?.REPONDEZ PAR OUI OU NON:NON

4) Exercice:

Un élève doit citer tous les nombres décimaux de 7 en 7 à partir de 12,7 jusqu'à 99,2 . Puis tous les nombres décimaux de 7 en 7 à partir de 11,7 jusqu'à 95,2 . Enfin tous les décimaux de 7 en 7 à partir de 19 jusqu'à 97.

Déterminez un programme permettant de résoudre ce problème . Vous

prenez vous aider du programme précédent.

5) Exercice:

Déterminez un programme permettant d'obtenir tous les entiers (naturels) multiples de 3 inférieurs à 125.

Indication: Déterminez tous les entiers multiples de 3 inférieurs à 125 signifie citez tous les nombres de 3 en 3 à partir de 0 jusqu'à 125.

6) Exercice:

Déterminez un programme permettant d'obtenir tous les entiers (naturels) multiples de 7 inférieurs à 99.

7) Exercice: généralisation des programmes précédents:

Soient A , B et C trois nombres décimaux ; B étant inférieur à C .

Déterminez un programme permettant de citer tous les décimaux de A en A à partir de B jusqu'à C .

Le programme s'adresse aux plus initiés.

Instruction FOR... TO...
NEXT...

Cette instruction n'est utilisable que dans les programmes.

I Instruction FOR... TO...

NEXT...

1) a) Problème 1:

Vous souhaitez effectuer un programme permettant d'obtenir sur l'écran un affichage par colonne de tous les entiers naturels inférieurs ou égaux à 55.

b) Programme: lisez attentivement ce qui suit:

```

NEW      ←
S HOME  ←
10 FOR I = 0 TO 55 ←
15 ? I, ←
20 NEXT I ←

```

c) Tapez le programme précédent

Demandez à l'ordinateur d'exécuter les instructions de ce programme, pour cela tapez:

Sur l'écran apparaît: l'affichage par colonne de tous les entiers naturels de 0 à 55. L'affichage par colonne provient de la virgule à la fin de l'instruction 15.

d) Etude du programme précédent:

Reprenons le paragraphe b) et suivons "pas à pas" le travail effectué par l'ordinateur.

• NEW ←]

nous savons maintenant que cette instruction permet d'effacer tout ancien programme de la mémoire centrale de l'ordinateur.

• En tapant un numéro de ligne devant chacune des autres instructions nous en avons retardé l'exécution.

• En tapant:

RUN ←]

L'ordinateur exécute les instructions du programme dans l'ordre croissant des numéros. Il revient donc à l'instruction 5.

5 HOME

l'écran s'efface puis l'ordinateur passe à l'instruction suivante 10

10 FOR I = 0 TO 55 " Pour I = 0 jusqu'à 55 "

l'ordinateur mémorise la valeur 0 dans la mémoire I puis il passe à l'instruction suivante 15

15 ? I ,

l'ordinateur imprime la valeur mémorisée dans la mémoire I puis il passe à l'instruction suivante: 20

20 NEXT I " PROCHAIN I " (prochaine valeur de I)

l'ordinateur revient en arrière, il retourne à l'instruction 10 (instruction FOR...TO), il mémorise la valeur 1 (valeur qui "suit" 0) dans

la mémoire I, puis il passe à l'instruction suivante 15.

15 ? I,

l'ordinateur imprime la valeur mémorisée dans la mémoire I (c'est à dire

1) puis il passe à l'instruction suivante 20.

20 NEXT I (prochaine valeur de I)

l'ordinateur revient à l'instruction 10 (instruction FOR... TO). Il

mémorise la valeur 2 (valeur qui "suit" 1) dans la mémoire I, puis il passe à l'instruction suivante 15.

15 ? I,

l'ordinateur imprime la valeur mémorisée dans la mémoire I (c'est à dire 2) puis il passe à l'instruction suivante 20.

20 NEXT I (prochaine valeur de I)

l'ordinateur revient à l'instruction 10

.....

.....

.....

20 NEXT I (prochaine valeur de I)

l'ordinateur revient à l'instruction 10 (instruction FOR... TO). Il

mémorise la valeur 55 (valeur qui "suit" 54) dans la mémoire I,

puis il passe à l'instruction suivante 15.

15 ? I,

l'ordinateur imprime la valeur mémorisée dans la mémoire I (c'est à dire 55) puis il passe à l'instruction suivante 20

20 NEXT I (prochaine valeur de I)

Il n'y a plus de prochaine valeur de I. En effet on devait mémoriser dans la mémoire I les entiers successifs de 0 à 55. Le travail est achevé, l'ordinateur passe à l'instruction suivante. Il n'y en a plus : l'exécution

du programme est achevée.

2) a) Problème 2:

Vous désirez effectuer un programme permettant d'obtenir sur l'écran l'affichage temporaire (pendant quelques secondes) de la phrase suivante :
BONJOUR , JE VAIS DISPARAITRE DE L'ECRAN.

b) Programme: lisez attentivement ce qui suit:

NEW ←]

S HOME ←]

10 ? " BONJOUR, JE VAIS DISPARAITRE DE L'ECRAN." ←]

15 FOR A = 1 TO 5000 ←]

20 NEXT A ←]

25 HOME ←]

c) Tapez le programme précédent

Demandez à l'ordinateur d'exécuter les instructions de ce programme, pour cela tapez

Sur l'écran apparaît:

La phrase précédente reste affichée 6 à 7 secondes sur l'écran avant de s'effacer.

d) Etude du programme précédent:

Reprenons les instructions du paragraphe 2) b) et suivons "pas à pas" le travail effectué par l'ordinateur.

• NEW ←]

Cette instruction permet d'effacer tout ancien programme de la mémoire centrale de l'ordinateur.

• En tapant un numéro de ligne devant chacune des autres instructions nous en avons retardé l'exécution.

• En tapant :

RUN ←]

l'ordinateur exécute les instructions du programme dans l'ordre croissant des numéros. Il revient donc à l'instruction 5.

5 HOME

l'écran s'efface puis l'ordinateur passe à l'instruction suivante 10.

10 ? " BONJOUR, JE VAIS DISPARAITRE DE L'ECRAN."

l'ordinateur imprime sur l'écran la phrase :

BONJOUR, JE VAIS DISPARAITRE DE L'ECRAN.

Puis l'ordinateur passe à l'instruction suivante 15.

15 FOR A = 1 TO 5000 " Pour A = 1 jusqu'à 5000 "

l'ordinateur mémorise la valeur 1 dans la mémoire A puis il passe à l'instruction suivante 20

20 NEXT A " prochain A " (Prochaine valeur de A)

l'ordinateur revient à l'instruction 15 (instruction FOR ... TO ...)

Il mémorise la valeur 2 dans la mémoire A (valeur "qui suit" 1) puis il passe à l'instruction suivante 20

...

...

20 NEXT A "prochain A" (prochaine valeur de A).

L'ordinateur revient à l'instruction 15 (instruction FOR... TO...).

Il mémorise la valeur 5000 (valeur qui suit 4999) dans la mémoire A.

Puis il passe à l'instruction suivante 20.

20 NEXT A "prochain A" (prochaine valeur de A).

Il n'y a plus de prochaine valeur de A. En effet on devrait mémoriser dans la mémoire A les entiers successifs de 1 à 5000. Ce travail est achevé, l'ordinateur passe à l'instruction suivante 25.

25 HOME

L'écran s'efface, il n'y a plus d'instruction après l'instruction 25.

L'exécution du programme est terminée.

e) Remarque:

Entre l'impression sur l'écran de la phrase:

BONJOUR, JE VAIS DISPARAITRE DE L'ECRAN.

et l'effacement de l'écran: l'ordinateur effectue une boucle (il compte de 1 à 5000).

Vous pouvez constater que l'ordinateur compte très rapidement puisqu'il parvient à compter de 1 à 5000 en un temps record de 6 à 7 secondes.

II Exemples de programmes utilisant l'instruction précédente:

1) Exercice 1:

Déterminez un programme permettant d'obtenir sur l'écran un affichage

par colonne de tous les entiers naturels inférieurs ou égaux à 100. Vous pouvez vous aider du programme du paragraphe précédent (I 1)

2) Inconvénient du programme précédent:

Le programme que vous venez d'effectuer présente un inconvénient. Malgré l'affichage par colonne vous n'avez plus le temps de copier les nombres donnés par l'ordinateur.

a) Problème:

Un édier doit effectuer un programme permettant d'obtenir tous les entiers de 1 à 100, l'affichage des résultats sur l'écran devant être suffisamment long pour permettre à l'élève de recopier les réponses proposées.

b) Programme: lisez attentivement ce qui suit :

```

NEW ← ]
S HOME ← ]
10 FOR I = 1 TO 54 ← ]
15 ? I, ← ]
20 NEXT I ← ]
25 FOR C = 1 TO 20000 ← ]
30 NEXT C ← ]
35 HOME ← ]
40 FOR I = 55 TO 100 ← ]
45 ? I, ← ]
50 NEXT I ← ]

```

c) Tapez le programme précédent puis demandez à l'ordinateur d'exécuter les instructions de ce programme.

d) quelques explications semblent nécessaires :

d'écran s'efface (instruction 5), puis les nombres entiers de 1 à 54 s'affichent par colonne (instructions 10 ; 15 ; 20).

L'ordinateur compte de 1 à 20000 (instructions 25 et 30). Ce compteur permet à l'élève de recopier convenablement les réponses proposées sur l'écran.

d'écran s'efface de nouveau (instruction 35).

Enfin les nombres entiers de 55 à 100 s'affichent.

Remarque : Il est possible de laisser plus ou moins de temps à l'élève pour recopier les valeurs proposées. Il suffit pour cela d'agir sur l'instruction 25.

3) Exercice 2 :

Effectuez un programme permettant d'obtenir sur l'écran un affichage par colonne de tous les entiers supérieurs ou égaux à 120 et inférieurs ou égaux à 155 (entiers de 120 à 155).

4) Exercice 4 :

Un écolier doit effectuer un programme permettant d'obtenir tous les entiers de 222 à 322, l'affichage des résultats sur l'écran devant être suffisamment long pour permettre à l'élève de recopier les réponses proposées.

Vous pouvez vous aider du programme du paragraphe II 2) b)

III "Construction" d'une horloge :

1) Programme : lisez attentivement ce qui suit :

```
NEW ← ]
5 HOME ← ]
10 ? " DONNEZ L'HEURE EXACTE EN HEURES , MINUTES ,
    SECONDES " ← ]
15 INPUT " DONNEZ LE NOMBRE DE Ø A 23 REPRESENTANT
    LES HEURES : " ; H ← ]
20 INPUT " DONNEZ LE NOMBRE DE Ø A 59 REPRESENTANT
    LES MINUTES : " ; M ← ]
25 INPUT " DONNEZ LE NOMBRE DE Ø A 59 REPRESENTANT
    LES SECONDES : " ; S ← ]
30 ? H ; " HEURES " ; M ; " MINUTES " ; S ; " SECONDES "
    ← ]
35 FOR I = 1 TO 66Ø ← ]
40 NEXT I ← ]
45 S = S + 1 ← ]
50 IF S < 6Ø THEN GOTO 9Ø ← ]
55 M = M + 1 ← ]
60 S = Ø ← ]
65 IF M < 6Ø THEN GOTO 9Ø ← ]
70 H = H + 1 ← ]
75 M = Ø ← ]
80 IF H < 24 THEN GOTO 9Ø ← ]
85 H = Ø ← ]
90 HOME ← ]
```

95 GOTO 3 Ø ←]

- 2) Tapez le programme précédent puis demandez à l'ordinateur d'exécuter les instructions de ce programme.

Attention:

l'horloge ne s'arrête pas toute seule, elle continue indéfiniment si vous ne donnez pas l'ordre à l'ordinateur d'interrompre le déroulement du programme.

Pour arrêter cette exécution vous pouvez appuyer simultanément sur les touches

CTRL et RESET

Vous pouvez également couper le courant d'alimentation si vous ne souhaitez plus travailler sur l'ordinateur.

- 3) Etude du programme précédent : fonctionnement de l'horloge:

Reprenons les instructions du paragraphe III 1) et suivons "pas à pas" le travail effectué par l'ordinateur.

a) NEW ←]

Cette instruction permet d'effacer tout ancien programme de la mémoire centrale de l'ordinateur.

b) En tapant un numéro de ligne devant chacune des autres instructions nous en avons retardé l'exécution.

c) En tapant :

RUN ←]

L'ordinateur exécute les instructions dans l'ordre croissant des numéros.

Il revient donc à l'instruction 5.

5 HOME

L'écran s'efface puis l'ordinateur passe à l'instruction suivante 10.

10 ? " DONNEZ L'HEURE EXACTE EN HEURES, MINUTES,
SECONDES "

L'ordinateur imprime la phrase se trouvant entre guillemets :

DONNEZ L'HEURE EXACTE EN HEURES, MINUTES, SECONDES
puis il passe à l'instruction suivante 15

15 INPUT " DONNEZ LE NOMBRE DE 0 A 23 REPRESENTANT
LES HEURES : " ; H

L'ordinateur imprime la phrase se trouvant entre guillemets :

DONNEZ LE NOMBRE DE 0 A 23 REPRESENTANT LES HEURES:
puis il s'arrête. Il attend une donnée numérique.

Vous devez taper votre réponse (un nombre de 0 à 23) puis appuyer sur la touche \leftarrow . Votre réponse est alors mémorisée dans la mémoire H.

L'ordinateur passe à l'instruction suivante 20

20 INPUT " DONNEZ LE NOMBRE DE 0 A 59 REPRESENTANT
LES MINUTES : " ; M

L'ordinateur imprime la phrase se trouvant entre guillemets :

DONNEZ LE NOMBRE DE 0 A 59 REPRESENTANT LES MINUTES:
puis il s'arrête. Il attend une donnée numérique.

Vous devez taper votre réponse (un nombre de 0 à 59) puis appuyer sur

la touche \leftarrow . Votre réponse est alors mémorisée dans la mémoire M. L'ordinateur passe à l'instruction suivante 25.

25 INPUT " DONNEZ LE NOMBRE DE Ø A 59 REPRESENTANT
LES SECONDES : "; S

L'ordinateur imprime la phrase se trouvant entre guillemets :

DONNEZ LE NOMBRE DE Ø A 59 REPRESENTANT LES
SECONDES :

puis il s'arrête, il attend une donnée numérique.

Vous devez taper votre réponse (un nombre de Ø à 59) puis appuyer sur la touche \leftarrow . Votre réponse est alors mémorisée dans la mémoire S.

L'ordinateur passe à l'instruction suivante 3Ø.

3Ø ? H ; " HEURES " ; M ; " MINUTES " ; S ; " SECONDES "

L'ordinateur imprime successivement sur une même ligne :

la valeur mémorisée dans la mémoire H, le mot se trouvant entre guillemets HEURES, la valeur mémorisée dans la mémoire M, le mot MINUTES, la valeur mémorisée dans la mémoire S, le mot SECONDES.

Puis l'ordinateur passe à l'instruction suivante 35.

35 FOR I = 1 TO 66Ø

6Ø NEXT I

L'ordinateur "compte" de 1 à 66Ø. Le compteur permet de laisser écouler 1 seconde.

L'ordinateur passe à l'instruction suivante 45.

45 S = S + 1

Une seconde n'est écoulée, donc on rajoute 1 à la valeur mémorisée dans la mémoire S des secondes.

L'ordinateur passe à l'instruction suivante 50

50 IF S < 60 THEN GOTO 90

Deux cas peuvent se produire :

. 1^e cas :

le nombre mémorisé dans la mémoire S est strictement inférieur à 60.

Dans ce cas l'ordinateur passe à l'instruction 90 (il saute les instructions 55 ; 60 ; 65 ; 70 ; 75 ; 80 et 85 comme si elles n'existaient pas).

90 HOME

l'écran s'efface. L'ordinateur passe à l'instruction suivante 95.

95 GOTO 30

l'ordinateur revient en arrière, il retourne à l'instruction 30.

le travail effectué par l'ordinateur est alors identique à celui que nous venons d'expliquer.

. 2^e cas :

le nombre mémorisé dans la mémoire S est égal à 60

Dans ce cas l'ordinateur passe directement à l'instruction suivante :

l'instruction 55

55 M = M + 1

60 secondes = 1 minute.

Donc on rajoute 1 à la valeur mémorisée dans la mémoire M des

minutes.

d'ordinateur passe à l'instruction suivante 6Ø

6Ø S = Ø.

"On remet la mémoire des secondes à zéro". d'ordinateur mémorise le chiffre Ø dans la mémoire S.

d'ordinateur passe à l'instruction suivante 65.

65 IF M < 6Ø THEN GOTO 9Ø

Deux cas peuvent se produire :

• 1^e cas :

Le nombre mémorisé dans la mémoire M est strictement inférieur à 6Ø.

Dans ce cas l'ordinateur passe à l'instruction 9Ø (il saute les instructions 7Ø ; 75 ; 8Ø et 85 comme si elles n'existaient pas).

9Ø HOME

d'écran s'efface. d'ordinateur passe à l'instruction suivante 95.

95 GOTO 3Ø

d'ordinateur revient en arrière, il retourne à l'instruction 3Ø.

Le travail effectué par l'ordinateur est alors identique à celui que nous venons d'expliquer.

• 2^e cas :

Le nombre mémorisé dans la mémoire M est égal à 6Ø

Dans ce cas l'ordinateur passe directement à l'instruction suivante 7Ø

7Ø H = H + 1

6Ø minutes = 1 heure.

Donc on rajoute 1 à la valeur mémorisée dans la mémoire H des heures. 141.

d'ordinateur passe à l'instruction suivante 75.

75 M = 0

"On remet à zéro" la mémoire des minutes. d'ordinateur mémorise le chiffre 0 dans la mémoire M.

d'ordinateur passe à l'instruction suivante 80

80 IF H < 24 THEN GOTO 90

Deux cas peuvent se produire :

. 1^{er} cas :

le nombre mémorisé dans la mémoire H est strictement inférieur à 24. Dans ce cas l'ordinateur passe à l'instruction 90 (il saute l'instruction 85 comme si elle n'existait pas.).

90 HOME

l'écran s'efface. d'ordinateur passe à l'instruction suivante 95.

95 GOTO 30

d'ordinateur revient en arrière, il retourne à l'instruction 30.

le travail effectué par l'ordinateur est alors identique à celui que nous venons d'expliquer.

. 2^{er} cas :

le nombre mémorisé dans la mémoire H est égal à 24.

Dans ce cas l'ordinateur passe directement à l'instruction suivante 85.

85 H = 0

" On remet à zéro " la mémoire des heures . L'ordinateur mémorise le chiffre zéro dans la mémoire H .
L'ordinateur passe à l'instruction suivante 90

90 HOME

L'écran s'efface. L'ordinateur passe à l'instruction suivante 95.

95 GOTO 30

L'ordinateur revient en arrière, il retourne à l'instruction 30 .
Le travail effectué par l'ordinateur est alors identique à celui que nous venons d'expliquer.

Instructions LEN...
MID\$...

I Instruction LEN... :

LEN est l'abréviation de LENGTH (longueur)

1) Exercice:

• Tapez A\$ = "ECOLE" ←

Le mot ECOLE n'est pas imprimé sur l'écran, il est mémorisé dans la mémoire A\$.

• Tapez ? A\$ ←

Sur l'écran apparaît :

• Tapez ? LEN(A\$) ←

Sur l'écran apparaît le chiffre :

• Remarque:

Le mot ECOLE mémorisé dans la mémoire A\$ est constitué de 5 lettres.

• Tapez B\$ = "APPRENDRE A PROGRAMMER" ←

La phrase APPRENDRE A PROGRAMMER n'est pas imprimée sur l'écran, elle est mémorisée dans la mémoire B\$

144.

• Tapez ? B \$ ←

Sur l'écran apparaît :

• Tapez ? LEN (B \$) ←

Sur l'écran apparaît le nombre :

• Remarques

La phrase APPRENDRE A PROGRAMMER mémorisée dans la mémoire B \$ est constituée de 22 caractères. L'intervalle (l'espace vide) est considéré comme un caractère.

Attention : la phrase ne doit pas contenir de guillemets.

2) a) Programme : lisez attentivement ce qui suit :

NEW ←

5 HOME ←

10 INPUT " DONNEZ UN MOT OU UNE PHRASE NE CONTENANT
PAS DE GUILLEMETS : " ; A \$ ←

15 N = LEN (A \$) ←

20 ? A \$; " EST CONSTITUE DE : " ; N ; " LETTRES OU
CARACTERES . " ←

b) Tapez le programme précédent

Demandez à l'ordinateur d'exécuter ce programme.

c) Etudiez rapidement le programme précédent :

d'écran s'efface . d'ordinateur imprime :

DONNEZ UN MOT OU UNE PHRASE NE CONTENANT PAS DE
GUILLEMETS:

d'ordinateur s'arrête , il attend une donnée alphanumérique .

Tous devez taper un mot ou une phrase ne contenant pas de guillemets
puis appuyer sur la touche \leftarrow . Votre réponse est alors mémorisée
dans la mémoire A\$

d'ordinateur mémorise dans la mémoire N le nombre total de caractères
constituant l'expression mémorisée dans A\$.

Enfin l'ordinateur imprime successivement : la phrase mémorisée dans
A\$, EST CONSTITUE DE : , le nombre mémorisé dans N ,
LETTRES OU CARACTERES .

II Instruction MID\$...

MID abréviation de MIDDLE (milieu)

1) Exercice:

• Tapez A\$ = " E C O L E " \leftarrow

• Tapez B\$ = " A P P R E N D R E A P R O G R A M M E R " \leftarrow

• Tapez ? MID\$ (A\$, 1 , 1) \leftarrow

Sur l'écran apparaît la lettre

• Tapez ? MID\$ (A\$, 2 , 1) \leftarrow

Sur l'écran apparaît la lettre

• Tapez ? M I D \$ (A \$, 3, 1) ←

Sur l'écran apparaît la lettre

• Tapez ? M I D \$ (A \$, 4, 1) ←

Sur l'écran apparaît la lettre

• Tapez ? M I D \$ (A \$, 5, 1) ←

Sur l'écran apparaît la lettre

• Remarques :

M I D \$ (A \$, 1, 1) représente la première lettre "du mot" A \$

M I D \$ (A \$, 2, 1) représente la deuxième lettre "du mot" A \$

M I D \$ (A \$, 3, 1) représente la troisième lettre "du mot" A \$

....

2) Exercice 2 :

- Quelle instruction dois-je taper pour donner l'ordre à l'ordinateur d'imprimer sur l'écran le 5^e caractère de la phrase mémorisée dans la mémoire B \$? (Vous pouvez vous aider de l'exercice précédent)

• Quelle instruction dois-je taper pour donner l'ordre à l'ordinateur d'imprimer sur l'écran le 20^e caractère de la phrase mémorisée dans la mémoire B\$?

3) Exercice 3:

• Tapez ? MID\$(A\$, 1, 2) ←

Sur l'écran apparaît:

• Tapez ? MID\$(A\$, 2, 2) ←

Sur l'écran apparaît:

• Tapez ? MID\$(A\$, 3, 2) ←

Sur l'écran apparaît:

• Tapez ? MID\$(A\$, 4, 2) ←

Sur l'écran apparaît:

• Remarques:

MID\$(A\$, 1, 2) représente 2 lettres consécutives du mot mémorisé dans A\$ à partir de la 1^e.

$MID \$ (A \$, 2, 2)$ représente 2 lettres consécutives du mot mémorisé dans $A \$$ à partir de la 2^e.

$MID \$ (A \$, 3, 2)$ représente 2 lettres consécutives du mot mémorisé dans $A \$$ à partir de la 3^e.

$MID \$ (A \$, 4, 2)$ représente 2 lettres consécutives du mot mémorisé dans $A \$$ à partir de la 4^e.

4) Exercice 4:

- Quelle instruction dois-je taper pour donner l'ordre à l'ordinateur d'imprimer sur l'écran 2 caractères consécutifs de l'expression mémorisée dans $B \$$ à partir du 3^e caractère ?

- Quelle instruction dois-je taper pour donner l'ordre à l'ordinateur d'imprimer sur l'écran 2 caractères consécutifs de l'expression mémorisée dans $B \$$ à partir du 20^e caractère ?

5) Exercice 5:

• Tapez ? MID \$ (A \$, 1 , 3) ←

Sur l'écran apparaît:

• Tapez ? MID \$ (A \$, 2 , 3) ←

Sur l'écran apparaît:

• Tapez ? MID\$(A\$, 3, 3) ←

Sur l'écran apparaît:

• Remarques:

MID\$(A\$, 1, 3) représente 3 lettres consécutives du mot mémorisé dans A\$ à partir de la 1^e.

MID\$(A\$, 2, 3) représente 3 lettres consécutives du mot mémorisé dans A\$ à partir de la 2^e.

MID\$(A\$, 3, 3) représente 3 lettres consécutives du mot mémorisé dans A\$ à partir de la 3^e.

6) Exercice 6:

- Quelle instruction dois-je taper pour donner l'ordre à l'ordinateur d'imprimer sur l'écran 3 caractères consécutifs de l'expression mémorisée dans B\$ à partir du 2^e caractère ?

- Quelle instruction dois-je taper pour donner l'ordre à l'ordinateur d'imprimer sur l'écran 3 caractères consécutifs de l'expression mémorisée dans B\$ à partir du 18^e caractère ?

7) Exercice 7:

• Tapez ? MID\$ (A\$, 1, 4) ←

Sur l'écran apparaît:

• Tapez ? MID\$ (A\$, 2, 4) ←

Sur l'écran apparaît:

• Remarques:

MID\$ (A\$, 1, 4) représente 4 lettres consécutives du mot mémorisé dans A\$ à partir de la 1^e.

MID\$ (A\$, 2, 4) représente 4 lettres consécutives du mot mémorisé dans A\$ à partir de la 2^e.

8) Exercice 8:

• Quelle instruction dois-je taper pour donner l'ordre à l'ordinateur d'imprimer sur l'écran 4 caractères consécutifs de l'expression mémorisée dans B\$ à partir du 2^e?

• Quelle instruction dois-je taper pour donner l'ordre à l'ordinateur d'imprimer sur l'écran 7 caractères consécutifs de l'expression mémorisée dans B\$ à partir du 1^e?

III Exemples de programmes utilisant les instructions précédentes:

151.

1) a) Problème 1:

Recherchons un programme permettant de déterminer le nombre de consonnes utilisées pour écrire un mot ou une phrase.

b) Programme: lisez attentivement ce qui suit:

```
NEW          ← ]
5 HOME      ← ]
10 ? " DETERMINATION DU NOMBRE DE CONSONNES DANS
    UN MOT OU UNE PHRASE ." ← ]
15 ? ← ]
20 ? " ECRIVEZ UN MOT OU UNE PHRASE EN LETTRES
    MAJUSCULES: " ← ]
25 ? ← ]
30 INPUT A$ ← ]
35 ? ← ]
40 FOR I=1 TO LEN(A$) ← ]
45 B$ = MID$(A$, I, 1) ← ]
50 IF B$ < > "A" AND B$ < > "E" AND
    B$ < > "I" AND B$ < > "O" AND B$ < > "U"
    AND B$ < > "Y" AND B$ < > " " THEN
    C = C + 1 ← ]
55 NEXT I ← ]
60 IF C < > 1 AND C < > 0 THEN GOTO 75 ← ]
65 ? A$ ; " EST CONSTITUE DE " ; C ; " CONSONNE "
    ← ]
```

152.

```
70 GOTO 80 ← ]
75 ? A$ ; " EST CONSTITUE DE " ; C ; " CONSONNES " ← ]
80 ? ← ]
85 C = 0 ← ]
90 INPUT " VOULEZ VOUS RECOMMENCER ? . REPONDEZ PAR
    OUI OU NON : " ; D$ ← ]
95 IF D$ = " OUI " OR D$ = " oui " THEN GOTO 15 ← ]
100 IF D$ = " NON " OR D$ = " non " THEN GOTO 115 ← ]
105 ? " LA REPONSE A LA QUESTION PRECEDENTE N'EST
    PAS CORRECTE . VOUS DEVEZ REPONDRE PAR OUI OU NON . "
    ← ]
110 GOTO 80 ← ]
115 END ← ]
```

c) Tapez le programme précédent.

Attention:

Le programme permet de compter le nombre de consonnes d'une phrase .
Mais cette phrase ne doit contenir aucune ponctuation , elle doit
uniquement être composée de lettres de l'alphabet et d'intervalles (pour
séparer les mots) .

Demandez à l'ordinateur d'exécuter les instructions de ce programme .

d) Etude du programme précédent:

Reprenons les instructions du paragraphe III 1) a) et suivons "pas à pas"
le travail effectué par l'ordinateur .

• NEW ←]

Cette instruction permet d'effacer tout ancien programme de la mémoire centrale de l'ordinateur.

- En tapant un numéro de ligne devant chacune des autres instructions nous en avons retardé l'exécution.

- En tapant :

RUN ←

l'ordinateur exécute les instructions dans l'ordre croissant des numéros.

Il revient donc à l'instruction 5.

5 HOME

l'écran s'efface. l'ordinateur passe à l'instruction suivante 10

10 ? " DETERMINATION DU NOMBRE DE CONSONNES DANS UN MOT OU UNE PHRASE. "

l'ordinateur imprime :

DETERMINATION DU NOMBRE DE CONSONNES DANS UN MOT
OU UNE PHRASE.

Puis il passe à l'instruction 15

15 ?

l'ordinateur saute une ligne et passe à l'instruction 20

20 ? " ECRIVEZ UN MOT OU UNE PHRASE EN LETTRES

MASUSCULES:

l'ordinateur imprime :

ECRIVEZ UN MOT OU UNE PHRASE EN LETTRES MAJUSCULES:

Puis il passe à l'instruction suivante 25

25 ?

d'ordinateur saute une ligne et passe à l'instruction suivante 30

30 INPUT A\$

d'ordinateur s'arrête. Il attend une donnée alphanumérique.

Vous devez taper votre réponse (un mot ou une phrase en lettres majuscules) puis appuyer sur la touche \leftarrow . Votre réponse est alors mémorisée dans la mémoire A\$.

d'ordinateur passe à l'instruction 35

35 ?

d'ordinateur saute une ligne et passe à l'instruction 40

40 FOR I=1 TO LEN(A\$)

LEN(A\$) représente le nombre total de lettres du mot (ou de la phrase) mémorisé dans la mémoire A\$.

L'ordinateur mémorise la valeur 1 dans la mémoire I puis passe à l'instruction suivante 45

45 B\$ = MID\$(A\$, I, 1)

MID\$(A\$, I, 1) représente la 1^e lettre du mot (ou 1^e caractère de la phrase) mémorisé dans A\$.

L'ordinateur mémorise la 1^e lettre du mot (ou 1^e caractère de la phrase) dans la mémoire B\$. Puis il passe à l'instruction 50

50 IF B\$ < > "A" AND B\$ < > "E" AND B\$ < > "I"

$\text{AND } B \S < > "O" \quad \text{AND } B \S < > "U" \quad \text{AND } B \S < > "Y"$
AND B § < > " " THEN C = C + 1

Deux cas peuvent se produire :

* 1^{er} cas :

Le 1^{er} caractère, du mot ou de la phrase, mémorisé dans B § n'est égal à aucune des voyelles A ; E ; I ; O ; U ; Y et ne représente pas un intervalle (espace vide). Dans ce cas l'ordinateur ajoute 1 à la valeur mémorisée dans la mémoire représentant le nombre de consonnes.

(N'oublions pas que : au début de l'exécution du programme la valeur 0 est mémorisée dans C)

* 2^{ème} cas :

Le 1^{er} caractère mémorisé dans B § est soit une voyelle soit un intervalle, dans ce cas l'ordinateur passe à l'instruction suivante SS

SS NEXT I

L'ordinateur retourne à l'instruction 4 Ø (instruction FOR ... TO ...) en donnant à I la valeur 2 (valeur qui suit 1) et il recommence le travail précédent.

En résumé :

L'ordinateur isole un à un chacun des caractères du mot ou de la phrase mémorisé dans A §. Il compare chaque caractère aux voyelles A, E, I, O, U, Y et à l'intervalle (espace vide). Si ce caractère n'est ni une voyelle ni un intervalle il ajoute 1 au nombre de consonnes mémorisées dans C. (D'où l'importance d'utiliser uniquement des lettres et des intervalles pour écrire la phrase mémorisée dans A §)

Après avoir effectué ce travail pour le dernier caractère de l'expression mémorisée dans A § l'ordinateur passe à l'instruction 6 Ø

60 IF C < > 1 AND C < > 0 THEN GOTO 75

Deux cas peuvent se produire :

* 1^{er} cas :

Le nombre de consonnes mémorisé dans C n'est égal ni à 1 ni à 0 .

Dans ce cas l'ordinateur passe à l'instruction 75 (il saute les instructions 65 et 70 comme si elles n'existaient pas).

75 ? A\$; " EST CONSTITUE DE " ; C ; " CONSONNES "

l'ordinateur imprime successivement l'expression mémorisée dans A\$,

EST CONSTITUE DE , le nombre de consonnes mémorisé dans C ,

CONSONNES

l'ordinateur passe à l'instruction suivante 80

80 ?

l'ordinateur saute une ligne et passe à l'instruction 85

85 C = 0

l'ordinateur mémorise la valeur 0 dans la mémoire C . Cette précaution est indispensable pour éviter tout risque d'erreur si vous souhaitez demander une nouvelle exécution de ce programme .

l'ordinateur passe à l'instruction suivante 90

.....

Les explications des dernières instructions ont déjà été données dans des chapitres antérieurs .

* 2^{es} cas :

Le nombre de consonnes mémorisé dans C est égal à 1 ou à 0 .

Dans ce cas l'ordinateur passe à l'instruction suivante 65

65 ? A\$; " EST CONSTITUE DE " ; C ; " CONSONNE "

l'ordinateur imprime successivement l'expression mémorisée dans A\$,

EST CONSTITUE DE , le nombre de consonnes mémorisé dans C , CONSONNE

Puis l'ordinateur passe à l'instruction suivante 70

70 GOTO 80

l'ordinateur passe à l'instruction 80 (il saute l'instruction 75)

80 ?

l'ordinateur saute une ligne et passe à l'instruction suivante 85

.....

.....

e) Remarque:

Entrez l'instruction 85 , puis demandez à l'ordinateur d'exécuter les instructions du programme pour différentes expressions .

Etudiez ce qui se passe .

2) Problème 2:

Recherchez un programme permettant de déterminer le nombre de voyelles utilisées pour écrire un mot ou une phrase .

3) a) Problème 3:

Déterminons un programme permettant de conjuguer tous les verbes réguliers du premier groupe (ER) au présent .

b) Programme: lisez attentivement ce qui suit :

NEW ←

S HOME ←

10 ? " CONJUGAISON AU PRESENT DES VERBES REGULIERS

DU PREMIER GROUPE (ER) " ←

15 ? ←

20 ? "DONNEZ UN VERBE REGULIER DU PREMIER
GROUPE : "

25 ?

30 INPUT A\$

35 N = LEN(A\$) - 2

40 B\$ = MID\$(A\$, 1, N)

45 ? "JE " ; B\$; "E"

50 ? "TU " ; B\$; "ES"

55 ? "IL " ; B\$; "E"

60 ? "NOUS " ; B\$; "ONS"

65 ? "VOUS " ; B\$; "EZ"

70 ? "ILS " ; B\$; "ENT"

c) Rappel :

Voici une liste des verbes irréguliers du 1^{er} groupe :

- . ALLER
- . verbes en CER
- . verbes en GER
- . verbes en IER
- . verbes en OYER et UYER
- . verbes en ELER et ETTER
- . verbes ayant un E ou un É (accent aigu) à l'avant-dernière syllabe.

d) Tapez le programme précédent et demandez à l'ordinateur d'exécuter les instructions de ce programme.

e) quelques explications:

considérons le verbe CHANTER :

JE CHANTE

TU CHANTES

IL CHANTE

NOUS CHANTONS

VOUS CHANTEZ

ILS CHANTENT

Pour conjuguer le verbe CHANTER au présent il faut conserver toutes les lettres de ce verbe sauf les deux dernières.

Dans le programme précédent

$LEN(A\$\)$ désigne le nombre total de lettres du verbe conservé dans $A\$\)$

$LEN(A\$\) - 2$ est mémorisé dans la mémoire N , $LEN(A\$\) - 2$

désigne le nombre de lettres du verbe lorsque les deux dernières lettres ont été enlevées.

$B\$\)$ désigne la mémoire dans laquelle est mémorisé $MID\$(A\$, 1, N)$.

$MID\$(A\$, 1, N)$ représente N lettres du verbe mémorisé dans $A\$\)$ à partir de la première lettre.

Donc $MID\$(A\$, 1, N)$ dans le cas du verbe CHANTER représente CHANT

IV Quelques idées de programmes:

Vous pouvez déterminer des programmes permettant de conjuguer à l'imparfait, passé simple, futur simple ... les verbes réguliers du premier groupe.

160-

Puis vous pourrez déterminer des programmes permettant de conjuguer les verbes irréguliers du premier groupe.

Enfin tous ces programmes effectués pour les verbes du premier groupe peuvent être déterminés par les verbes du deuxième et troisième groupe.

Toute représentation, traduction, adaptation ou reproduction, même partielle
par tous procédés, en tous pays, faite sans autorisation préalable est illicite
et exposerait le contrevenant à des poursuites judiciaires. (loi du 11 mars
1975.)
Dépot légal: **NOVEMBRE 1986**

Stupurenendre à progrommer 8. Jahre

AMSTRAD

CPC



MÉMOIRE ÉCRITE
MEMORY ENGRAVED
MEMORIA ESCRITA



<https://acpc.me/>

[FRA] Ce document a été préservé numériquement à des fins éducatives et d'études, et non commerciales.

[ENG] This document has been digitally preserved for educational and study purposes, not for commercial purposes.

[ESP] Este documento se ha conservado digitalmente con fines educativos y de estudio, no con fines comerciales.