

**JEAN-FRANÇOIS
SEHAN**

**ETUDES
POUR ZX81 TOME 2**

**20 PROGRAMMES EN BASIC
ET EN ASSEMBLEUR**



EDITIONS DU P.S.I.

PROGRAMMES

PROGRAMMES

ETUDES POUR ZX 81

TOME 2

20 PROGRAMMES EN BASIC ET EN ASSEMBLEUR

COLLECTION LANGAGES

Dictionnaire du Basic — David A. Lien traduit par Yves Leclerc
Langages de programmation — Stéphane Berche et Claude Lhermitte
Le langage ADA — Daniel-Jean David
Programmer en APL — Daniel-Jean David
Programmer en Assembleur — Alain Pinaud
Programmer en Basic — Michel Plouin
Le Basic et ses Fichiers — tomes 1 et 2 — Jacques Boisgontier
Programmer en Fortran — Daniel-Jean David
Programmer en LSE — Stéphane Berche et Yves Noyelle
Programmer en Pascal — Daniel-Jean David et Jean-Luc Deschamps
Comment Programmer — Jean-Claude Barbance
Comprendre les microprocesseurs — Roland Dubois

COLLECTION MATÉRIELS

La découverte de l'Applesoft — tome 1 — Frédéric Lévy et Dominique Schraen
La découverte de l'Applesoft — tome 2 — Frédéric Lévy
La pratique de l'Apple II volumes I et II — Nicole Bréaud Pouliquen
La pratique de l'Apple II volume III — Nicole Bréaud Pouliquen et Daniel-Jean David
La découverte du FX-702P — Jean-Pierre Richard
La découverte du Goupil — Jean-Yves Michel
Programmer HP-41 — Philippe Descamps et Jean-Jacques Dhénin
La pratique du MZ-80 A et K — Jean-Pierre Lhoir
La découverte du PC-1211 — Jean-Pierre Richard
La découverte du CBM — Daniel-Jean David
La pratique du CBM — volumes I et II — Daniel-Jean David
La pratique du TRS-80 — volumes I, II et III — Pierre Giraud et Alain Pinaud
Exercices sur TRS-80 — Frédéric Lévy
Les graphiques sur TRS-80 — Donn Inman traduit par Alain Pinaud
La découverte de la TI-57 — Xavier de la Tullaye
La découverte de la TI-99/4A — Frédéric Lévy et Dominique Schraen
La découverte du VIC — Daniel-Jean David
La pratique du VIC — volume 1 — Daniel-Jean David
La pratique du ZX-81 — tome 1 — Xavier Linant de Bellefonds
La pratique du ZX-81 — tome 2 — Marcel Henrot
Le petit livre du ZX-81 — Trevor Toms traduit par Ghislaine Lapeyre

COLLECTION GUIDES PRATIQUES

L'APL sur TRS-80 — Claude Nowakowski
CP/M pas à pas — Alain Pinaud
LISP sur Apple II — Nicole Bréaud Pouliquen
La réalisation des programmes — Michel Benelfoul
Méthodes de calcul numérique — Claude Nowakowski
Visicalc sur Apple — Hervé Thiriez
Visicalc sur TRS-80 — Hervé Thiriez

COLLECTION PROGRAMMES

Collège, pochettes et maths — Jacques Deconchat
Études pour ZX-81 — tomes 1 et 2 — Jean-François Séhan
Jeux, trucs et comptes pour PET/CBM — Michel Benelfoul
Jeux, trucs et comptes pour TRS-80 — Michel Benelfoul et Denis Chauvin
Mathématiques et statistiques — Hervé Haut
Modèles pratiques de décision — tomes 1 et 2 — Jean-Pierre Blanger
Récréations pour TI-57 — tomes 1 et 2 — Jacques Deconchat
Variations pour PC-1211 — Jean-François Séhan

COLLECTION MEMENTOS

Clefs pour le CBM — Daniel-Jean David
Le Basic de A à Z — Jacques Boisgontier

COLLECTION UTILISATIONS DE L'ORDINATEUR

Le Basic et l'école — Jacques Gouet
La comptabilité sur Apple II — Gérard Lilio et Serge Lilio
Les finances familiales — Jean-Claude Barbance

INITIATION

Visa pour l'informatique — Jean-Michel Jégo
Mon ordinateur — Jean-Claude Barbance
L'ordinateur individuel — Yves Leclerc (en coédition avec S.C.E. Canada)

La loi du 11 mars 1957 n'autorisant, aux termes des alinéas 2 et 3 de l'article 41, d'une part, que les « copies ou reproductions strictement réservées à l'usage privé du copiste et non destinées à une utilisation collective » et, d'autre part, que les analyses et les courtes citations dans un but d'exemple et d'illustration, « toute représentation ou reproduction intégrale, ou partielle, faite sans le consentement de l'auteur ou de ses ayants droit ou ayants cause, est illicite » (alinéa 1^{er} de l'article 40).

Cette représentation ou reproduction, par quelque procédé que ce soit, constituerait donc une contrefaçon sanctionnée par les articles 425 et suivants du Code Pénal.

JEAN-FRANÇOIS
SEHAN

Présentation

**ETUDES POUR
ZX 81
TOME 2
20 PROGRAMMES EN BASIC
ET EN ASSEMBLEUR**



Editions du P.S.I.
41-51, rue Jacquard
77400 Lagny/Marne

Présentation

C'est plus particulièrement au langage assembleur appliqué au module d'extension comme l'imprimante ou la carte génératrice de caractères que sont dédiées ces nouvelles "études" pour ZX81.

A travers les différents programmes proposés et pour lesquels vous trouverez une présentation et des exemples de ce que l'on obtient sur l'écran, l'auteur vous engage à exercer vos talents de compositeur (grâce à la carte sonore et au générateur de caractères), à dessiner des histogramme (grâce à la carte haute résolution) et à jouer, sans extension, au Baccara, aux Piranhas et au Taquin.

La structure générale des programmes est présentée sous forme d'un ou plusieurs organigrammes détaillés. Chaque ligne est ensuite abondamment commentée, expliquant les astuces utilisées. Un exemple d'exécution et un tableau d'identification des variables complète chaque "étude".

Si l'auteur a opté pour cette présentation pédagogique c'est pour vous conduire, à partir des cas étudiés, à élaborer vous même de nouveaux programmes.

Du même auteur :

- Variations pour PC-1211*
- Etudes pour ZX81- Tome I*

TABLAU DES EXERCICES

EXERCICES			
EXERCICE	PROBLEME	RESOLUTION	INDICATEUR
MUSIQUE			
BALISTIQUE			
HORLOGE			
CUISINE			
LABYRINTHE			
MORSE			
HISTOGRAMME			
PARITES			
PIRANHAS			
BACCARA			
GENERATEUR DE CARACTERES			
CIRCUIT 24			
DEBARQUEMENT IMMEDIAT			
GRAPHIQUES			
TAQUIN			
SOUCOUPES VOLANTES			
FACTURATION			
TRIS...SWEET TRIS			
TEST MEMOIRE			
ESPERANCE DE VIE			

Sommaire

Pages

MUSIQUE	9
BALISTIQUE	19
HORLOGE	27
CUISINE	35
LABYRINTHE	43
MORSE	51
HISTOGRAMME	59
PARITES	67
PIRANHAS	75
BACCARA	83
GENERATEUR DE CARACTERES	93
CIRCUIT 24	103
DEBARQUEMENT IMMEDIAT	111
GRAPHIQUES	119
TAQUIN	123
SOUCOUPES VOLANTES	129
FACTURATION	137
TRIS...SWEET TRIS	151
TEST MEMOIRE	161
ESPERANCE DE VIE	167

TABLEAU DES EXTENSIONS

NOM DES PROGRAMMES	EXTENSIONS			
	IMPRIMANTE	CARTE SON	GENERATEUR DE CARACTERES	HAUTE RESOLUTION
BACCARA				
BALISTIQUE				
CIRCUIT 24				
CUISINE				
DEBARQUEMENT IMMEDIAT				
ESPERANCE DE VIE				
FACTURATION				
GENERATEUR DE CARACTERES			●	
GRAPHIQUES				●
HISTOGRAMME				●
HORLOGE				
LABYRINTHE	●			
MORSE		●		
MUSIQUE		●	●	
PARITES				
PIRANHAS				
SOUCOUPES VOLANTES				
TAQUIN				
TEST				
TRIS ... SWEET TRIS				

Avertissement

Vue la complexité des programmes de ce livre, il est nécessaire de posséder l'extension 16 ko du ZX81, la version de base n'ayant qu'un petit ko de mémoire vive.

Certains programmes nécessitent l'utilisation d'une ou deux extensions. Vérifiez dans le tableau de la page précédente la configuration que vous devez installer.

Dès que les programmes sont entrés dans le ZX81, sauvez-les sur votre magnétophone. Pour les programmes très longs, il est conseillé d'effectuer une première sauvegarde au milieu de la frappe du programme. N'attendez pas qu'il soit entièrement entré et testé... Une coupure de courant est si vite arrivée !

*Méfiez-vous ! L'instruction **FAST** coupe complètement le signal vidéo de votre téléviseur, laissant, pendant quelques instants, l'impression que votre ZX81 ne marche plus. Il est toujours possible d'arrêter le programme en cours par la touche **BREAK** si vous pensez qu'il est arrivé une catastrophe et avant d'effectuer une manoeuvre qui perdrait définitivement votre programme.*

Pour recopier les lignes comportant des caractères graphiques dont le découpage ne vous paraîtrait pas clair, reportez-vous aux exemples donnés avec chaque listing. Aidez-vous aussi des lignes précédentes pour repérer chaque caractère, en comparant avec le nombre et la position des caractères de ces autres lignes.

<p>NOTE IMPORTANTE : les caractères qui apparaissent en minuscule sur les listings doivent être tapés du clavier en majuscule et en inverse vidéo (mode graphique : shift 9).</p>
--

Musique

Mélomanes, l'auteur a pensé à vous !

Votre ZX81 devient tout à coup un instrument de musique qui ne se contente pas d'émettre des sons mais qui les affiche aussi à l'écran.

Juste après le lancement du programme, déplacez le commutateur de la carte génératrice de caractères pour obtenir une portée avec la clef de sol ; puis allumez l'amplificateur relié à la carte son.

La petite flèche indique l'emplacement de la prochaine note. Les notes vont de DO (touche 1) au MI de l'octave suivante (touche 0) comme l'indique le tableau suivant :

<i>Note</i>	<i>Touche du clavier</i>
DO	1
RE	2
MI	3
FA	4
SOL	5
LA	6
SI	7
DO	8
RE	9
MI	0

Pour déplacer le curseur (la petite flèche), utilisez la touche **M** pour revenir en arrière (M comme moins) et **P** pour avancer (P comme Plus). On peut choisir une noire ou une blanche avant d'entrer la note en tapant soit la lettre **N** soit la lettre **B**.

Dès que votre mélodie sera réalisée, appuyez sur **NEWLINE** pour l'entendre.

Le programme comporte une partie en assembleur. Il s'agit programme "CHRØ" livré avec la carte génératrice de caractères. Chargez cette cassette et tapez à la suite les lignes 80 à 2130 en ayant soin de ne pas modifier les lignes 1 à 14.

N'essayez surtout pas de recopier ces dernières à partir du listing car il est impossible de connaître les codes de certains caractères. Ce qui peut paraître comme un point d'interrogation (code 15) peut très bien être un code 125.

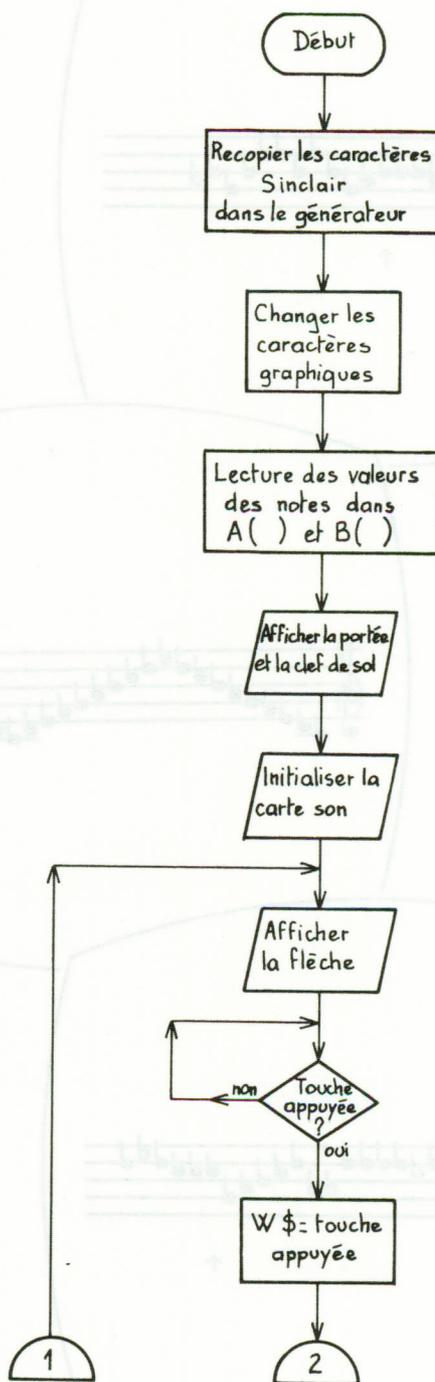
Pour les veinards qui possèdent une imprimante, la touche C permet une copie d'écran avec les caractères graphiques (voir l'exemple de la page suivante).

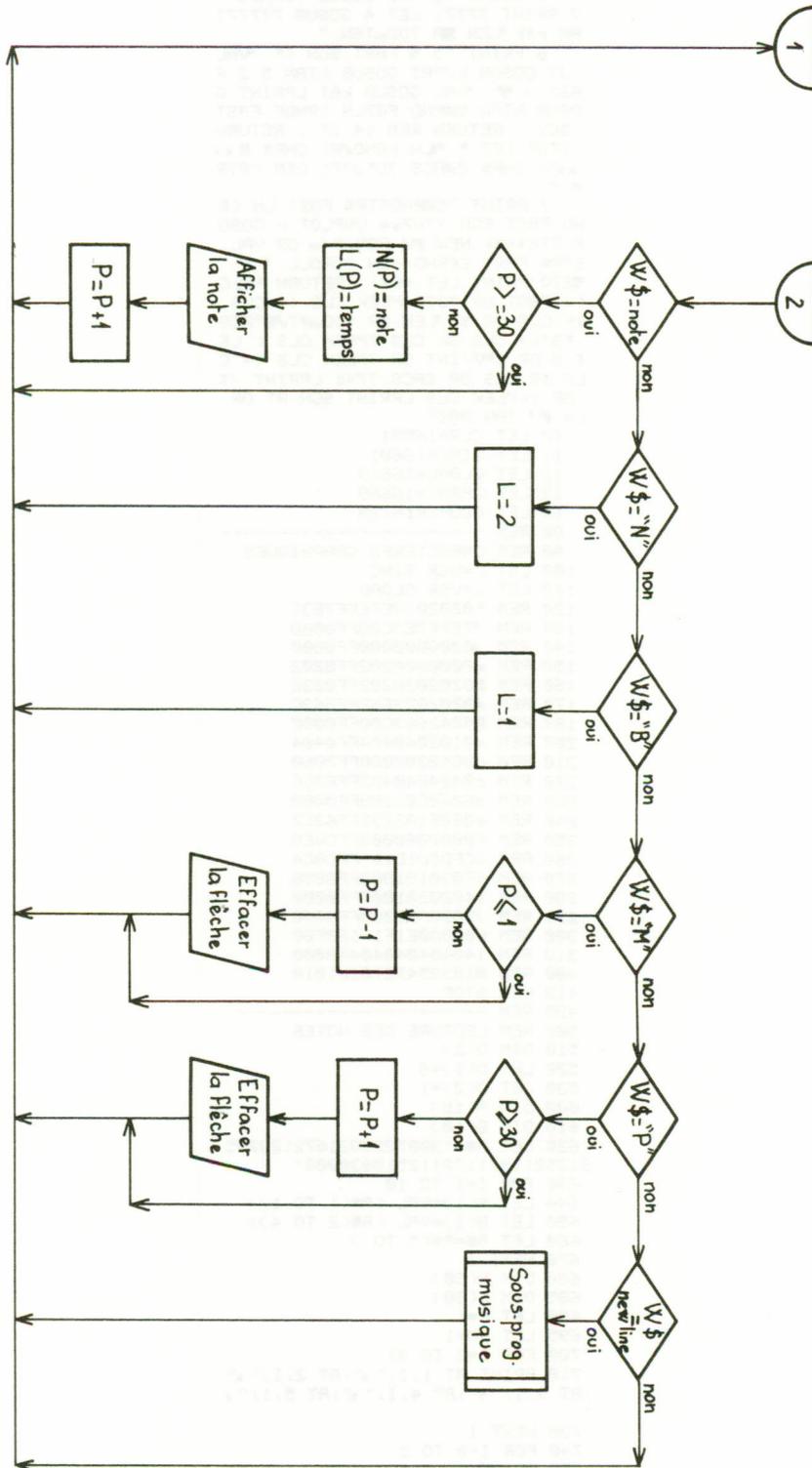
Mémoires, l'auteur a pensé à vous !
 Votre ZX81 devient tout à coup un instrument de travail qui ne se contente pas d'émettre des sons mais qui les écoute aussi à l'écran.
 Juste après le lancement du programme, déplacez le curseur vers la carte génératrice de caractères pour obtenir une portée avec le ciel de soi ; puis appuyez l'amplificateur relié à la carte son.
 La petite flèche indique l'emplacement de la prochaine note (les notes vont de DO (touche I) au MI de l'octave suivante (touche P) comme l'indique le tableau suivant :

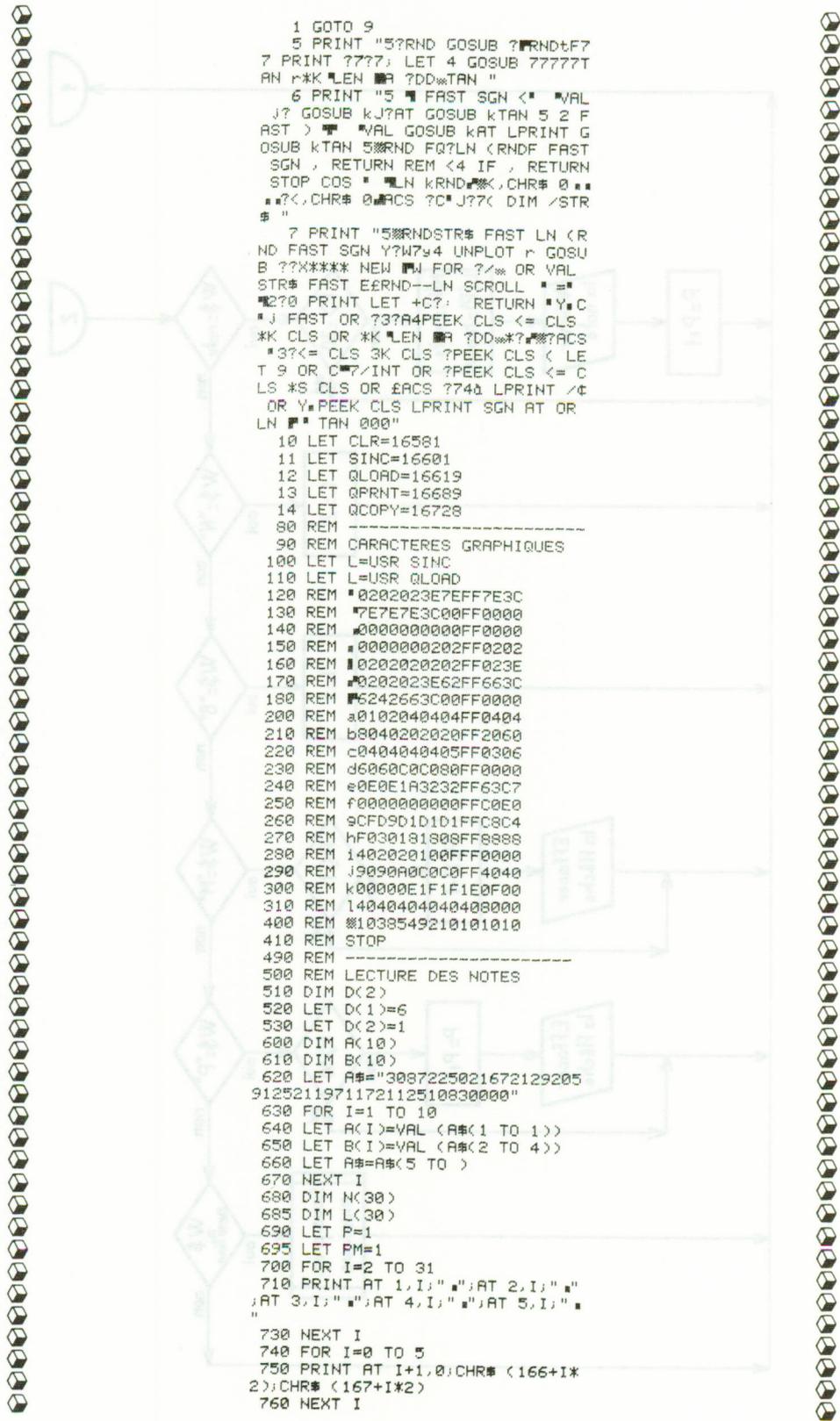
Notes	Touche à appuyer
DO	I
RE	J
MI	K
FA	L
SOL	M
LA	N
SI	O
DO	P
RE	Q
MI	R

Pour déplacer le curseur (la petite flèche), utilisez la touche N pour revenir en arrière (M comme moins) et P pour avancer (P comme plus). On peut choisir une note ou une flèche avant d'entrer la note en tapant soit la lettre N ou la lettre R.

Dès que votre mélodie sera réalisée, appuyez sur NEWLINE pour l'entendre.
 Le programme comporte une partie en assembleur. Il s'agit du programme "CHR" livré avec la carte génératrice de caractères. Changez cette cassette et tapez à la suite les lignes 84 à 112 en ayant soin de ne pas modifier les lignes 1 à 14.







ETUDES POUR ZX81

```

980 REM -----
990 REM SAISIE DES NOTES
1000 LET A=32767
1010 LET D=32766
1020 POKE A,7
1030 POKE D,254
1080 LET L=2
1090 PRINT AT 8,P+1;"# "
1100 IF INKEY#="" THEN GOTO 1100
1110 LET W#=INKEY#
1115 IF CODE W#>27 AND CODE W#<3
8 THEN GOTO 1180
1120 IF CODE W#=118 THEN GOTO 20
00
1130 IF W#="N" THEN GOTO 1310
1140 IF W#="B" THEN GOTO 1330
1150 IF W#="M" THEN GOTO 1350
1160 IF W#="P" THEN GOTO 1380
1165 IF W#="C" THEN LET C=USR QC
OPY
1170 GOTO 1100
1180 IF P>=30 THEN GOTO 1100
1190 LET W=VAL W#
1195 IF W=0 THEN LET W=10
1200 LET N<P>=W
1205 LET L<P>=L
1210 LET PY=7-INT ((W/2)+.5)
1215 PRINT AT 1,P+1;"#";AT 2,P+1
;"#";AT 3,P+1;"#";AT 4,P+1;"#";A
T 5,P+1;"#";AT 6,P+1;" "
1220 IF W/2=INT (W/2) THEN GOTO
1260
1230 PRINT AT PY,P+1;CHR# D<L>
1240 PRINT AT PY-1,P+1;CHR# 4
1250 GOTO 1280
1260 PRINT AT PY,P+1;CHR# <D<L>+
1)
1270 PRINT AT PY-1,P+1;CHR# 5
1280 PRINT AT 8,P+1;" "
1290 LET P=P+1
1295 IF P<PM THEN LET PM=PM+1
1300 GOTO 1080
1310 LET L=2
1320 GOTO 1100
1330 LET L=1
1340 GOTO 1100
1350 IF P>1 THEN LET P=P-1
1360 PRINT AT 8,P+2;" "
1370 GOTO 1090
1380 IF P<PM THEN LET P=P+1
1390 PRINT AT 8,P;" "
1400 GOTO 1090
1980 REM -----
1990 REM MUSIQUE .....
2000 FOR I=1 TO PM-1
2010 POKE A,0
2020 POKE D,B<N<I>>
2030 POKE A,1
2040 POKE D,A<N<I>>
2045 POKE A,8
2048 POKE D,15
2050 FOR J=1 TO 10/L<I>
2060 NEXT J
2100 POKE A,8
2110 POKE D,0
2120 NEXT I
2130 GOTO 1100

```

LE PROGRAMME

Lignes 1 à 14

Il s'agit du programme "CHR\$" livré avec la carte génératrice de caractères. Vous n'avez donc pas à retaper cette partie et vous devez tout simplement charger ce programme. Vous trouverez tous les renseignements utiles dans la documentation.

Lignes 160 à 410

C'est la seule partie utilisant le programme en langage machine. On recopie les caractères SINCLAIR (ligne 100) et l'on charge ceux de la musique (notes, portée et clef de sol : lignes 110 à 400). Le "stop" de la ligne 410 est obtenu en tapant sur votre clavier SHIFT A et non le "stop" lettre par lettre.

Lignes 510 à 760

Toutes les valeurs des notes sont placées dans la chaîne A\$. Le premier caractère correspond à la valeur grave, les deux suivants à l'aigu. On les place respectivement dans les tableaux A et D. Les lignes 700 à 760 dessinent à l'écran la portée et la clef de sol.

Lignes 1000 à 1030

La carte son nécessite une initialisation de son registre.

La variable A (adresse 32767) correspond à l'adresse du registre, la variable B correspondant à l'adresse des datas. La valeur 254 dans le registre 7 branche la tonalité sur le canal "A".

Lignes 1100 à 1400

On saisit au clavier les commandes de l'utilisateur qui sont réparties aux lignes 1115 à 1165. Les notes sont placées dans le tableau N et leur durée dans le tableau L (2 = noire, 1 = blanche).

Lignes 2000 à 2130

Cette partie du programme envoie sur la carte son la musique stockée dans les tableaux N et L.

Le registre 0 (POKE A,0) correspond au ton grave, le registre 1 (POKE A,1) au ton aigu, le registre 8 étant le volume du canal "A". Une valeur 15 donne une puissance maximum. Après la boucle d'attente, on remet le registre à 0, c'est-à-dire au niveau le plus bas.

IDENTIFICATION des VARIABLES

NOM SYMBOLIQUE	TYPE (*)	ORIGINE			LONG.	OBSERVATIONS
		Clavier	Programme	Cassette		
A, \$	VARIABLE	X				Liste des valeurs des notes
A	CONSTANTE	X				Adresse du registre de la carte son
A ()	TABLEAU	X				Fréquence basse
B ()	TABLEAU	X				Fréquence haute
C L R	CONSTANTE	X				Point d'entrée d'effacement du générateur de caractères
D	CONSTANTE	X				Adresse des données de la carte son
D ()	TABLEAU	X				Liste des caractères semi-graphiques
I	INDICE	X				Boucle FOR-NEXT
L	VARIABLE	X				Variable d'appel des sous-programmes du générateur de caractères
L ()	TABLEAU	X				Durée des notes
N ()	TABLEAU	X				Notes
P	COMPTEUR	X				Pointeur des notes
P M	COMPTEUR	X				Position des notes
Q C O P Y	CONSTANTE	X				Point d'entrée de recopie d'écran
Q L O A D	CONSTANTE	X				Point d'entrée de chargement d'un caractère
Q P R N T	CONSTANTE	X				Point d'entrée de la fonction PRINT
S, I, N, C	CONSTANTE	X				Point d'entrée pour recopier le générateur de caractères du ZX81
						.../...

(*) Légende : aiguillage, compteur, constante, index, indice, tableau, variable. Reproduction interdite (c) Editions du P.S.I.

Balistique



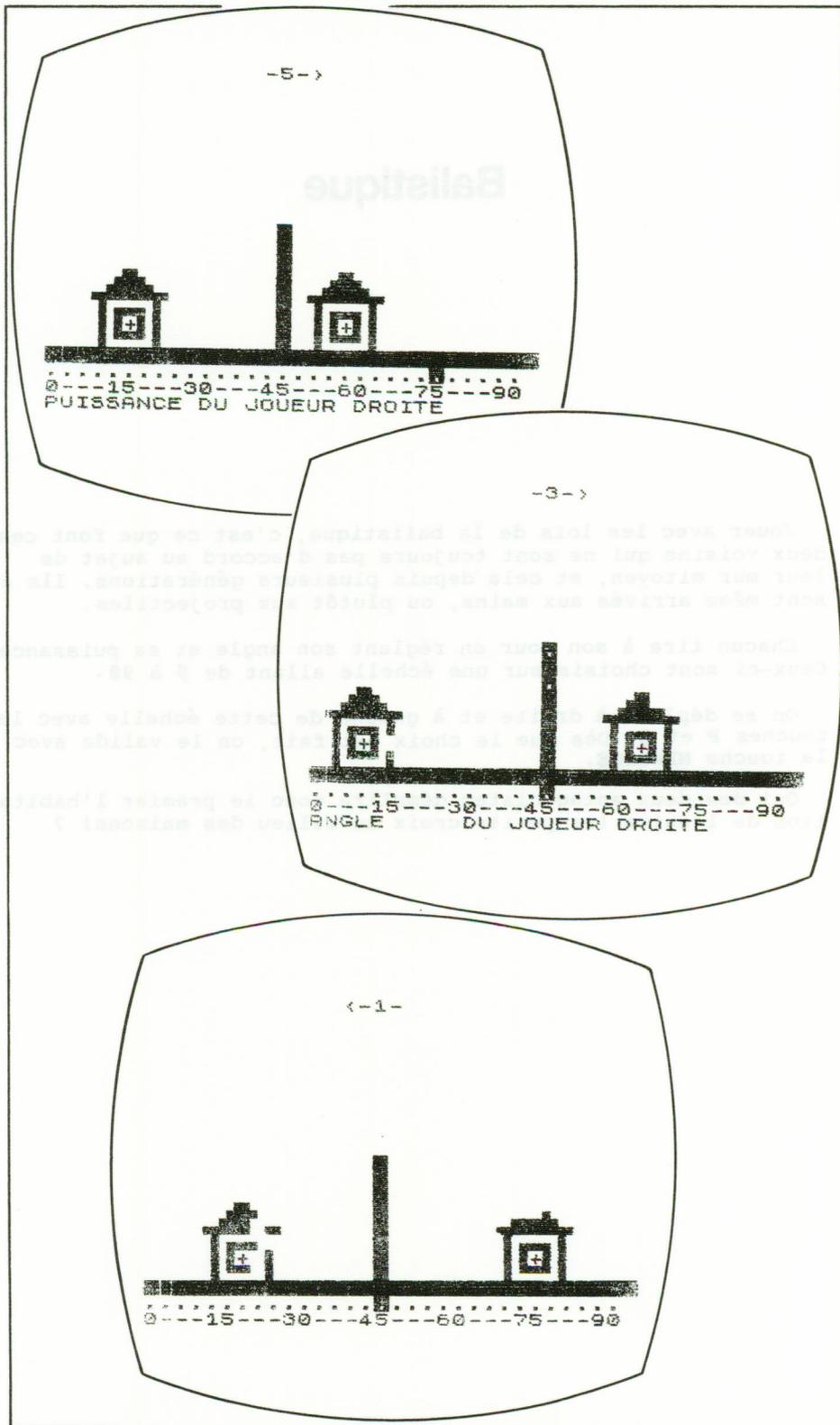
Jouer avec les lois de la balistique, c'est ce que font ces deux voisins qui ne sont toujours pas d'accord au sujet de leur mur mitoyen, et cela depuis plusieurs générations. Ils en sont même arrivés aux mains, ou plutôt aux projectiles.

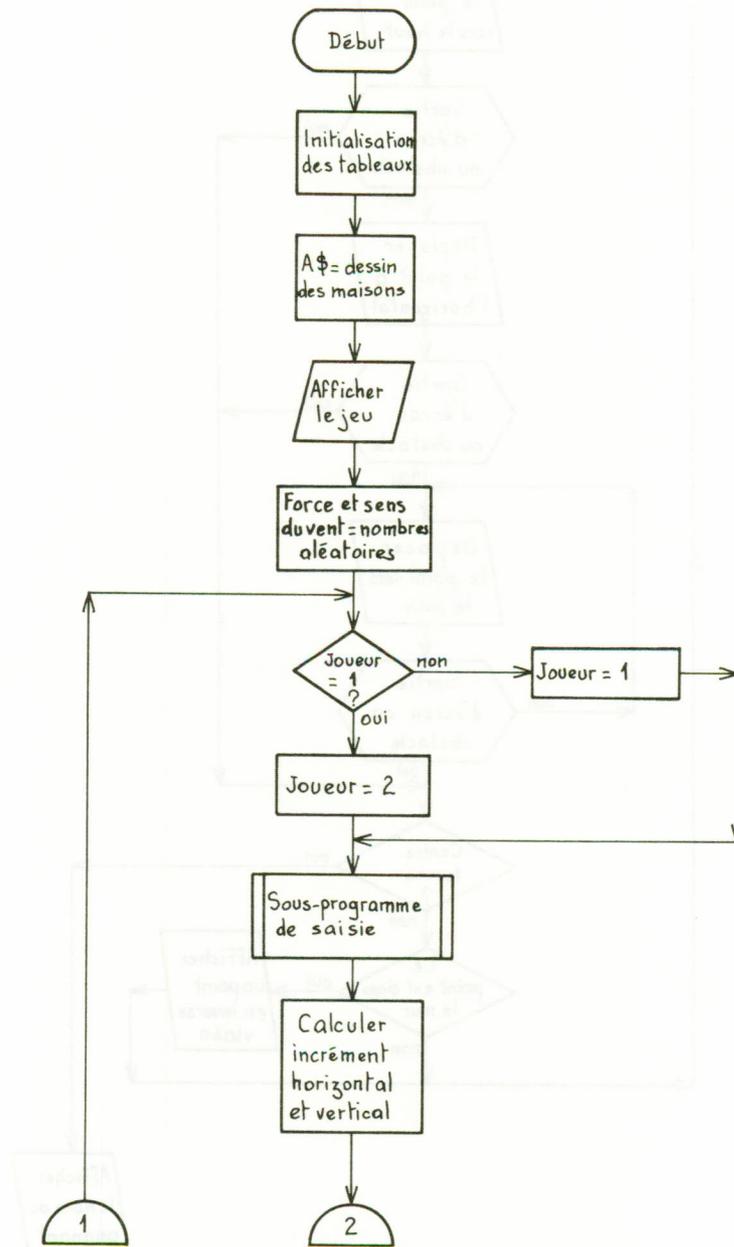
Chacun tire à son tour en réglant son angle et sa puissance. Ceux-ci sont choisis sur une échelle allant de 0 à 90.

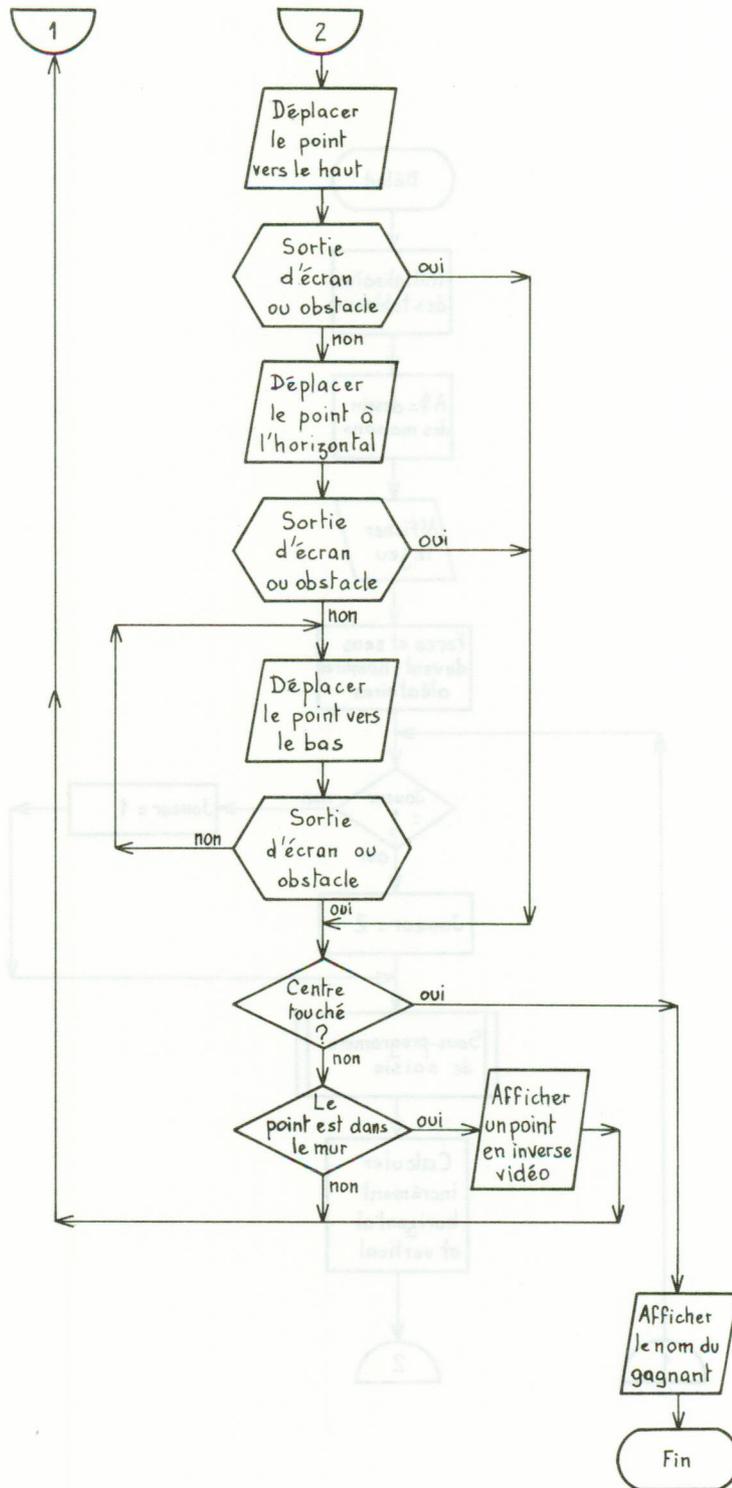
On se déplace à droite et à gauche de cette échelle avec les touches P et Q. Dès que le choix est fait, on le valide avec la touche **NEWLINE**.

Qui des deux antagonistes démolira donc le premier l'habitation de l'autre (la petite croix au milieu des maisons) ?









```

5 REM BALISTIQUE
10 REM -----
12 LET E=PEEK 16398+256*PEEK 1
6399
15 DIM A$(5,5)
17 DIM R(2)
20 DIM Y(2)
25 DIM X(2)
30 DIM S(2)
32 DIM P(2)
35 DIM J$(2,10)
40 LET J$(1)="GAUCHE"
45 LET J$(2)="DROITE"
50 DIM S$(2,10)
55 LET S$(1)="ANGLE"
60 LET S$(2)="PUISSANCE"
62 LET Y(1)=13
64 LET X(1)=6
66 LET S(1)=1
68 LET Y(2)=13
70 LET X(2)=17
72 LET S(2)=-1
85 DIM V(2)
90 LET J=1
95 DIM J(2)
100 FOR I=0 TO 31
110 PRINT AT 18,I;"■"
120 NEXT I
130 LET A$(1)="      "
140 LET A$(2)="      █"
150 LET A$(3)="      █ █"
160 LET A$(4)="      █ █ █"
170 LET A$(5)="      █ █ █ █"
172 LET P(1)=RND*8
175 LET P(2)=RND*8
180 FOR I=1 TO 5
190 PRINT AT 12+I,1+P(1);A$(I);
    AT 12+I,17+P(2);A$(I)
200 NEXT I
210 FOR I=1 TO 8
220 PRINT AT 9+I,15;"■"
230 NEXT I
250 LET ANGL=(PI/2)/90
280 REM -----
290 REM JEU
300 LET VV=INT (RND*5)+1
310 LET SV=INT (RND+.5)
330 IF SV=0 THEN PRINT AT 0,14;
    "-";VV;"->"
340 IF SV=0 THEN GOTO 360
350 PRINT AT 0,13;"<-";VV;"-"
360 LET V(1)=VV
370 LET V(2)=VV
380 IF SV=0 THEN LET V(2)=V(2)*
(-1)
390 IF SV=1 THEN LET V(1)=V(1)*
(-1)
400 IF J=1 THEN GOTO 430
410 LET J=1
420 GOTO 440
430 LET J=2
440 GOSUB 3000
450 LET R(1)=R(1)*3
460 LET IX=COS (R(1)*ANGL)
470 LET IY=SIN (R(1)*ANGL)
480 LET X=X(J)+P(J)
490 LET Y=Y(J)

```

```

600 FOR I=1 TO R(2)+V(J)
610 PRINT AT INT (Y),INT (X);"
"
620 LET X=X+(IX*S(J))
630 LET Y=Y-IY
640 IF PEEK (E+INT (X)+(INT (Y)
*33)>>>0 THEN GOTO 900
680 PRINT AT INT (Y),INT (X);"
"
690 NEXT I
700 FOR I=1 TO (R(2)-(V(J)*S(J)
)))/2
710 PRINT AT INT (Y),INT (X);"
"
720 LET X=X+(IX*S(J))
730 IF PEEK (E+INT (X)+(INT (Y)
*33)>>>0 THEN GOTO 900
780 PRINT AT INT (Y),INT (X);"
"
790 NEXT I
800 PRINT AT INT (Y),INT (X);"
"
820 LET X=X+(IX*S(J))
830 LET Y=Y+IY
840 IF PEEK (E+INT (X)+(INT (Y)
*33)>>>0 THEN GOTO 900
860 PRINT AT INT (Y),INT (X);"
"
870 GOTO 800
880 REM -----
890 REM FIN DE TIR
900 IF PEEK (E+INT (X)+(INT (Y)
*33)>=21 THEN GOTO 1000
910 IF X<1 OR X>31 OR Y<0 OR Y>
21 THEN GOTO 940
920 PRINT AT INT (Y),INT (X);"
"
930 IF INT (X)=15 AND Y>0 THEN
PRINT AT INT (Y),INT (X);"."
940 GOTO 400
1000 PRINT AT 21,0;"LE JOUEUR DE
";J*(J);" GAGNE"
1400 REM -----
1490 REM FIN DE PARTIE
1500 IF INKEY#="" THEN GOTO 1500
1510 CLS
1520 RUN
2900 REM -----
2990 REM SAISIE DES DONNEES
3000 FOR I=1 TO 2
3005 PRINT AT 19,0;".....
....."
3010 PRINT AT 20,0;"0---15---30-
---45---60---75---90"
3015 PRINT AT 21,0;S*(I);"DU JOU
EUR ";J*(J)
3020 LET Q=15
3030 PRINT AT 19,Q;"."
3040 IF INKEY#="" THEN GOTO 3040
3050 LET R#=INKEY#
3060 IF R#=CHR# 118 THEN GOTO 31
30
3070 PRINT AT 19,Q;"."
3080 IF R#="P" THEN LET Q=Q+1
3090 IF R#="Q" THEN LET Q=Q-1
3100 IF Q>30 THEN LET Q=30
3110 IF Q<0 THEN LET Q=0
3120 GOTO 3030
3130 LET R(I)=Q
3200 NEXT I
3210 PRINT AT 21,0;"
"
3220 RETURN

```

LE PROGRAMME**Lignes 12 à 250**

La variable **E** contient l'adresse du début d'écran ce qui permettra ultérieurement d'obtenir directement par des "PEEK" le code des caractères situés à des emplacements particuliers.

On assigne au tableau **A\$** le dessin des maisons. Les tableaux **X** et **Y** correspondent aux coordonnées de départ des projectiles des joueurs gauche et droite.

On affiche le jeu avec les deux maisons à des positions aléatoires pour "corser" le jeu. La variable **ANGL** permet de faire la conversion des données saisies en degrés pour les calculs en radians (sinus et cosinus).

Lignes 300 à 490

Comme pour la position des maisons, la vitesse et le sens du vent sont aléatoires.

Les calculs des incréments verticaux et horizontaux sont obtenus à partir de la variable **ANGL**.

Lignes 600 à 870

Le projectile est déplacé en trois temps : la montée (lignes 600 à 690), l'horizontale (lignes 700 à 790) et la descente (lignes 800 à 870). Celles-ci utilisent tous les paramètres déjà calculés, incréments, sens et vitesse du vent...

On teste à chaque déplacement qu'aucun obstacle ne se trouve sur son chemin.

Lignes 900 à 1520

Si la croix au milieu des maisons n'est pas touchée, on passe au joueur suivant. Sinon on affiche le gagnant.

Lignes 3000 à 3220 sous-programme

C'est la partie qui saisit l'angle et la puissance du tir. Ils sont placés dans le tableau **R** pour être exploités par le reste du programme au retour du sous-programme.

IDENTIFICATION des VARIABLES

NOM SYMBOLIQUE	TYPE (*)	ORIGINE				LONG.	OBSERVATIONS
		Clavier	Programme	Cassette	Disquette		
A ₁ N ₁ G L	CONSTANTE	X					Conversion degrés-radians.
A \$ ()	TABLEAU	X					Dessin des maisons.
E ₁	VARIABLE	X					Pointeur début d'écran.
I ₁	INDICE	X					Boucle FOR/NEXT.
I X ₁	VARIABLE	X					Incrément horizontal.
I Y ₁	VARIABLE	X					Incrément vertical.
J ₁	VARIABLE	X					Numéro du joueur.
J \$ ₁	TABLEAU	X					Noms des joueurs.
P ₁ (,) ₁	TABLEAU	X					Position des maisons.
Q ₁	VARIABLE	X					Pointeur sur l'échelle.
R ₁ (,) ₁	TABLEAU	X					Réponse angle et puissance.
R \$ ₁	VARIABLE	X					Réponse du joueur.
S V ₁	VARIABLE	X					Sens du vent.
S ₁ (,) ₁	TABLEAU	X					Sens du déplacement du tir.
S ₁ \$ (,) ₁	TABLEAU	X					Type des données à saisir.
V V ₁	VARIABLE	X					Vitesse du vent.
V ₁ (,) ₁	TABLEAU	X					Vitesse et sens du vent pour chaque joueur.
X ₁ () ₁	TABLEAU	X					Pointeur fin du tir.
Y ₁ () ₁	TABLEAU	X					Pointeur début du tir.

6/1980

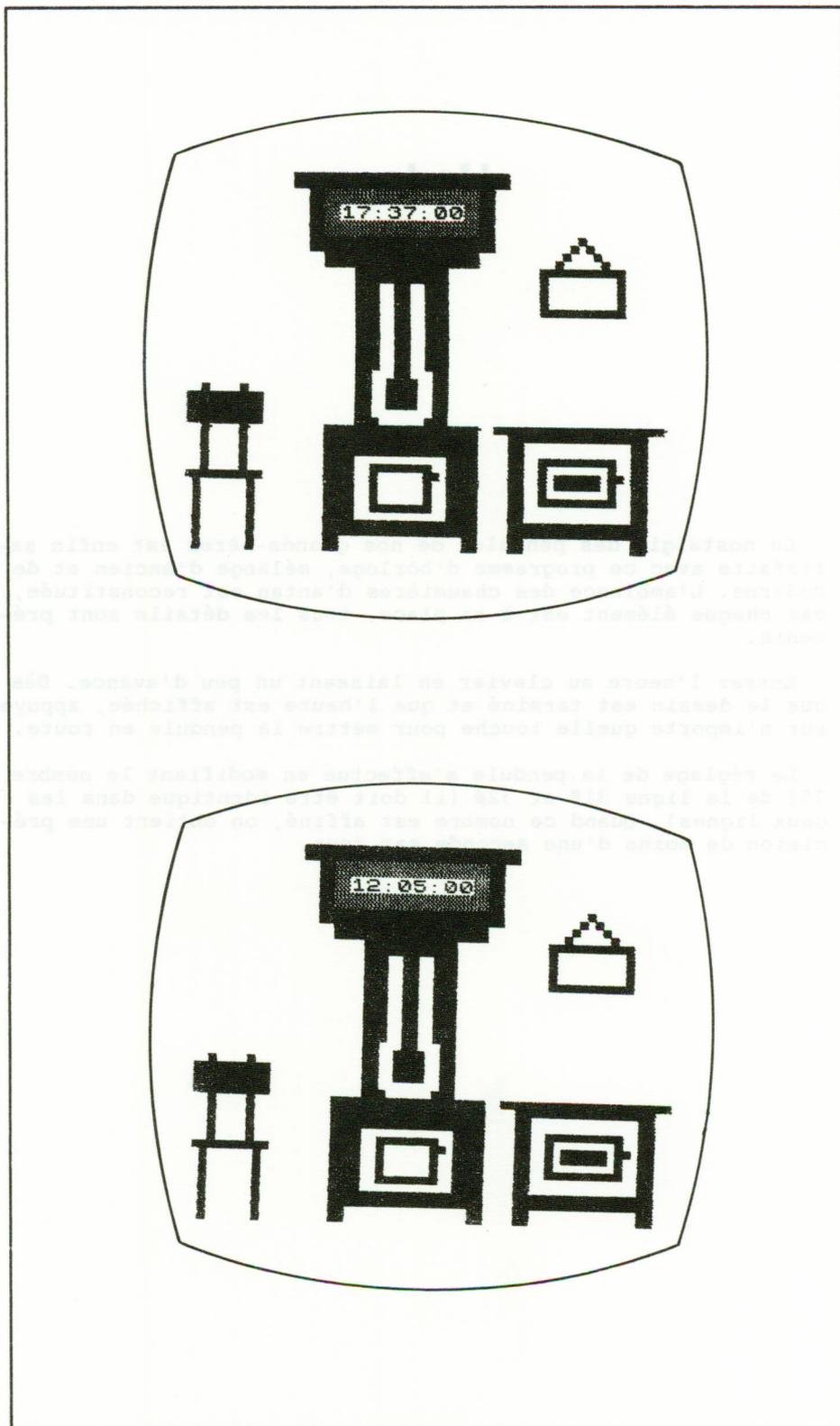
(*) Légende : aiguillage, compteur, constante, index, indice, tableau, variable. Reproduction interdite (c) Editions du P.S.I

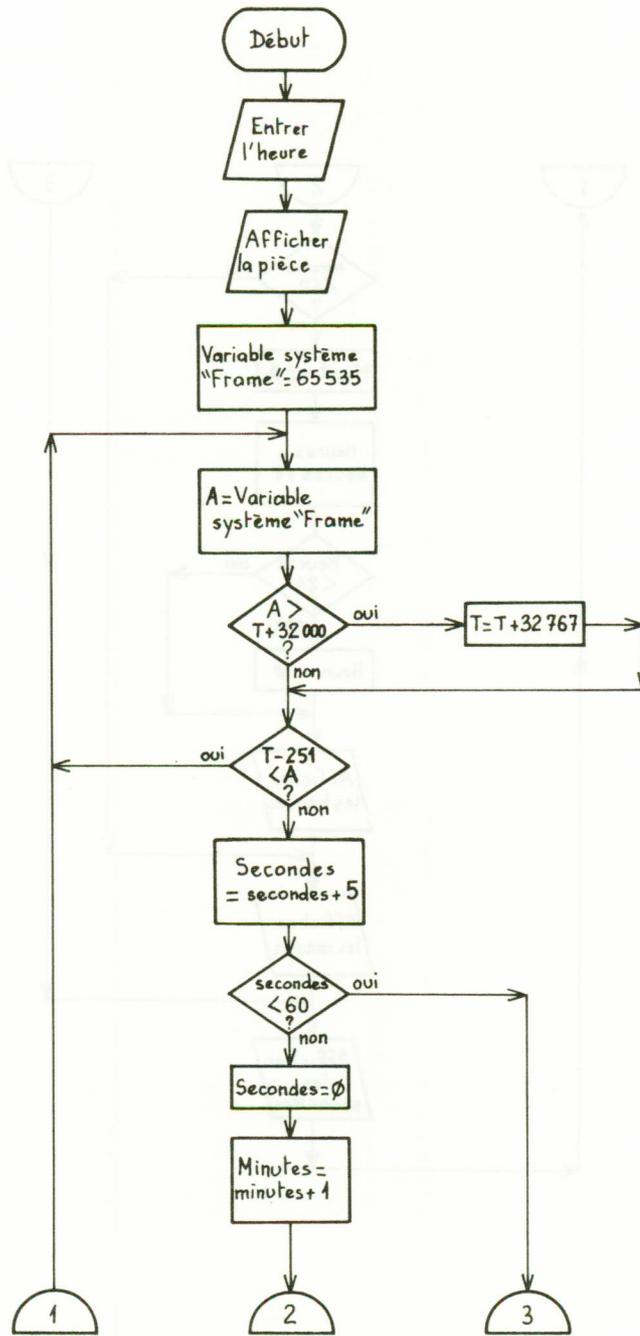
Horloge

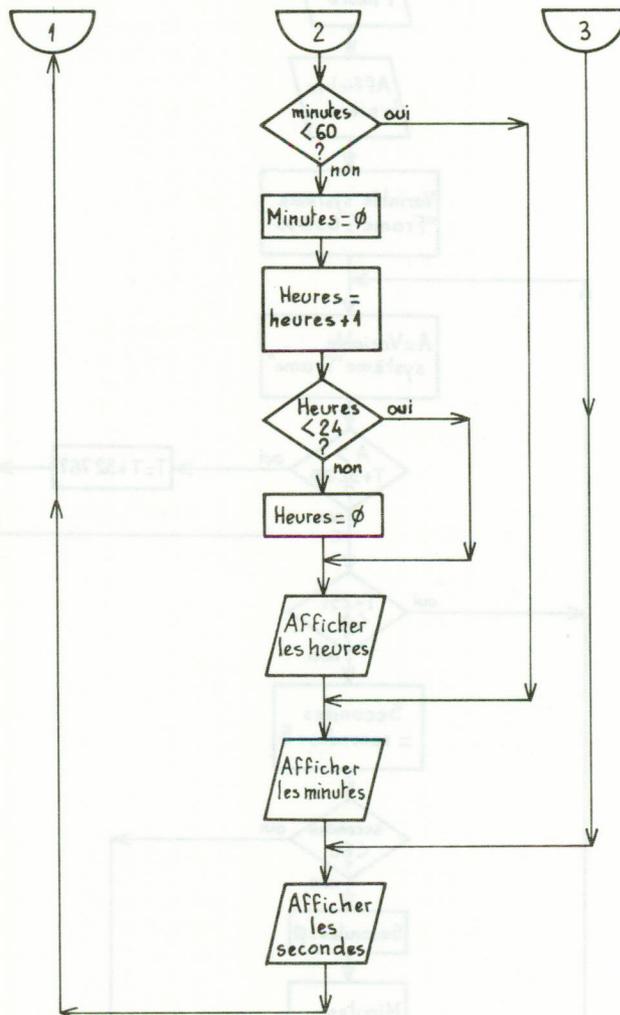
La nostalgie des pendules de nos grands-mères est enfin satisfaite avec ce programme d'horloge, mélange d'ancien et de moderne. L'ambiance des chaumières d'antan est reconstituée, car chaque élément est à sa place, tous les détails sont présents.

Entrez l'heure au clavier en laissant un peu d'avance. Dès que le dessin est terminé et que l'heure est affichée, appuyez sur n'importe quelle touche pour mettre la pendule en route.

Le réglage de la pendule s'effectue en modifiant le nombre 251 de la ligne 310 et 320 (il doit être identique dans les deux lignes). Quand ce nombre est affiné, on obtient une précision de moins d'une seconde par jour.







```

1980 REM -----
1990 REM AFFICHER LA PIECE
2000 PRINT TAB 8;"
"
2020 PRINT TAB 8;"
2030 PRINT TAB 8;"
2040 PRINT TAB 8;"
2050 PRINT TAB 8;"
2060 PRINT TAB 8;"
2070 PRINT TAB 8;"
2080 PRINT TAB 8;"
2090 PRINT TAB 8;"
2100 PRINT TAB 8;"
2110 PRINT TAB 8;"
2120 PRINT TAB 8;"
2130 PRINT TAB 8;"
2140 PRINT TAB 8;"
2150 PRINT TAB 8;"
2160 PRINT TAB 8;"
2170 PRINT TAB 8;"
2180 PRINT TAB 8;"
2190 PRINT TAB 8;"
2200 PRINT TAB 8;"
2210 PRINT TAB 8;"
2220 PRINT TAB 8;"
2230 POKE 16442,4
2240 PRINT "
"
2250 PRINT AT 13,0;"
2300 PRINT "
"
2310 PRINT "
"
2320 PRINT "
"
2330 PRINT "
"
2340 PRINT "
"
2350 PRINT "
"
2360 PRINT "
"
2370 PRINT "
"
2380 PRINT "
"
2390 PRINT AT 16,0;"
2400 PRINT TAB 21;"
2410 PRINT TAB 21;"
2420 PRINT TAB 21;"
2430 PRINT TAB 21;"
2440 PRINT TAB 21;"
2450 PRINT TAB 21;"
2460 PRINT AT 4,0;"
2470 PRINT TAB 24;"
2480 PRINT TAB 24;"
2490 PRINT TAB 24;"
2500 PRINT TAB 24;"
2510 PRINT TAB 24;"
2990 PRINT AT 2,11;H#
3000 RETURN

```

```

10 REM HORLOGE
20 REM -----
30 LET T=65535
40 GOSUB 1000
100 IF INKEY#="" THEN GOTO 100
200 POKE 16436,255
210 POKE 16437,255
300 LET A=PEEK 16436+256*PEEK 1
6437
305 IF A>T+32000 THEN LET T=327
67+T
310 IF T-251<A THEN GOTO 300
320 LET T=T-251
400 LET S=S+5
410 IF S<60 THEN GOTO 600
420 LET S=0
430 LET M=M+1
440 IF M<60 THEN GOTO 640
450 LET M=0
460 LET H=H+1
470 IF H<24 THEN GOTO 600
480 LET H=0
580 REM -----
590 REM AFFICHER L HORLOGE
600 IF H<10 THEN GOTO 630
610 PRINT AT 2,11;H
620 GOTO 640
630 PRINT AT 2,11;"0";H
640 IF M<10 THEN GOTO 670
650 PRINT AT 2,14;M
660 GOTO 600
670 PRINT AT 2,14;"0";M
680 IF S<10 THEN GOTO 710
690 PRINT AT 2,17;S
700 GOTO 300
710 PRINT AT 2,17;"0";S
720 GOTO 300
980 REM -----
990 REM ENTREZ L HEURE
1000 PRINT AT 21,0;"HEURE SOUS L
A FORME HH:MM ?"
1010 INPUT H#
1020 LET H#=H#+":00"
1030 LET H=VAL H#( TO 2)
1040 LET M=VAL H#(4 TO 5)
1050 LET S=VAL H#(7 TO )
1060 CLS

```

LE PROGRAMME**Lignes 30 à 48**

La variable Système FRAME est décrémentée à chaque rafraîchissement de l'image, soit cinquante fois par seconde.

Chaque fois que cette variable est décrémentée de 250 fois, on ajoute 5 secondes à l'horloge, puis une minute quand on arrive à 60 secondes et ainsi de suite (voir organigramme).

Lignes 590 à 720

On affiche à l'écran les heures, minutes et secondes en ayant soin de placer ces valeurs sur deux caractères, dont un zéro devant les valeurs inférieures à dix.

Lignes 1000 à 1060 sous-programme

On découpe la variable H\$ en deux parties pour les assigner aux heures (variable H) et aux minutes (variable M).

Lignes 2000 à 3000

Cette partie du programme affiche à l'écran les aspects d'une pièce à l'ancienne ! Celle-ci peut bien sûr être imaginée autrement par le lecteur.

A noter que le "POKE 16442,4" permet d'écrire sur les deux dernières lignes de l'écran, ce qui porte à 24 le nombre de lignes utilisables.

Cuisine

Si vous êtes gourmand(e) ou si vous manquez d'idées de recettes, ce programme est pour vous.

Pas compliqué ! Vous tapez RUN et hop! en " trois coups de cuiller à pot" une recette de gâteau au chocolat, aux noix ou aux raisins s'affiche à l'écran. Il ne reste plus qu'à réunir tous les ingrédients. La marche à suivre (très simple) est donnée après la liste des composants. Le programme vous conseille même la boisson la plus appropriée à la dégustation de votre oeuvre.

Bon appétit !

Remarque : Avant de vous lancer dans la préparation, vérifiez toutefois que les ingrédients entrant dans la composition du gâteau soient cohérents ! Dans tous les cas, il suffit d'appuyer sur n'importe quelle touche pour obtenir une autre recette.

RECETTE

FARINE : 100 GRAMMES
 SUCRE : 50 GRAMMES
 EAU : 20 CLS
 CERISES : 200 GRAMMES
 SEL : 1 PINCEE (S)
 LEVURE : UN DEMI-SACHET

MELANGER FARINE, SUCRE ET EAU POUR OBTENIR LA PATE. AJOUTER CERISES, SEL ET LEVURE. BEURRER UN MOULE ET Y METTRE LA PATE OBTENUE. PLACER CE MOULE DANS UN FOUR A 200 DEGRES. LAISSER CUIRE 45 MINUTES. DEGUSTER CE GATEAU AVEC DU VIN BLANC.

RECETTE

FARINE : 300 GRAMMES
 SUCRE : 50 GRAMMES
 LAIT : 20 CLS
 EAU : 20 CLS
 POMMES : 400 GRAMMES
 AMANDES : 50 GRAMMES
 COGNAC : UN DEMI-VERRE
 VANILLE : 1 SACHET
 FRUITS C. : 50 GRAMMES

MELANGER FARINE, SUCRE, LAIT ET EAU POUR OBTENIR LA PATE. AJOUTER POMMES, AMANDES, COGNAC, VANILLE ET FRUITS C. BEURRER UN MOULE ET Y METTRE LA PATE OBTENUE. PLACER CE MOULE DANS UN FOUR A 140 DEGRES. LAISSER CUIRE 35 MINUTES. DEGUSTER CE GATEAU AVEC DU JUS D'ORANGE.

RECETTE

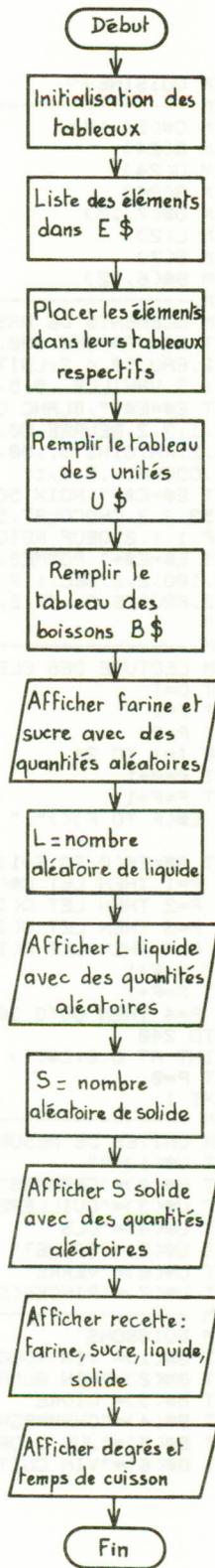
FARINE : 100 GRAMMES
 SUCRE : 50 GRAMMES
 OEUF : 3
 COGNAC : UN DEMI-VERRE
 SEL : 2 PINCEE
 RAISINS S. : 100 GRAMMES

MELANGER FARINE, SUCRE ET OEUF POUR OBTENIR LA PATE. AJOUTER COGNAC, SEL, OEUF ET RAISINS S. BEURRER UN MOULE ET Y METTRE LA PATE OBTENUE. PLACER CE MOULE DANS UN FOUR A 210 DEGRES. LAISSER CUIRE 30 MINUTES. DEGUSTER CE GATEAU AVEC DU VIN CUIT.

RECETTE

FARINE : 200 GRAMMES
 SUCRE : 100 GRAMMES
 EAU : 10 CLS
 LAIT : 10 CLS
 AMANDES : 50 GRAMMES
 BEURRE : 50 GRAMMES
 RHUM : UN DEMI-VERRE

MELANGER FARINE, SUCRE, EAU ET LAIT POUR OBTENIR LA PATE. AJOUTER AMANDES, BEURRE ET RHUM. BEURRER UN MOULE ET Y METTRE LA PATE OBTENUE. PLACER CE MOULE DANS UN FOUR A 190 DEGRES. LAISSER CUIRE 55 MINUTES. DEGUSTER CE GATEAU AVEC DU VIN BLANC.



```

10 REM CUISINE
20 REM -----
40 DIM C$(24,10)
50 DIM Q(24)
60 DIM U(24)
70 DIM A(24)
80 DIM U$(7,10)
90 DIM L(2)
100 DIM S(7)
110 DIM B$(6,12)
130 REM -----
140 REM ELEMENTS DE BASE
150 LET E#="FARINE,100,2,3,SUCR
E,50,2,2,EAU,10,4,2,LAIT,10,4,2,
OEUF,1,1,3,VANILLE,.5,5,2"
160 LET E#=E#+",BLANC OEUF,1,1,
3,HUILE,1,3,2,BEURRE,50,2,2,LEVU
RE,.5,5,2,RAISINS S.,50,2,2,RHUM
,.5,6,1,COGNAC,.5,6,1"
170 LET E#=E#+",NOIX,50,2,2,NOI
SETTES,50,2,2,CHOCOLAT,50,2,2,JA
UNE OEUF,1,1,2,OEUF NEIGE,1,1,3"
180 LET E#=E#+",POMMES,200,2,2,
CERISES,100,2,2,SEL,1,7,2,AMANDE
S,50,2,2,FRUITS C.,50,2,2,CREME,
25,2,2,"
185 REM -----
190 REM LECTURE DES ELEMENTS
200 LET D=1
210 LET F=1
220 LET P=0
230 FOR I=1 TO 24
240 LET P=P+1
250 LET F=F+1
260 IF E$(F TO F)<>"," THEN GOT
O 250
270 LET A#=(D TO F-1)
280 IF P=1 THEN LET C$(I)=A#
290 IF P=2 THEN LET Q(I)=VAL A#
300 IF P=3 THEN LET U(I)=VAL A#
310 IF P=4 THEN LET A(I)=VAL A#
320 LET D=F+1
330 LET F=F+1
340 IF P=4 THEN GOTO 360
350 GOTO 240
360 PRINT AT 0,0;C$(I)
370 LET P=0
380 NEXT I
385 REM -----
390 REM UNITES DE MESURE
400 LET U$(1)=" "
410 LET U$(2)="GRAMMES"
420 LET U$(3)="CUILLERE(S)"
430 LET U$(4)="CLS"
440 LET U$(5)="SACHET"
450 LET U$(6)="VERRE"
460 LET U$(7)="PINCEE(S)"
480 REM -----
490 REM BOISSONS
500 LET B$(1)="VIN ROUGE"
510 LET B$(2)="VIN BLANC"
520 LET B$(3)="CIDRE"
530 LET B$(4)="CHAMPAGNE"
540 LET B$(5)="JUS D ORANGE"
550 LET B$(6)="VIN CUIT"

```

ETUDES POUR ZX81

```

580 REM -----
590 REM INGREDIENTS
600 CLS
610 PRINT TAB 10;"RECETTE"
620 PRINT
660 REM -----
670 REM ELEMENTS DE LA PATE
680 LET NE=1
690 GOSUB 1500
700 LET NE=2
710 GOSUB 1500
720 LET L=INT (RND*2)+1
730 FOR I=1 TO L
740 LET NE=INT (RND*3)+3
750 IF I=1 THEN GOTO 770
760 IF NE=L(1) THEN GOTO 740
770 GOSUB 1500
780 LET L(I)=NE
790 NEXT I
880 REM -----
890 REM ELEMENTS DU GATEAU
900 LET S=INT (RND*3)+3
910 FOR I=1 TO S
920 LET NE=INT (RND*19)+6
930 IF I=1 THEN GOTO 970
940 FOR J=1 TO I-1
950 IF NE=S(J) THEN GOTO 920
960 NEXT J
970 GOSUB 1500
980 LET S(I)=NE
990 NEXT I
1000 PRINT
1080 REM -----
1090 REM RECETTE
1100 PRINT "MELANGER FARINE, SUC
RE";
1110 IF L=1 THEN GOTO 1140
1120 PRINT ", ";C$(L(1));" ET ";
C$(L(2));
1130 GOTO 1150
1140 PRINT " ET ";C$(L(1));
1150 PRINT "POUR OBTENIR LA PATE
.";
1160 PRINT "AJOUTER ";C$(S(1));
1170 FOR I=2 TO S-1
1180 PRINT ", ";C$(S(I));
1190 NEXT I
1200 PRINT " ET ";C$(S(S));". ";
1210 PRINT "BEURRER UN MOULE ET
Y METTRE LA PATE OBTENUE. ";
1220 PRINT "PLACER CE MOULE DANS
UN FOUR A ";100+(10*INT (RND*16
));" DEGRES. ";
1230 PRINT "LAISSER CUIRE ";30+(
5*INT (RND*7));" MINUTES."
1240 PRINT " DEGUSTER CE GATEAU
AVEC DU ";B$(INT (1+RND*6));
"."
1390 REM -----
1400 REM FIN
1410 IF INKEY#="" THEN GOTO 1410
1420 GOTO 600
1480 REM -----
1490 REM CALCUL DES INGREDIENTS
1500 LET Q=Q(NE)*INT (RND*(NE)+
1)
1510 IF Q=.5 THEN GOTO 1540
1520 PRINT C$(NE);": ";Q;" ";U$(U
(NE))
1530 RETURN
1540 PRINT C$(NE);": UN DEMI-";U$(
U(NE))
1550 RETURN

```

LE PROGRAMME**Lignes 40 à 110**

On déclare dans ces lignes tous les tableaux. Le rôle de chaque tableau est expliqué dans la liste des variables.

Lignes 150 à 180

On place dans la variable E\$ la liste des éléments constituant les gâteaux. On trouve à la suite le nom du composant, son poids ou sa quantité de base, le numéro de l'unité de mesure (voir lignes 400 à 460), suivi du nombre maximum de bases. Ces éléments sont placés dans la variable E\$ sur plusieurs lignes pour que les corrections lors de la frappe soient plus faciles.

Lignes 200 à 380

Tous les éléments de E\$ sont placés dans leurs tableaux respectifs.

Lignes 400 à 460

On remplit le tableau U\$ avec toutes les unités de mesure nécessaires aux éléments. U\$(1) est une chaîne vide puisqu'elle correspond à une quantité.

Lignes 500 à 550

Le tableau B\$ contient la liste des boissons.

Lignes 600 à 790

La farine et le sucre sont les deux éléments indispensables à toute pâte. On appelle donc toujours le sous-programme à la ligne 1500 pour ces deux éléments.

On ajoute pour compléter cette pâte un des deux éléments liquides. Le nombre de liquides est déterminé par la variable L. Dans le cas où deux éléments liquides sont utilisés, on vérifie qu'il ne s'agit pas deux fois du même.

Lignes 900 à 1000

Puis c'est au tour des éléments solides. On utilise le même procédé que pour les liquides. Chaque solide est placé dans le tableau S pour la vérification des doublons. Ce tableau sera aussi utilisé pour l'édition du texte de la recette (lignes 1150 à 2000).

Lignes 1100 à 1240

Chaque élément liquide et solide est repris dans le texte de la recette. Deux nombres aléatoires donnent la température et le temps de cuisson du gâteau. Puis on affiche un des éléments du tableau des boissons.

Lignes 1500 à 1550 (sous-programme)

La variable NE correspond au numéro de l'élément à éditer.
 La quantité à utiliser est comprise entre une base (tableau Q)
 et cette base multipliée par le nombre aléatoire maximum.
 (tableau A). Dans le cas où l'on obtient 0.5, la ligne 1540
 affiche le message " UN DEMI".

TYPE	DESCRIPTION	TYPE	DESCRIPTION
TAB	Liste des codes aléatoires	X	
VAR	Partie de la chaîne LI	X	
TAB	Liste des boisons	X	
TAB	Liste des composants	X	
COM	Pointeur au début de liste	X	
VAR	Liste des éléments	X	
COM	Pointeur de fin de liste	X	
IND	Boucle FORNEXT	X	
IND	Boucle FORNEXT	X	
VAR	Nombre de listons	X	
TAB	Liste des éléments listons	X	
VAR	Numéro d'élément à éditer	X	
COM	Pointeur d'un des quatre listons	X	
VAR	composant, quantité, unité, nombre aléatoire		
VAR	Quantité à utiliser	X	
TAB	Liste des quantités de base de chaque composant	X	
VAR	Nombre de solides	X	
TAB	Liste des éléments solides	X	
TAB	Liste des unités	X	
TAB	Noms des unités	X	

IDENTIFICATION des VARIABLES

NOM SYMBOLIQUE	TYPE (*)	ORIGINE				LONG.	OBSERVATIONS
		Clavier	Programme	Cassette	Disquette		
A, (,)	TABEAU	X					Liste des codes aléatoires
A, \$	VARIABLE	X					Partie de la chaîne E\$
B, \$ (,)	TABEAU	X					Liste des boissons
C, \$ (,)	TABEAU	X					Liste des composants
D	COMPTEUR	X					Pointeur du début de liste
E \$	VARIABLE	X					Liste des éléments
F	COMPTEUR	X					Pointeur de fin de liste
I	INDICE	X					Boucle FOR/NEXT
J	INDICE	X					Boucle FOR/NEXT
L	VARIABLE	X					Nombre de liquides
L (,)	TABEAU	X					Liste des éléments liquides
N E	VARIABLE	X					Numéro d'élément à éditer
P	COMPTEUR	X					Pointeur d'un des quatre tableaux
							composants, quantité, unité, nombre aléatoire
Q	VARIABLE	X					Quantité à utiliser
Q (,)	TABEAU	X					Liste des quantités de base de chaque composant
S	VARIABLE	X					Nombre de solides
S (,)	TABEAU	X					Liste des éléments solides
U, (,)	TABEAU	X					Liste des unités
U \$ (,)	TABEAU	X					Noms des unités

6/1980

(*) Légende : aiguillage, compteur, constante, index, indice, tableau, variable. Reproduction interdite (c) Editions du P.S

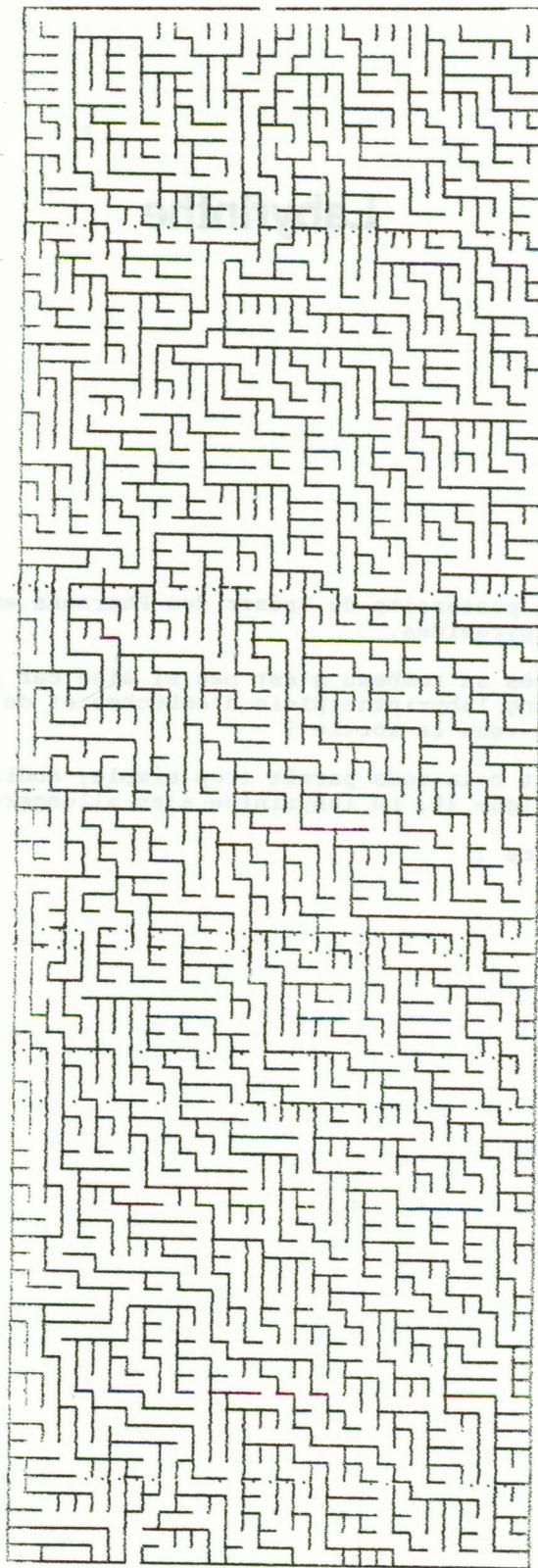
Labyrinthe

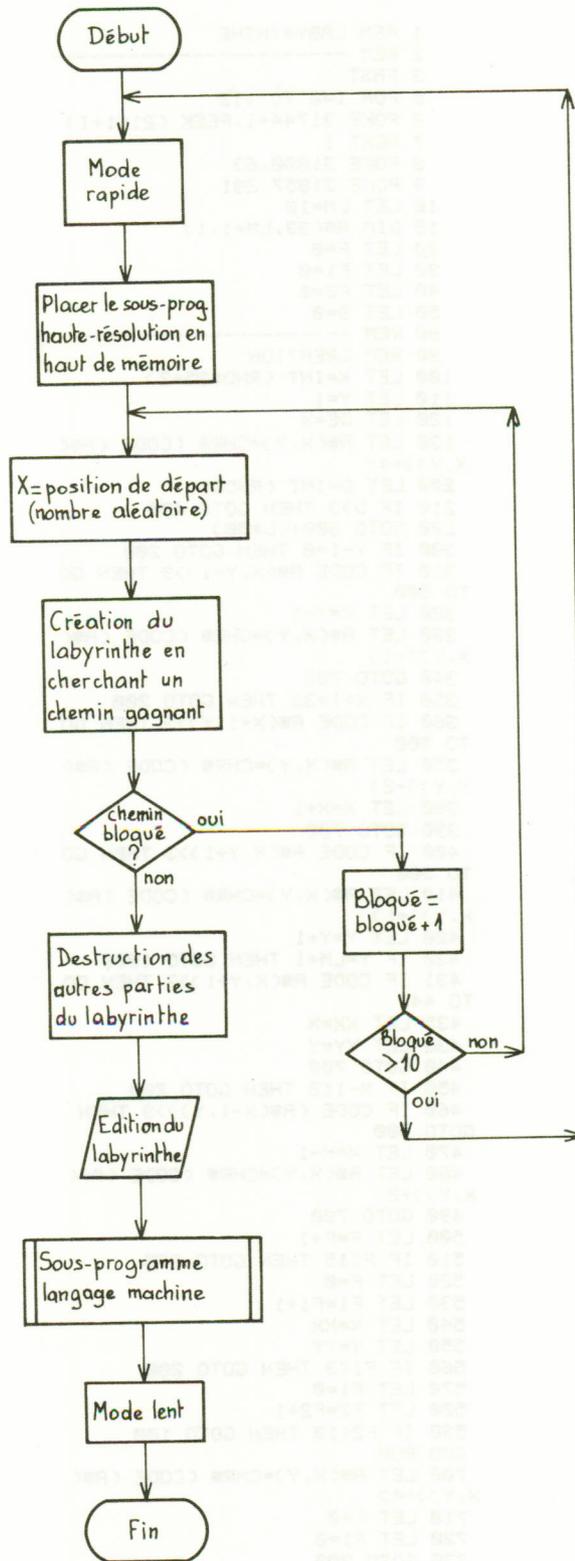
Rentrez en possession du Trésor des Pharaons en explorant les grandes pyramides.

Mais l'accès au tombeau n'est pas si aisé car il faut traverser un long labyrinthe plein d'embûches et de culs-de-sac. En trouverez-vous la sortie ?

Si vraiment cela vous paraît trop simple, modifiez la valeur de **LM** à la ligne 10. Le labyrinthe s'en allongera d'autant.

Bonne chance !





```

1 REM LABYRINTHE
2 REM -----
3 FAST
4 FOR I=0 TO 112
5 POKE 31744+I,PEEK (2161+I)
6 NEXT I
7 POKE 31800,63
8 POKE 31857,201
9 LET LM=10
10 DIM A$(33,LM+1,1)
11 LET F=0
12 LET F1=0
13 LET F2=0
14 LET B=0
15 REM -----
16 REM CREATION
17 LET X=INT (RND*25+2)
18 LET Y=1
19 LET DE=X
20 LET A$(X,Y)=CHR$ (CODE (A$(X,Y))+4)
21 LET D=INT (RND*5)
22 IF D>3 THEN GOTO 400
23 GOTO 300+(D*50)
24 IF Y-1=0 THEN GOTO 200
25 IF CODE A$(X,Y-1)>3 THEN GO
TO 500
26 LET Y=Y-1
27 LET A$(X,Y)=CHR$ (CODE (A$(X,Y))+1)
28 GOTO 700
29 IF X+1=32 THEN GOTO 200
30 IF CODE A$(X+1,Y)>3 THEN GO
TO 500
31 LET A$(X,Y)=CHR$ (CODE (A$(X,Y))+2)
32 LET X=X+1
33 GOTO 700
34 IF CODE A$(X,Y+1)>3 THEN GO
TO 500
35 LET A$(X,Y)=CHR$ (CODE (A$(X,Y))+1)
36 LET Y=Y+1
37 IF Y=LM+1 THEN GOTO 1000
38 IF CODE A$(X,Y+1)>3 THEN GO
TO 440
39 LET XX=X
40 LET YY=Y
41 GOTO 700
42 IF X-1<2 THEN GOTO 200
43 IF CODE (A$(X-1,Y))>3 THEN
GOTO 500
44 LET X=X-1
45 LET A$(X,Y)=CHR$ (CODE (A$(X,Y))+2)
46 GOTO 700
47 LET F=F+1
48 IF F<15 THEN GOTO 200
49 LET F=0
50 LET F1=F1+1
51 LET X=XX
52 LET Y=YY
53 IF F1<3 THEN GOTO 200
54 LET F1=0
55 LET F2=F2+1
56 IF F2<10 THEN GOTO 100
57 RUN
58 LET A$(X,Y)=CHR$ (CODE (A$(X,Y))+4)
59 LET F=0
60 LET F1=0
61 GOTO 200

```

```

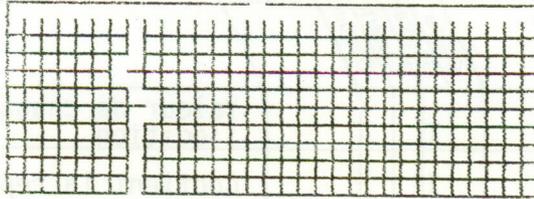
980 REM -----
990 REM DESTRUCTION
1000 LET DA=X
1010 FOR J=1 TO LM
1020 LET Z=INT (RND*5)
1030 FOR I=2 TO 32
1040 IF CODE A$(I,J)>3 AND Z<>2
THEN GOTO 1200
1050 LET A=INT (RND*2)
1060 GOSUB 1100+A*40
1070 IF B=0 AND I<>32 AND J<>LM
THEN GOSUB 1140-A*40
1080 LET B=0
1090 GOTO 1200
1100 IF J=LM OR CODE A$(I,J+1)>3
OR CODE A$(I,J)=1 THEN RETURN
1110 LET A$(I,J)=CHR$(CODE (A$(I,J))>+1)
1120 LET B=1
1130 RETURN
1140 IF I=32 OR CODE A$(I+1,J)>3
OR CODE A$(I,J)=2 THEN RETURN
1150 LET A$(I,J)=CHR$(CODE (A$(I,J))>+2)
1160 LET B=1
1170 RETURN
1200 NEXT I
1210 NEXT J
1980 REM -----
1990 REM EDITION
2000 FOR I=0 TO LM
2010 IF I>0 THEN GOTO 3010
2020 FOR J=0 TO 31
2030 FOR K=1 TO 8
2040 IF J=0 OR (J=31 AND K>1) TH
EN GOTO 2090
2050 IF J=15 THEN GOTO 2070
2060 IF K=1 THEN GOTO 2110
2070 POKE 32255+K+8*(J),0
2080 GOTO 2120
2090 POKE 32255+K+8*(J),1
2100 GOTO 2120
2110 POKE 32255+K+8*(J),255
2120 NEXT K
2130 NEXT J
2140 GOTO 3460
3010 FOR J=1 TO 32
3020 LET A=CODE A$(J,I)
3030 IF J=1 THEN LET A=1
3040 IF A>3 THEN LET A=A-4
3050 IF A=3 THEN GOTO 3170
3060 IF A<>0 AND A<>1 THEN GOTO
3100
3070 FOR K=1 TO 8
3080 POKE 32255+K+8*(J-1),1
3090 NEXT K
3100 IF A<>0 AND A<>2 THEN GOTO
3450
3110 POKE 32255+8+8*(J-1),255
3120 IF A=0 THEN GOTO 3450
3130 FOR K=1 TO 7
3140 POKE 32255+K+8*(J-1),0
3150 NEXT K
3160 GOTO 3450
3170 FOR K=1 TO 8
3180 POKE 32255+K+8*(J-1),0
3190 NEXT K
3450 NEXT J
3460 FOR H=0 TO 31
3470 POKE 16444+H,H
3480 NEXT H
3490 LET HPRINT=USR 31744
3500 NEXT I
4000 SLOW

```

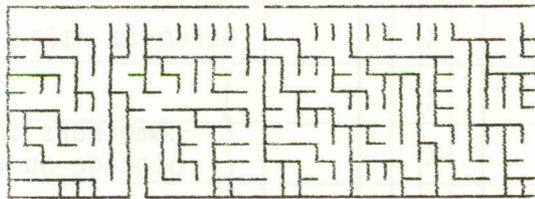

DESTRUCTION

Lignes 1000 à 1210

Si l'on éditait tout de suite le labyrinthe, on obtiendrait quelque chose de ce genre :



Il faut donc détruire toutes les autres cases en supprimant soit la barre verticale, soit la barre horizontale. Dans le cas des cases adjacentes au chemin gagnant, on supprime une de ces barres seulement si la variable Z est égale à 2, ce qui ajoute des "culs-de-sac".



Labyrinthe de l'exemple précédent après destruction

EDITION

Lignes 2000 à 4000

Le programme d'édition utilise un sous-programme en langage machine pour une impression en haute-résolution. On recopie donc le programme écrit dans la mémoire morte du ZX81, mais en modifiant l'adresse du pointeur de la liste de caractères à imprimer. Ce programme est placé en haut de la mémoire à l'adresse 31744.

DATE		NOM DU PROGRAMME : LABYRINTHE				PAGE 1 / 1	
IDENTIFICATION des VARIABLES							
NOM SYMBOLIQUE	TYPE (*)	ORIGINE				LONG.	OBSERVATIONS
		Clavier	Programme	Cassette	Disquette		
A ₁	VARIABLE	X					Valeur de chaque case pour l'édition
A \$ () 	TABLERAU	X					Contient la représentation du labyrinthe
B ₁	AIGUILLAGE	X					B = 1 si l'on a détruit une partie de case
D A ₁	VARIABLE	X					Garde la dernière valeur de X
D E ₁	VARIABLE	X					Garde la valeur de départ de X
F ₁	COMPTEUR	X					} Compteurs indiquant le blocage du chemin gagnant
F 1 ₁	COMPTEUR	X					
F 2 ₁	COMPTEUR	X					
H ₁	INDICE	X					Boucle FOR/NEXT
H P R I N T 	VARIABLE	X					Utilisée pour l'appel du sous-programme en langage machine
I ₁	INDICE	X					Boucle FOR/NEXT
J ₁	INDICE	X					Boucle FOR/NEXT
K ₁	INDICE	X					Boucle FOR/NEXT
L M ₁	COMPTEUR	X					Longueur maximum du labyrinthe
X ₁	VARIABLE	X					Numéro de colonne. Nombre aléatoire au début du labyrinthe
Y ₁	VARIABLE	X					Numéro de ligne
Z ₁	AIGUILLAGE	X					Si Z = 2, on détruit une case du chemin gagnant.

6/1980

(*) Légende : aiguillage, compteur, constante, index, indice, tableau, variable. Reproduction interdite (c) Editions du P.S

Morse

Devenez radio-amateur. C'est un hobby passionnant !

Mais pour cela, il faut passer un examen de télégraphie. Ne perdez pas de temps, ajoutez vite une carte son à votre ZX et devenez un "bon pianiste" !

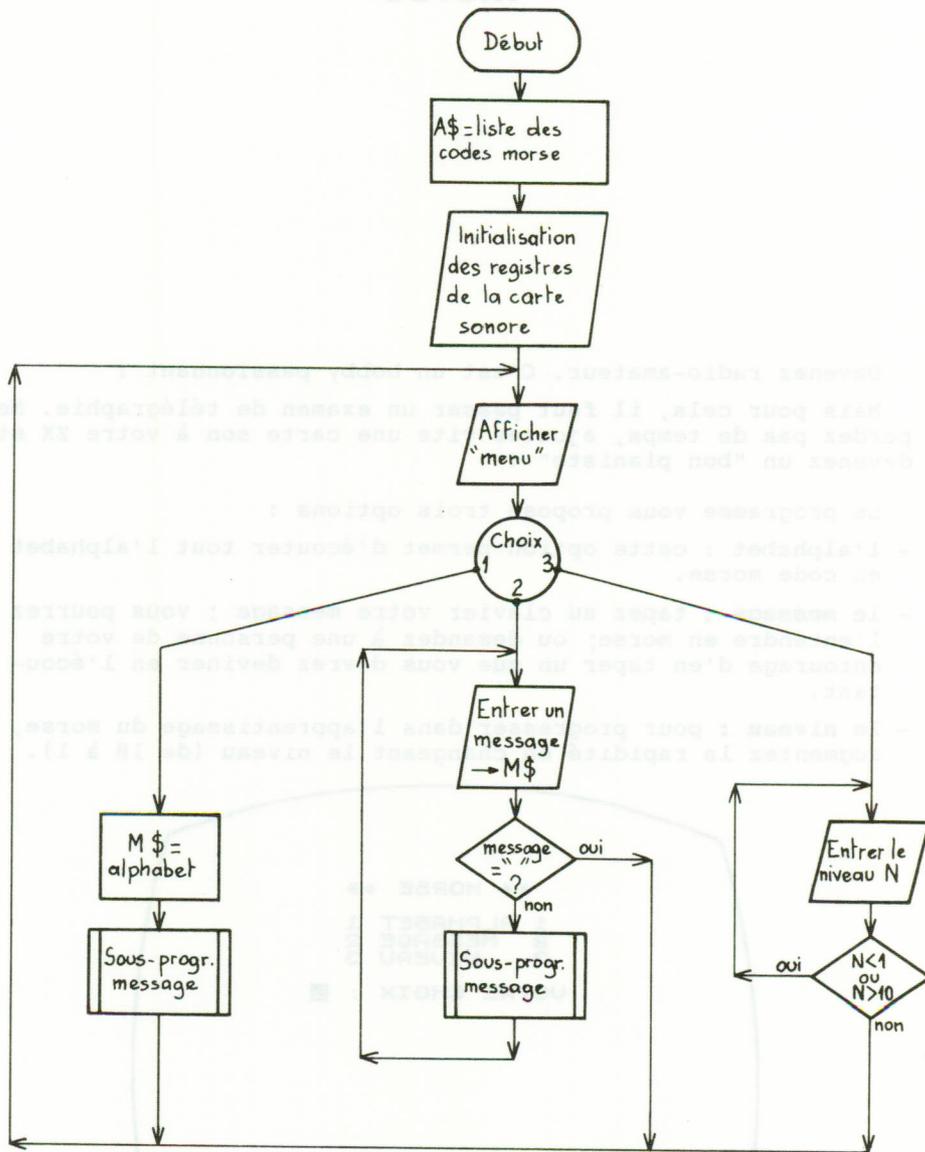
Le programme vous propose trois options :

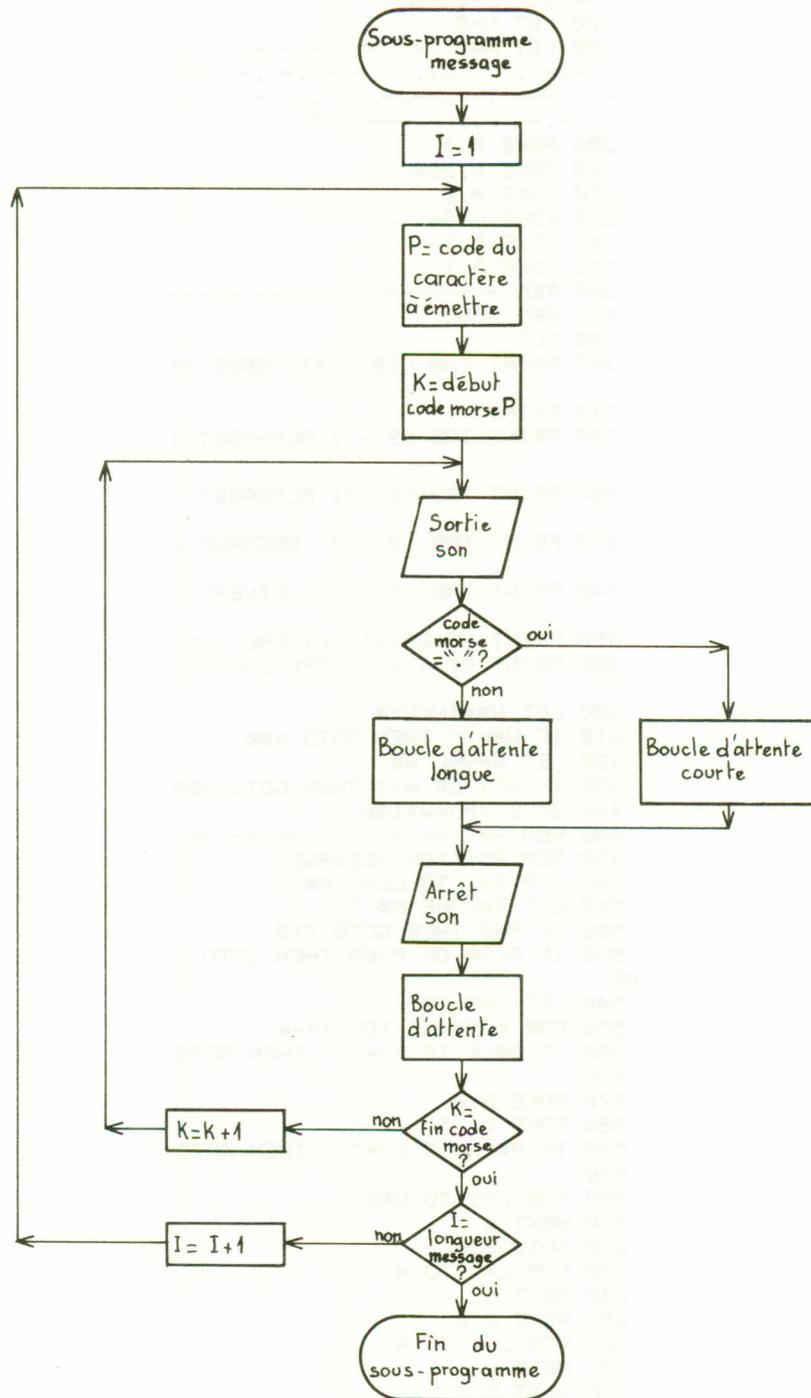
- **l'alphabet** : cette option permet d'écouter tout l'alphabet en code morse.
- **le message** : tapez au clavier votre message ; vous pourrez l'entendre en morse; ou demandez à une personne de votre entourage d'en taper un que vous devrez deviner en l'écoutant.
- **le niveau** : pour progresser dans l'apprentissage du morse, augmentez la rapidité en changeant le niveau (de 10 à 1).

```

** MORSE **
1 ALPHABET 1
2 MESSAGE 2
3 NIVEAU 3
VOTRE CHOIX : █

NIVEAU: 5
    
```





```

10 REM MORSE
20 REM -----
30 LET A=32767
40 LET D=32766
100 LET N=5
150 LET A#=".- -.-.-.-.
.-.-.-.-.
.-.-.-.-.
.-.-.-.-.
.-.-.-.-.
"
200 POKE A,7
210 POKE D,254
220 POKE A,0
230 POKE D,64
240 POKE A,1
250 POKE D,1
260 REM -----
290 REM MENU
300 CLS
305 PRINT TAB (10);"** MORSE **
"
310 PRINT
320 PRINT TAB (9);"1 ALPHABET 1
"
320 PRINT TAB (9);"1 ALPHABET 1
"
330 PRINT TAB (9);"2 MESSAGE 2
"
340 PRINT TAB (9);"3 NIVEAU 3
"
350 PRINT AT 20,11;"NIVEAU:";N
360 PRINT AT 6,8;"VOTRE CHOIX :
1"
400 LET W#=INKEY#
410 IF W#="" THEN GOTO 400
420 LET W=VAL W#
430 IF W<1 OR W>3 THEN GOTO 400
440 GOTO 700+W*100
480 REM -----
490 REM EDITION MESSAGE
500 FOR I=1 TO LEN (M#)
510 LET P=CODE M#(I)
520 IF P=0 THEN GOTO 710
530 IF P<38 OR P>63 THEN GOTO 7
00
540 LET P=P-38
550 FOR K=P*4+1 TO P*4+4
560 IF A#(K TO K)="" THEN GOTO
690
570 POKE A,8
580 POKE D,15
590 IF A#(K TO K)="." THEN GOTO
630
600 FOR L=1 TO N*3
610 NEXT L
620 GOTO 650
630 FOR L=1 TO N
640 NEXT L
650 POKE D,0
660 FOR L=1 TO N
670 NEXT L
680 NEXT K
690 FOR L=1 TO N
700 NEXT L
710 NEXT I
720 RETURN

```

```

780 REM -----
790 REM ALPHABET
800 PRINT AT 15,11;"ALPHABET"
810 LET M#="ABCDEFGHIJKLMNOPS
TUVWXYZ"
820 GOSUB 500
830 GOTO 300
880 REM -----
890 REM MESSAGE
900 CLS
910 PRINT AT 21,0;"ENTREZ VOTRE
MESSAGE ?"
920 INPUT M#
930 IF M#="" THEN GOTO 300
940 GOSUB 500
950 GOTO 900
980 REM -----
990 REM NIVEAU DE RAPIDITE
1000 CLS
1010 PRINT AT 21,0;"NIVEAU (ENTR
E 1 ET 10) ?"
1020 INPUT N
1030 IF N<1 OR N>10 THEN GOTO 10
00
1040 GOTO 300
    
```

LE PROGRAMME**Lignes 30 à 250**

On place dans A\$ la liste des codes morse pour tout l'alphabet. Chaque code doit prendre quatre caractères ; des blancs sont ajoutés pour les codes ne comprenant que un, deux, trois points ou tirets.

La carte son nécessite l'initialisation de ses registres avant utilisation. La variable A correspond à l'adresse du registre et la variable D aux données.

La valeur 254 dans le registre 7 branche la tonalité sur le canal "A".

Lignes 300 à 410

Menu : sans commentaire !

Lignes 500 à 720 (sous-programme)

On envoie sur la carte sonore les codes morse contenus dans la variable M\$. Chaque caractère est pris séparément (boucle I) et retransmis en code morse (boucle K).

Lignes 800 à 830

On place dans M\$ l'alphabet que l'on fait interpréter par le sous-programme des lignes 500 à 720.

Lignes 900 à 950

L'utilisateur entre au clavier son message dans la variable M\$. Il est interprété, comme l'alphabet, par le sous-programme en 500.

Lignes 1000 à 1040

On saisit dans la variable N le niveau du jeu. Cette variable est utilisée dans les boucles de temporisation du sous-programme "MESSAGE".

Histogramme

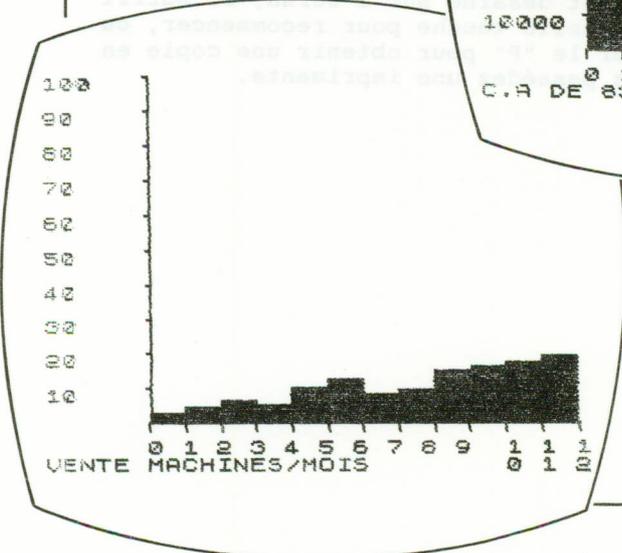
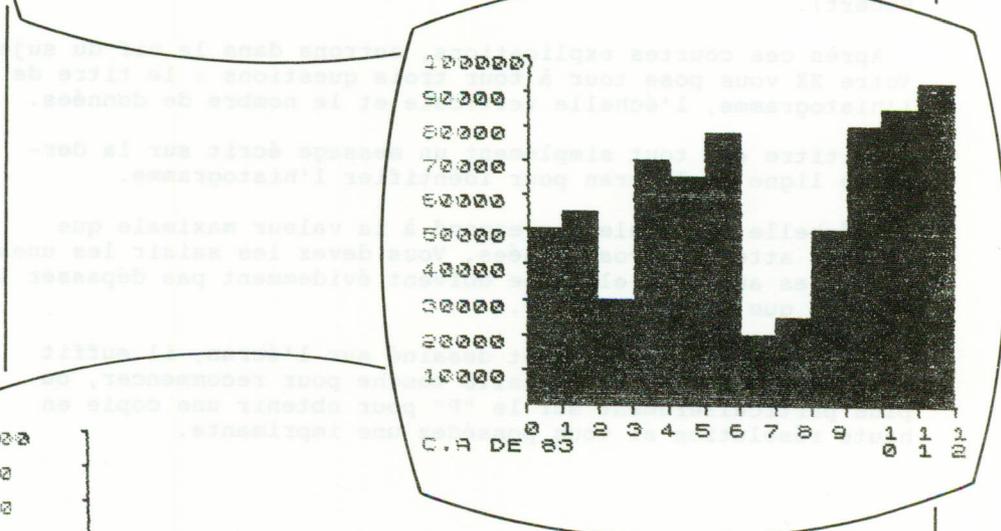
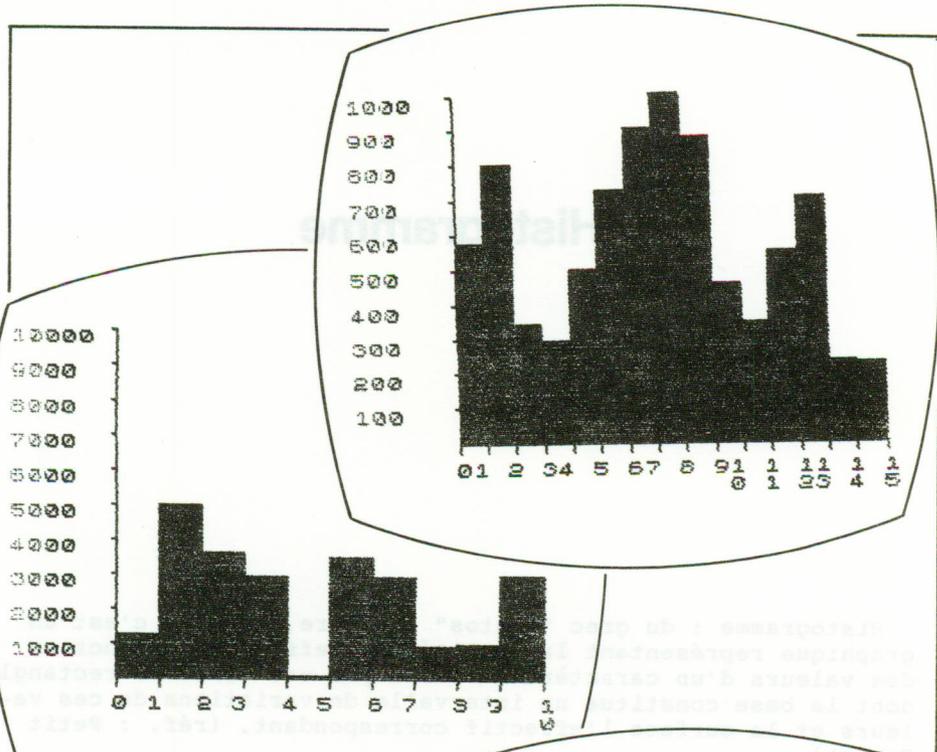
Histogramme : du grec "histos" (texture, trame), c'est un graphique représentant la densité d'un effectif en fonction des valeurs d'un caractère et formé par une série de rectangles dont la base constitue un intervalle de variations de ces valeurs et la surface l'effectif correspondant. (réf. : Petit Robert).

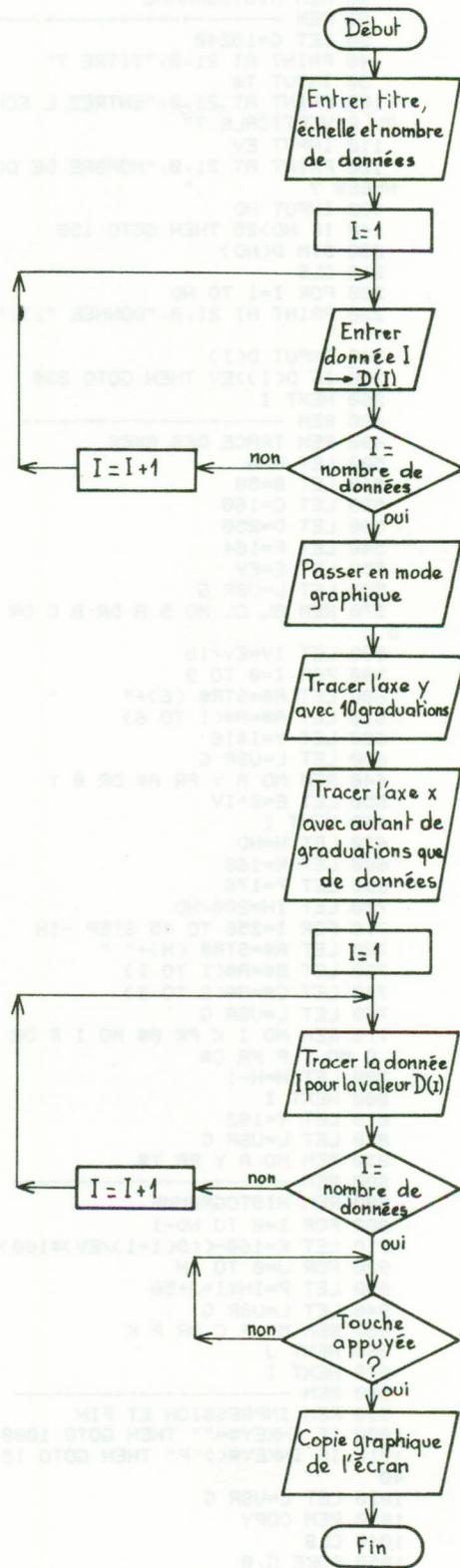
Après ces courtes explications, entrons dans le vif du sujet. Votre ZX vous pose tour à tour trois questions : le titre de l'histogramme, l'échelle verticale et le nombre de données.

Le titre est tout simplement un message écrit sur la dernière ligne de l'écran pour identifier l'histogramme.

L'échelle verticale correspond à la valeur maximale que peuvent atteindre vos données. Vous devez les saisir les unes après les autres ; elles ne doivent évidemment pas dépasser le maximum que vous avez fixé.

Dès que l'histogramme est dessiné sur l'écran, il suffit d'appuyer sur n'importe quelle touche pour recommencer, ou plus particulièrement sur le "P" pour obtenir une copie en haute résolution si vous possédez une imprimante.





ETUDES POUR ZX81

```

10 REM HISTOGRAMME
20 REM -----
30 LET G=10240
40 PRINT AT 21,0;"TITRE ?"
50 INPUT T$
100 PRINT AT 21,0;"ENTREZ L ECH
ELLE VERTICALE ?"
110 INPUT EV
120 PRINT AT 21,0;"NOMBRE DE DO
NNEES ? "
130 INPUT ND
140 IF ND>25 THEN GOTO 130
200 DIM D(ND)
210 CLS
220 FOR I=1 TO ND
230 PRINT AT 21,0;"DONNEE ";I;"
?"
240 INPUT D(I)
250 IF D(I)>EV THEN GOTO 230
260 NEXT I
480 REM -----
490 REM TRACE DES AXES
500 LET A=0
510 LET B=50
520 LET C=160
530 LET D=250
540 LET F=164
550 LET E=EV
560 LET L=USR G
570 REM BL CL MO B A DR B C DR
D C
580 LET IV=EV/10
590 FOR I=0 TO 9
600 LET A#=STR$(E)+ " "
610 LET A#=A#(1 TO 6)
620 LET Y=I*16
630 LET L=USR G
640 REM MO A Y PR A# DR B Y
650 LET E=E-IV
660 NEXT I
670 LET H=ND
680 LET K=168
690 LET P=176
700 LET IH=200/ND
710 FOR I=250 TO 45 STEP -IH
720 LET A#=STR$(H)+ " "
730 LET B#=A#(1 TO 1)
740 LET C#=A#(2 TO 2)
760 LET L=USR G
770 REM MO I K PR B# MO I F DR
I C MO I P PR C#
780 LET H=H-1
800 NEXT I
810 LET Y=182
820 LET L=USR G
830 REM MO A Y PR T#
880 REM -----
890 REM HISTOGRAMME
900 FOR I=0 TO ND-1
910 LET K=160-((D(I+1)/EV)*160)
920 FOR J=0 TO IH
930 LET P=IH*I+J+50
940 LET L=USR G
950 REM MO P C DR P K
960 NEXT J
970 NEXT I
980 REM -----
990 REM IMPRESSION ET FIN
1000 IF INKEY#="" THEN GOTO 1000
1010 IF INKEY#("<") THEN GOTO 10
40
1020 LET L=USR G
1030 REM COPY
1040 CLS
1050 POKE G,0

```

LE PROGRAMME

Lignes 30 à 260

La variable G correspond au point d'entrée du programme en langage machine contenu dans l'EPROM de la carte haute résolution (EPROM = mémoire morte programmable électriquement).

On saisit dans les variables T\$, EV et ND le titre, l'échelle verticale et le nombre de données. L'échelle verticale correspond aux valeurs maximums des données. Ces dernières sont placées dans le tableau D.

Lignes 500 à 830

La syntaxe de l'interpréteur de la carte haute résolution nécessite l'emploi de variables et non de valeurs numériques. C'est pourquoi l'on utilise autant de variables différentes entre les lignes 500 et 550. Celles-ci représentent des points bien particuliers sur l'écran. L'échelle verticale est découpée en dix graduations ; les valeurs sont toujours bien alignées, ce qui n'est malheureusement pas le cas pour l'horizontale.

Lignes 900 à 970

On trace sur l'écran toutes les données une à une. Chaque donnée est représentée par un bloc noir dont la base correspond à une graduation et la hauteur à la valeur de la donnée. Le lecteur peut toujours modifier cette partie pour obtenir d'autres figures. (Ex : tracer une simple boîte par l'instruction BOX X,Y).

Lignes 1000 à 1050

L'histogramme étant totalement tracé, l'utilisateur doit appuyer sur une touche. Si celle-ci est "P", on utilise la fonction COPY de la carte haute-résolution pour garder une trace de l'écran sur papier.

IDENTIFICATION des VARIABLES

NOM SYMBOLIQUE	TYPE (*)	ORIGINE				OBSERVATIONS
		Clavier	Programme	Cassette	Disquette	
A	VARIABLE	X				Point de repère sur l'écran pour le tracé des axes
A \$	VARIABLE	X				Graduation de l'échelle horizontal sous forme alphanumérique pour la carte haute-résolution
B	VARIABLE	X				Point de repère sur l'écran pour le tracé des axes
B \$	VARIABLE	X				Premier caractère de A\$
C	VARIABLE	X				Point de repère sur l'écran pour le tracé des axes
C \$	VARIABLE	X				Deuxième caractère de A\$
D	VARIABLE	X				Point de repère sur l'écran pour le tracé des axes
D ()	TABLEAU	X				Contient les données de l'histogramme
E	COMPTEUR	X				Utilisé pour sauvegarder EV
E V	VARIABLE	X				Echelle verticale
F	VARIABLE	X				Point de repère sur l'écran pour le tracé des axes
G	CONSTANTE	X				Point d'entrée du sous-programme haute résolution
H	COMPTEUR	X				Graduation de l'échelle horizontal
I	INDICE	X				Boucle FOR/NEXT

6.1980

(*) Légende : aiguillage, compteur, constante, index, indice, tableau, variable. Reproduction interdite (c) Editions du P.S

Parités

Voici un jeu pour exercer vos neurones. Ses règles en sont très simples.

Quarante cartes retournées apparaissent à l'écran. Chaque joueur retourne deux cartes à tour de rôle en donnant leurs coordonnées. Si elles sont identiques, elles sont effacées de l'écran ; le compteur du joueur est augmenté de deux points et c'est toujours à lui de jouer.

Dans le cas contraire, ces deux cartes sont retournées à nouveau et le joueur suivant fait son choix. Le but de ce jeu est de trouver toutes les paires de cartes. Mais pour cela, il faut beaucoup de mémoire (papier et crayons sont interdits!).

	A	B	C	D	E	F	G	H	I	J
1	■	■	■	■	■	■	■	■	■	■
2	■	■	■	■	■	■	■	■	■	■
3	■	■	■	■	■	■	■	■	■	■
4	■	■	■	■	■	■	■	■	■	■

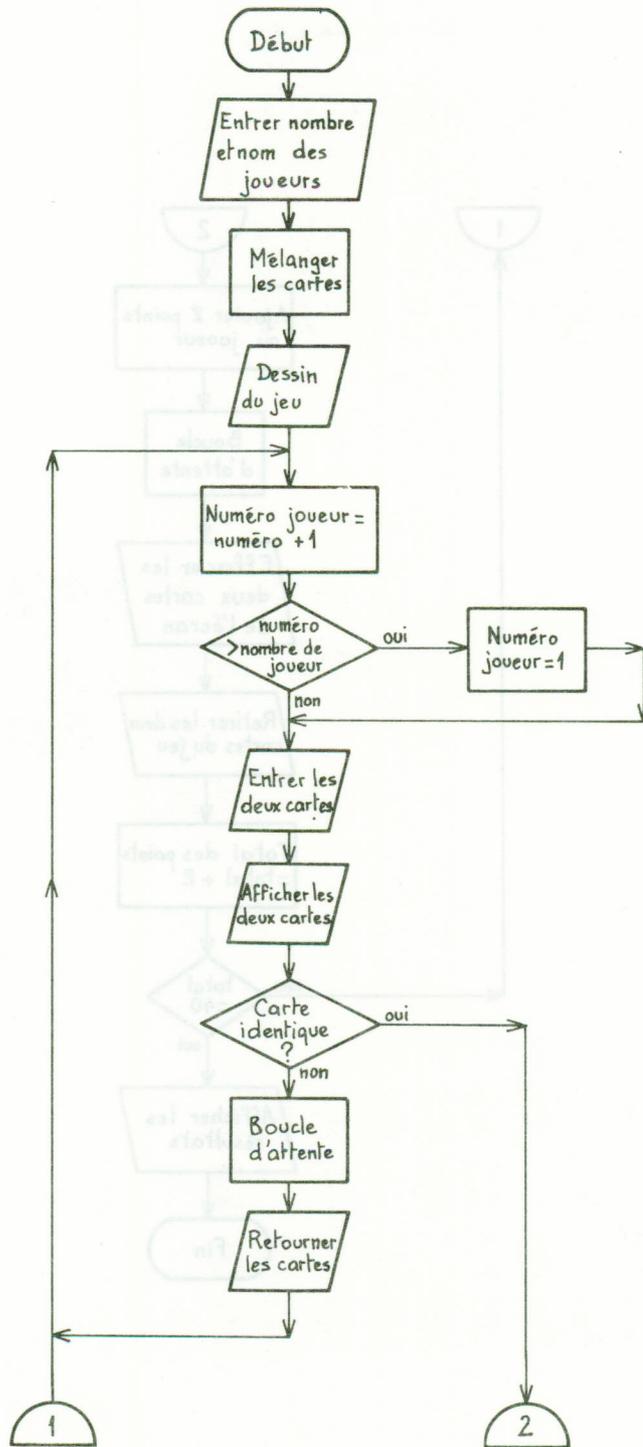
J-FRANCOIS (CARTES=0) :
CARTE 1 ?

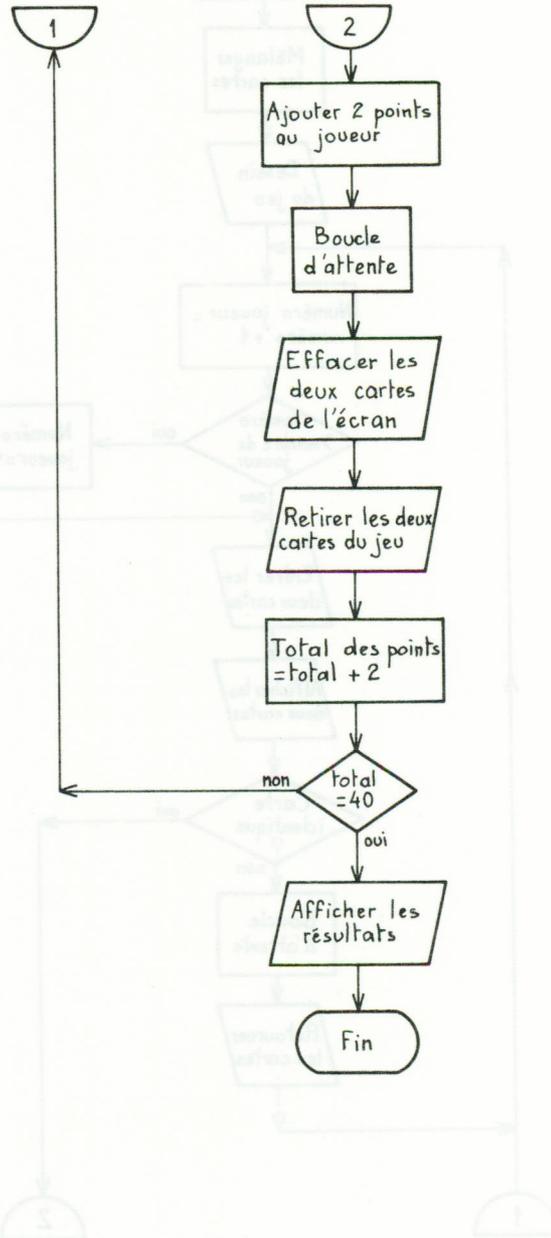
	A	B	C	D	E	F	G	H	I	J
1	■	■	■	■	■	■	■	■	■	■
2	■	■	■	■	■	■	■	■	■	■
3	■	■	■	■	■	■	■	■	■	■
4	■	■	■	■	■	■	■	■	■	■

CATHERINE (CARTES=10) :
CARTE 1 ?

	A	B	C	D	E	F	G	H	I	J
1	■	■	■	■	■	■	■	■	■	■
2	■	■	■	■	■	■	■	■	■	■
3	■	■	■	■	■	■	■	■	■	■
4	■	■	■	■	■	■	■	■	■	■

J-FRANCOIS (CARTES=6) :
CARTE 2 ?





```

10 REM PARITES
20 REM -----
30 PRINT AT 21,0;"NOMBRE DE JO
UEURS ?"
40 INPUT N
50 IF N<2 OR N>5 THEN GOTO 40
60 DIM N$(N,10)
70 DIM P(N)
80 LET NJ=0
90 LET TP=0
100 FOR I=1 TO N
110 PRINT AT 21,0;"NOM DU JOUEU
R ";I;" ? "
120 INPUT N$(I)
130 NEXT I
140 DIM R$(2,2)
150 DIM X(2)
160 DIM Y(2)
180 CLS
190 PRINT AT 10,10;"JE MELANGE"
;AT 12,13;"->"
200 DIM C(10,4)
210 LET F$="3343"
220 LET C$="3333"
230 FOR I=1 TO 5
240 FOR J=1 TO 4
250 LET C(I,J)=I*10+J
260 LET C(I+5,J)=I*10+J
270 NEXT J
280 NEXT I
300 FOR I=1 TO 120
310 LET L1=INT (RND*10)+1
320 LET C1=INT (RND*4)+1
330 LET L2=INT (RND*10)+1
340 LET C2=INT (RND*4)+1
350 LET A=C(L1,C1)
360 LET C(L1,C1)=C(L2,C2)
370 LET C(L2,C2)=A
380 PRINT AT 12,15;I
390 NEXT I
480 REM -----
490 REM DESSIN DU JEU
500 CLS
510 FOR I=1 TO 10
520 PRINT TAB I*3;CHR$(37+I);
530 NEXT I
540 FOR I=1 TO 4
550 PRINT AT I*4,0;I
560 FOR J=1 TO 10
570 PRINT AT I*4-1,J*3;"■";AT
I*4,J*3;"■";AT I*4+1,J*3;"■"
580 NEXT J
590 NEXT I

```

```

680 REM -----
690 REM LE JEU...
700 LET NJ=NJ+1
710 IF NJ>N THEN LET NJ=1
720 PRINT AT 19,0;N#(NJ);" (CAR
TES=";P(NJ);") : "
730 FOR K=1 TO 2
735 LET R#(2)="
740 PRINT AT 21,0;"CARTE ";K;"
?"
750 INPUT R#(K)
760 IF CODE R#(K)<38 OR CODE R#
(K)>47 THEN GOTO 750
770 IF CODE R#(K,2 TO 2)<29 OR
CODE R#(K,2 TO 2)>32 THEN GOTO 7
50
780 LET X(K)=CODE R#(K)-37
790 LET Y(K)=CODE R#(K,2 TO 2)-
28
800 IF R#(1)=R#(2) THEN GOTO 75
0
810 IF C(X(K),Y(K))=0 THEN GOTO
750
820 LET A=INT (C(X(K),Y(K))/10)
830 LET B=C(X(K),Y(K))-A*10
840 PRINT AT Y(K)*4,X(K)*3;F#(A
TO A);C#(B TO B)
850 NEXT K
900 IF C(X(1),Y(1))=C(X(2),Y(2)
) THEN GOTO 1000
910 FOR I=1 TO 100
920 NEXT I
930 PRINT AT Y(1)*4,X(1)*3;"■■■"
940 PRINT AT Y(2)*4,X(2)*3;"■■■"
950 GOTO 700
980 REM -----
990 REM GAGNE...
1000 LET P(NJ)=P(NJ)+2
1010 FOR I=1 TO 100
1020 NEXT I
1030 FOR I=1 TO 2
1040 PRINT AT Y(I)*4-1,X(I)*3;"
";AT Y(I)*4,X(I)*3;" ";AT Y(I)
*4+1,X(I)*3;" "
1050 LET C(X(I),Y(I))=0
1060 NEXT I
1070 LET TP=TP+2
1080 IF TP=40 THEN GOTO 1500
1090 GOTO 720
1480 REM -----
1490 REM FIN DU JEU
1500 CLS
1510 PRINT TAB 10;"RESULTATS"
1520 PRINT
1530 PRINT "NOM          POINTS"
1540 FOR I=1 TO N
1550 PRINT
1560 PRINT N#(I);"-> ";P(I)
1570 NEXT I
1580 IF INKEY#="" THEN GOTO 1580
1590 CLS
1600 RUN

```

PAGE	NOM DU PROGRAMME	DATE
1	LE PROGRAMME	
	Lignes 30 à 39	
	Après la saisie du nombre et des noms des joueurs, on place dans le tableau C le code des cartes. Ce code est composé du numéro de figure multiplié par 10, plus le numéro de la couleur.	
	Ce tableau est mélangé en prenant quatre nombres aléatoires (L1, L2, C1, C2) pour inverser deux cartes.	
	Lignes 700 à 850	
	Les deux cartes sont saisies en vérifiant : - qu'elles ne sont pas identiques, - qu'elles existent bien à l'endroit indiqué.	
	Chaque réponse est décodée et placée dans les tableaux X et Y (coordonnées des cartes).	
	Lignes 900 à 950	
	Dans le cas où ces deux cartes sont identiques, on saute directement à la ligne 1000 ; sinon on les efface après une petite boucle d'attente.	
	Lignes 1000 à 1090	
	On incrémente le total des points du joueur (tableau P), puis l'on efface de l'écran et du tableau C les cartes données. Si le total des cartes effacées est inférieur à 40, on continue le jeu.	
	Lignes 1500 à 1600	
	On affiche le nom et le nombre de points de chaque joueur avant de recommencer une partie.	

DATE		NOM DU PROGRAMME : PARITES				PAGE 1 / 1	
IDENTIFICATION des VARIABLES							
NOM SYMBOLIQUE	TYPE (*)	ORIGINE				LONG.	OBSERVATIONS
		Clavier	Programme	Cassette	Disquette		
A	VARIABLE	X					Numéro de la figure à afficher.
B	VARIABLE	X					Numéro de la couleur à afficher.
C 1	VARIABLE	X					{ Nombres aléatoires pour le mélange des cartes.
C 2	VARIABLE	X					
C ()	TABLEAU	X					Tableau des cartes.
C \$	TABLEAU	X					Liste des couleurs.
F \$	TABLEAU	X					Liste des figures.
I	INDICE	X					Boucle FOR/NEXT.
J	INDICE	X					Boucle FOR/NEXT.
K	INDICE	X					Boucle FOR/NEXT.
L 1	VARIABLE	X					{ Nombres aléatoires pour le mélange des cartes.
L 2	VARIABLE	X					
N	VARIABLE	X					Nombre de joueurs.
N, J	VARIABLE	X					Numéro du joueur.
N \$ ()	TABLEAU	X					Noms des joueurs.
P, ()	TABLEAU	X					Total des points de chaque joueur.
R, \$ ()	TABLEAU	X					Réponse du joueur.
T, P	COMPTEUR	X					Total des points.
X ()	TABLEAU	X					{ Positions des deux cartes choisies par le joueur.
Y ()	TABLEAU	X					

6/1980

(*) Légende : aiguillage, compteur, constante, index, indice, tableau, variable. Reproduction interdite (c) Editions du P.S.

Piranhas

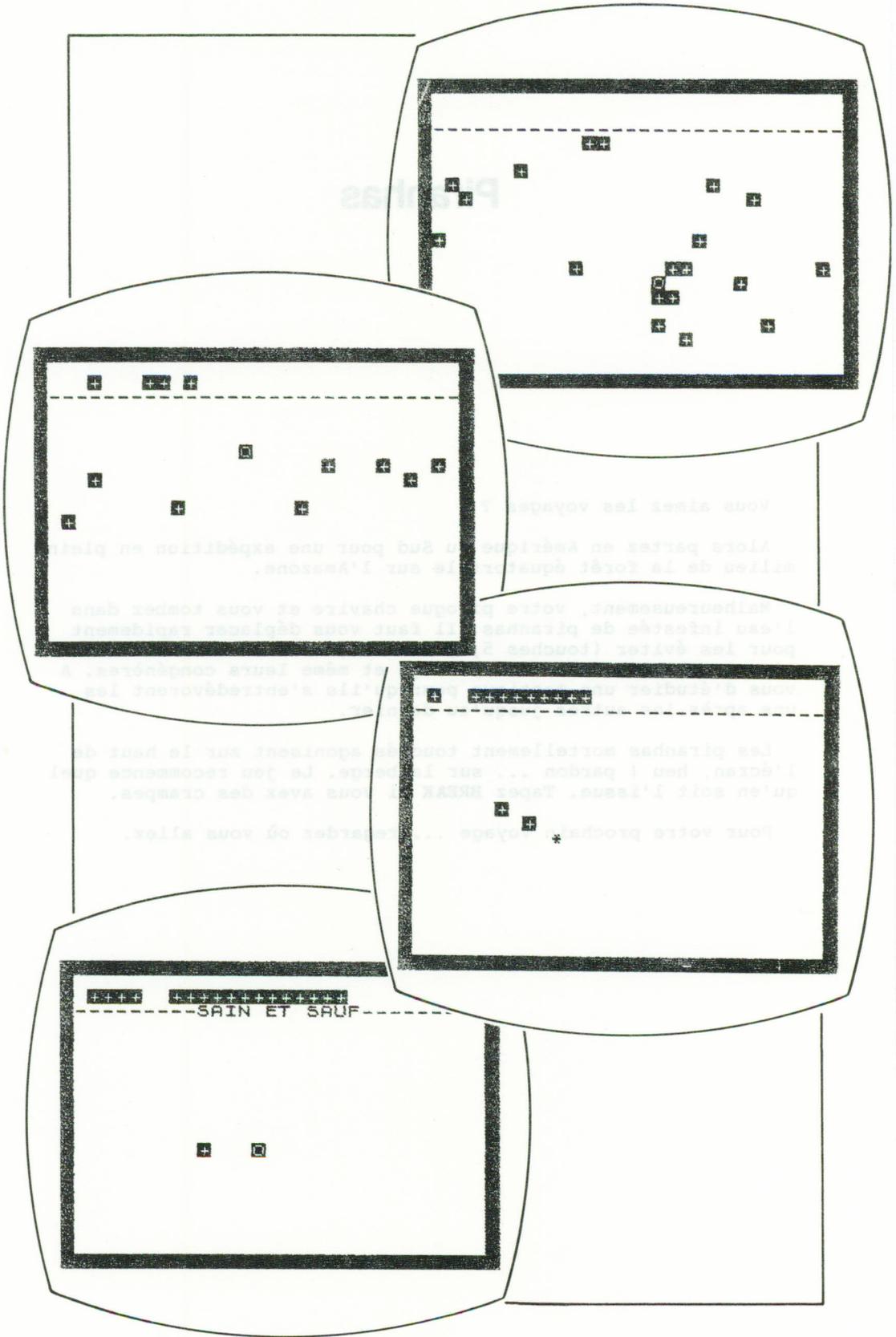
Vous aimez les voyages ?

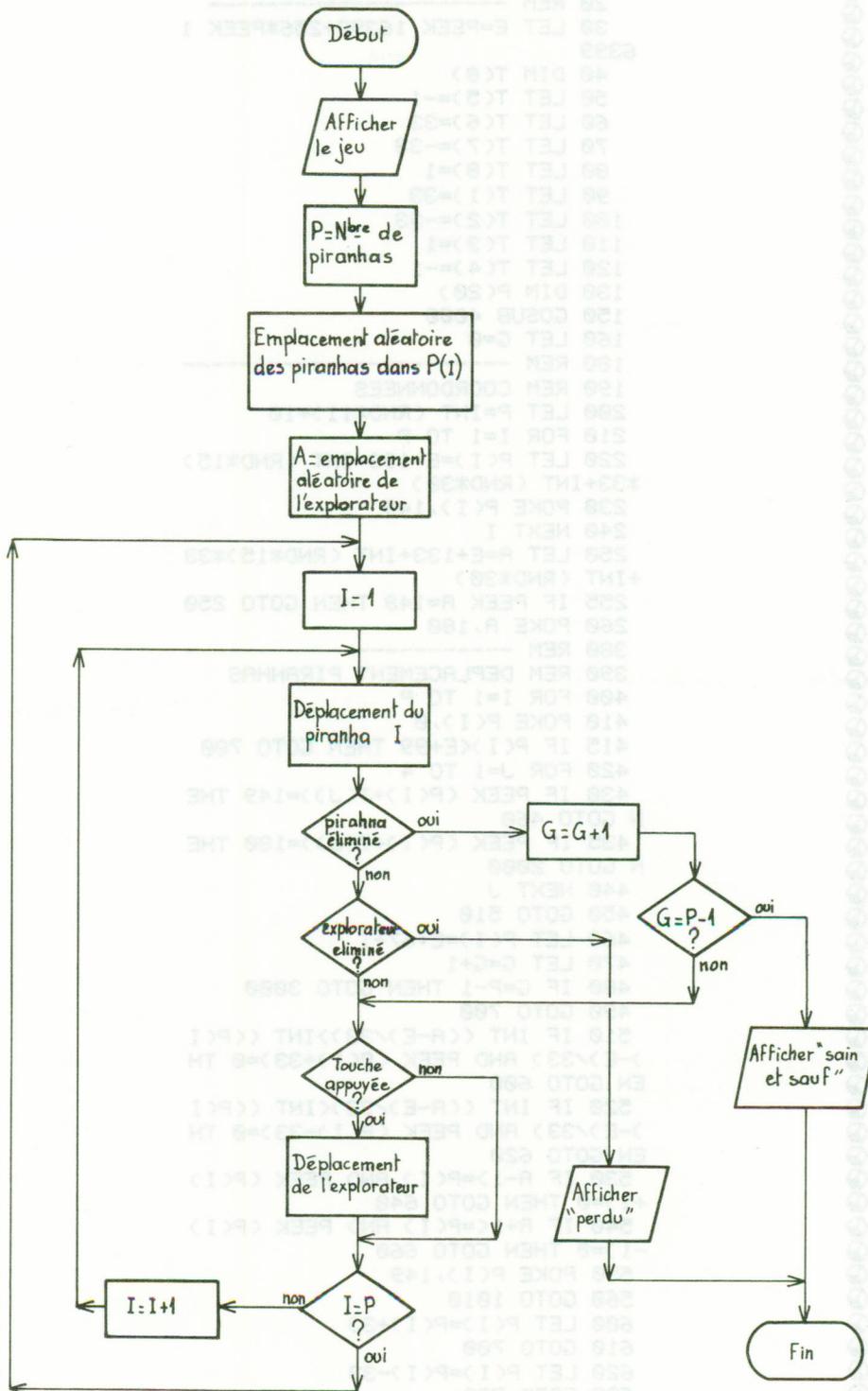
Alors partez en Amérique du Sud pour une expédition en plein milieu de la forêt équatoriale sur l'Amazonie.

Malheureusement, votre pirogue chavire et vous tombez dans l'eau infestée de piranhas. Il faut vous déplacer rapidement pour les éviter (touches 5,6, 7 et 8), car ils dévorent tout ce qui se trouve sur leur passage et même leurs congénères. A vous d'étudier une tactique pour qu'ils s'entredévorent les uns après les autres jusqu'au dernier.

Les piranhas mortellement touchés agonisent sur le haut de l'écran, heu ! pardon ... sur la berge. Le jeu recommence quel qu'en soit l'issue. Tapez **BREAK** si vous avez des crampes.

Pour votre prochain voyage ... regardez où vous allez.





```

10 REM PIRANHAS
20 REM -----
30 LET E=PEEK 16398+256*PEEK 1
6399
40 DIM T(8)
50 LET T(5)=-1
60 LET T(6)=33
70 LET T(7)=-33
80 LET T(8)=1
90 LET T(1)=33
100 LET T(2)=-33
110 LET T(3)=1
120 LET T(4)=-1
130 DIM P(20)
150 GOSUB 4000
160 LET G=0
180 REM -----
190 REM COORDONNEES
200 LET P=INT (RND*11)+10
210 FOR I=1 TO P
220 LET P(I)=E+133+INT (RND*15)
*33+INT (RND*30)
230 POKE P(I),149
240 NEXT I
250 LET A=E+133+INT (RND*15)*33
+INT (RND*30)
255 IF PEEK A=149 THEN GOTO 250
260 POKE A,180
380 REM -----
390 REM DEPLACEMENT PIRANHAS
400 FOR I=1 TO P
410 POKE P(I),0
415 IF P(I)<E+99 THEN GOTO 700
420 FOR J=1 TO 4
430 IF PEEK (P(I)+T(J))=149 THE
N GOTO 460
435 IF PEEK (P(I)+T(J))=180 THE
N GOTO 2000
440 NEXT J
450 GOTO 510
460 LET P(I)=E+67+I
470 LET G=G+1
480 IF G=P-1 THEN GOTO 3000
490 GOTO 700
510 IF INT ((A-E)/33)>INT ((P(I)
)-E)/33) AND PEEK (P(I)+33)=0 TH
EN GOTO 600
520 IF INT ((A-E)/33)<INT ((P(I)
)-E)/33) AND PEEK (P(I)-33)=0 TH
EN GOTO 620
530 IF A-1>P(I) AND PEEK (P(I)
+1)=0 THEN GOTO 640
540 IF A+1<=P(I) AND PEEK (P(I)
-1)=0 THEN GOTO 660
550 POKE P(I),149
560 GOTO 1010
600 LET P(I)=P(I)+33
610 GOTO 700
620 LET P(I)=P(I)-33
630 GOTO 700
640 LET P(I)=P(I)+1
650 GOTO 700
660 LET P(I)=P(I)-1

```

```

680 REM -----
690 REM DEPLACEMENT EXPLORATEUR
700 IF INKEY#="" THEN GOTO 1000
710 LET D#=INKEY#
720 IF CODE D#<33 OR CODE D#>36
THEN GOTO 1000
730 LET D=VAL D#
740 IF PEEK (A+T(D))<>0 THEN GO
TO 1000
750 POKE A,0
760 LET A=A+T(D)
770 POKE A,180
1000 POKE P(I),149
1010 NEXT I
1020 GOTO 400
1980 REM -----
1990 REM FIN DE PARTIE: PERDU
2000 FOR I=1 TO 100
2010 POKE A,149
2020 POKE A,180
2030 POKE A,23
2040 POKE A,0
2050 NEXT I
2060 CLS
2070 GOTO 150
2980 REM -----
2990 REM FIN DE PARTIE: GAGNE
3000 FOR I=1 TO 50
3010 PRINT AT 3,10;"SAIN ET SAUF
"
3020 POKE A,23
3030 PRINT AT 3,10;"
"
3040 POKE A,0
3050 NEXT I
3060 CLS
3070 GOTO 150
3980 REM -----
3990 REM DESSIN DU JEU
4000 FOR I=0 TO 31
4010 IF I>21 THEN GOTO 4040
4020 POKE E+(I*33),128
4030 POKE E+(I*33)+31,128
4040 POKE E+I,128
4050 POKE E+693+I,128
4055 IF I>30 THEN GOTO 4070
4060 POKE E+99+I,22
4070 NEXT I
4080 RETURN

```

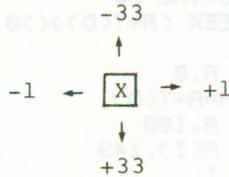
LE PROGRAMME

Lignes 30

Dans les cases mémoires 16398 et 16399 se trouve le pointeur de début d'écran. Cette adresse permet d'écrire sur l'écran par des instructions POKE beaucoup plus rapides que des PRINT AT.

Lignes 40 à 130

On place dans le tableau T toutes les directions de déplacement des piranhas ou de l'explorateur.



Lignes 200 à 260

Le nombre de piranhas est déterminé par la variable P. On remplit le tableau P avec les coordonnées aléatoires des piranhas, A étant celles de l'explorateur. On les affiche à l'écran par des instructions POKE.

Lignes 400 à 660

Chaque piranha est déplacé sur l'écran dans le sens qui le rapproche le plus de l'explorateur sauf s'il trouve un congénère sur une case adjacente. Dans ce cas, il s'empresse de l'avalier.

Lignes 700 à 1020

Si une touche est appuyée, on déplace l'explorateur en se référant toujours au tableau T.

Lignes 2000 à 2070

C'est la fin de la partie puisqu'un piranha vous a dévoré.

Lignes 3000 à 3070

C'est aussi une fin de partie, mais plus agréable : vous êtes sauvé car il ne reste plus qu'un seul piranha.

Lignes 4000 à 4080

On dessine une bordure noire sur l'écran pour éviter que les piranhas ou l'explorateur en sortent.

Baccara

Retrouvez l'ambiance fabuleuse des casinos en jouant au Baccara, car ici l'on joue gros, très gros.

Le banquier - votre adversaire - tire quatre cartes d'un sabot composé de 8 jeux de 52 cartes. Il vous donne deux cartes et garde les deux autres.

Toutes les cartes, depuis l'AS (qui vaut 1) jusqu'au 9 ont leurs valeurs nominales ; les autres cartes valent 0. Pour obtenir la valeur d'un jeu, il suffit d'additionner les cartes en ne conservant du total que le chiffre des unités.

Pour le ponte (c'est vous), 4 possibilités se présentent :

- vous avez une main naturelle c'est-à-dire 8 ou 9 ; vous abattez vos cartes. Le jeu s'arrête là.
- vous avez 6 ou 7 ; c'est au tour du banquier de parler.
- vous avez 5 ; vous choisissez de prendre ou non une carte.
- vous avez entre 0 et 4 ; vous devez obligatoirement prendre une carte.

Seule la 3ème carte que vous auriez pu tirer est visible du banquier.

Puis c'est au tour du banquier de parler. Si vous avez 8 ou 9, il faut comparer les jeux. Sinon le banquier a le choix de prendre une carte ou non (son choix est déterminé par un tableau).

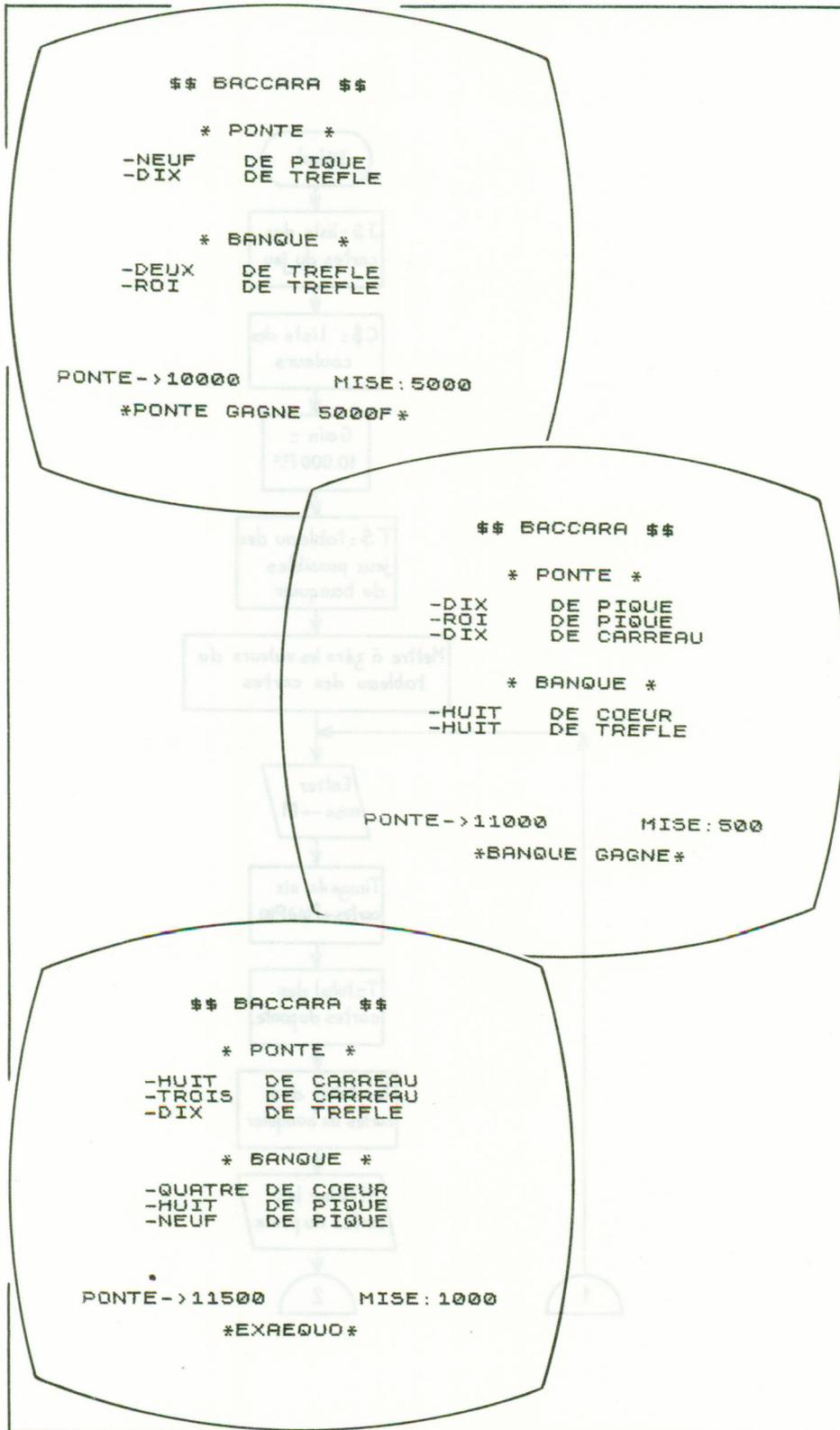
A ce moment-là toutes les cartes sont retournées ; le jeu le plus près de 9 gagne.

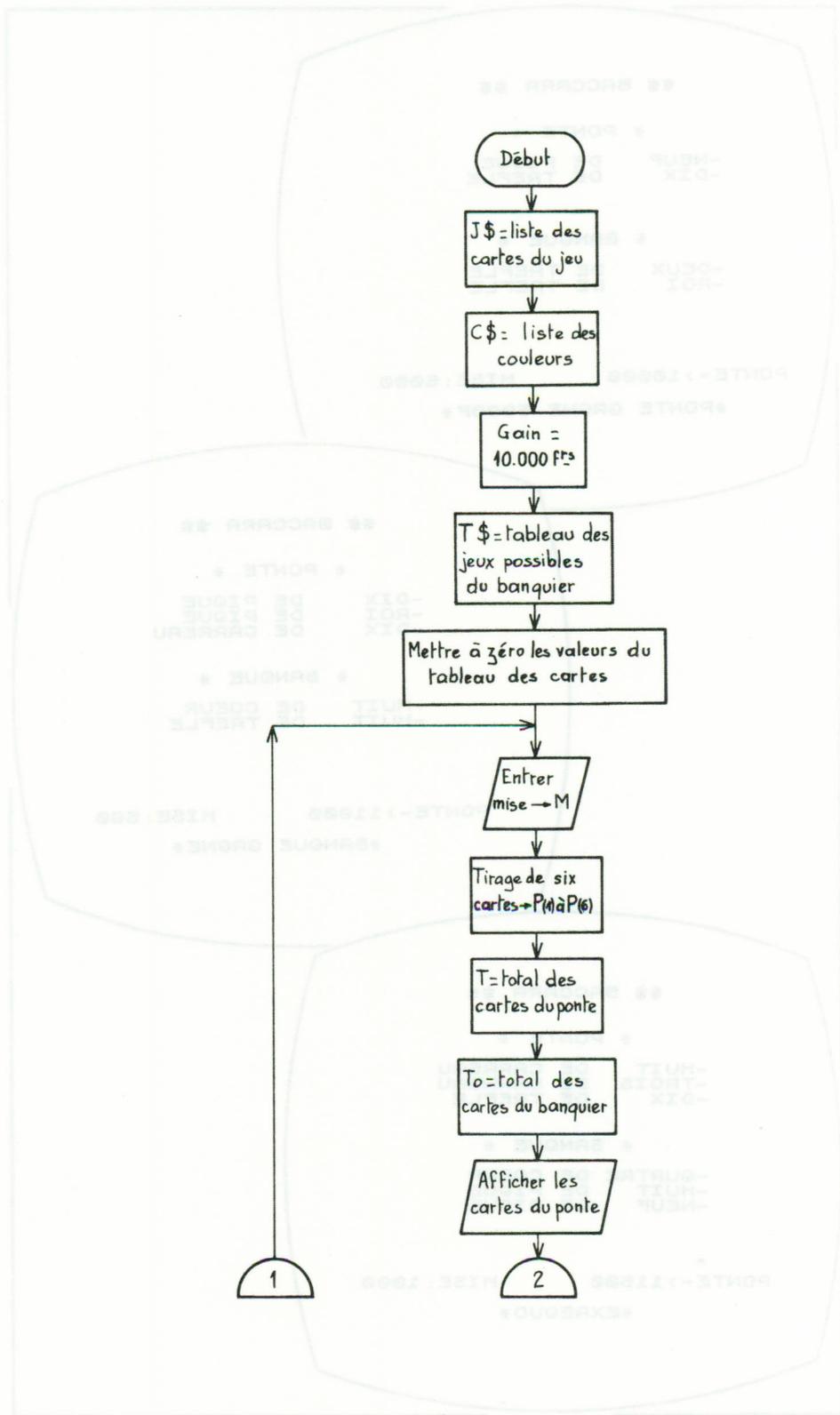
Un crédit de 10.000 Francs vous est alloué au début du jeu. Celui-ci ne s'arrête que faute de combattants (ou plutôt faute d'argent !).

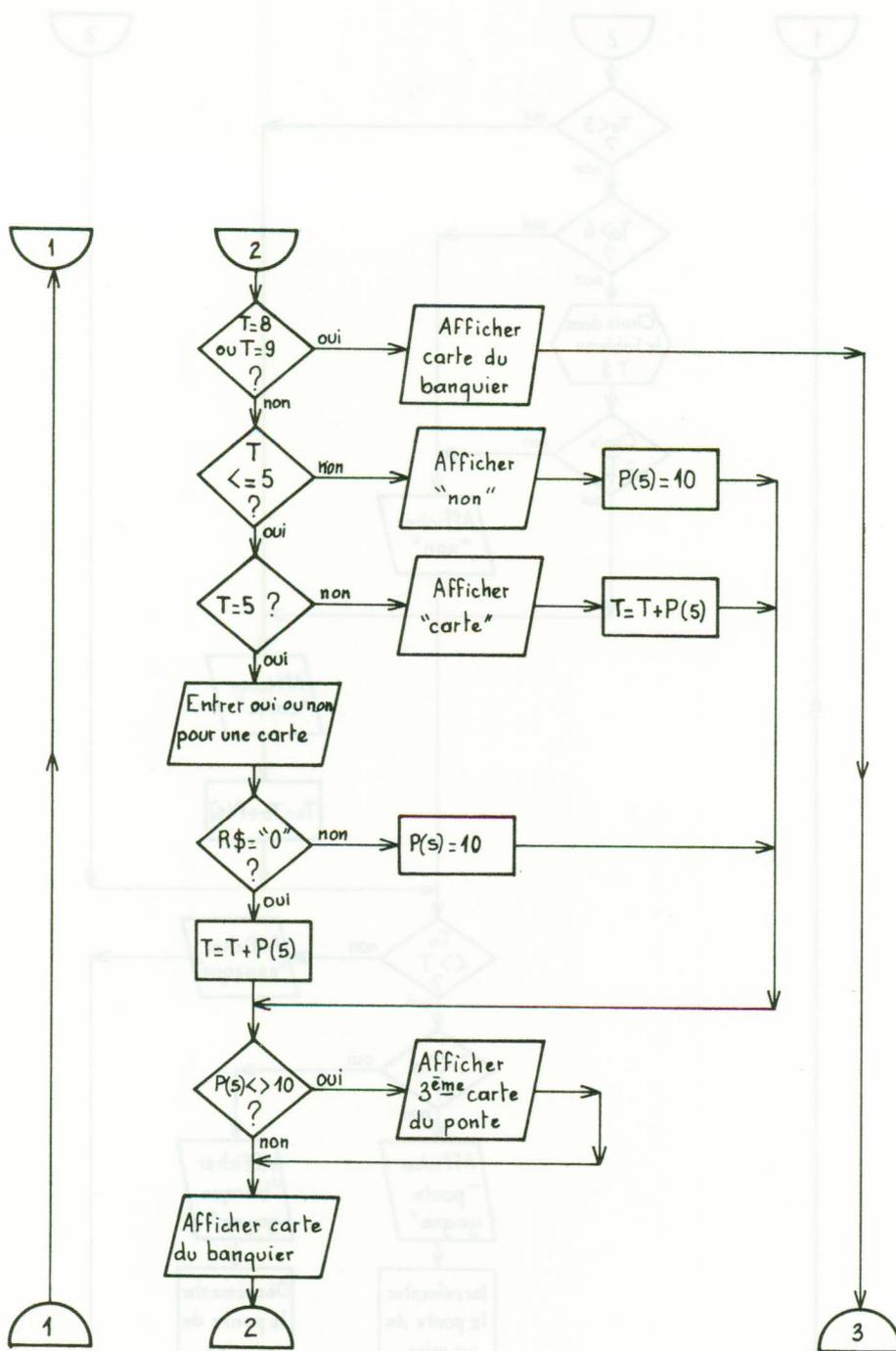
```
    ** BACCARA **  
    * PONTE *  
-AS      DE TREFLE  
-DEUX    DE TREFLE  
  
    * BANQUE *  
-  
-  
  
    PONTE->9500      MISE:1000  
    *PONTE:CARTE*
```

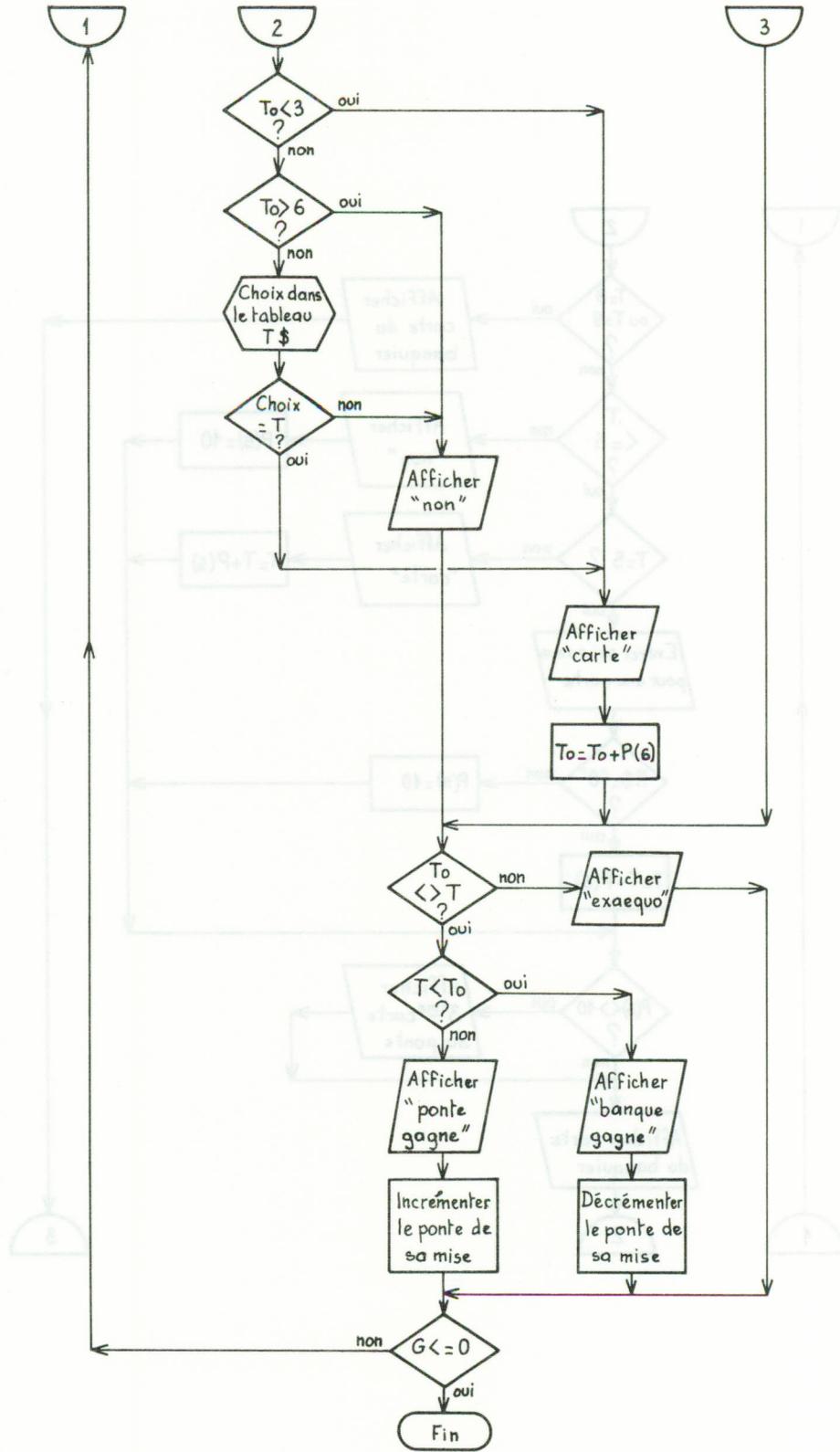
```
    ** BACCARA **  
    * PONTE *  
-DIX     DE CARREAU  
-CINQ    DE PIQUE  
  
    * BANQUE *  
-  
-  
  
    PONTE->9500      MISE:1000  
    CARTE ? (O/N)
```

```
    ** BACCARA **  
    * PONTE *  
-SEPT    DE TREFLE  
-ROI     DE TREFLE  
  
    * BANQUE *  
-AS      DE TREFLE  
-SIX     DE CARREAU  
  
    PONTE->12500     MISE:500  
    *BANQUE:NON*
```









```

10 REM BACCARA
20 REM -----
30 LET J#="AS DEUX TROIS Q
UATRECINQ SIX SEPT HUIT NEU
F DIX VALET DAME ROI "
40 LET C#="PIQUE COEUR CARRE
AUTREFLE "
50 LET G=10000
100 DIM J(13,4)
110 DIM P(6)
120 DIM P#(6,18)
130 DIM T#(6,11)
140 LET T#(3)="TTTTTTTTRTT"
150 LET T#(4)="RRTTTTTRRT"
160 LET T#(5)="RRRRTTTTTRT"
170 LET T#(6)="RRRRRTTRRR"
180 CLS
190 PRINT AT 11,6;"JE MELANGE L
ES CARTES"
200 FOR I=1 TO 13
210 FOR J=1 TO 4
220 LET J(I,J)=0
230 NEXT J
240 NEXT I
250 LET NC=0
260 CLS
290 REM -----
290 REM MISE DU PONTE
300 PRINT AT 21,0;"MISE ? (MAXI
";G;" FRANCS)"
310 INPUT M
320 IF M>G OR M<=0 THEN GOTO 30
0
380 REM -----
390 REM TIRAGE DE 6 CARTES
400 FOR I=1 TO 6
410 LET P(I)=0
420 LET J=INT (RND*13)+1
430 LET C=INT (RND*4)+1
440 IF J<J,C>=8 THEN GOTO 420
450 LET J<J,C>=J<J,C>+1
460 IF J>9 THEN GOTO 480
470 LET P(I)=J
480 LET P#(I)=J#<<(J-1)*6+1 TO <
J-1)*6+6>+" DE "+C#<<(C-1)*7+1 TO
(C-1)*7+7)
490 NEXT I
500 LET T=P(1)+P(3)
510 IF T>9 THEN LET T=T-10
520 LET TO=P(2)+P(4)
530 IF TO>9 THEN LET TO=TO-10
580 REM -----
590 REM AFFICHAGE DES CARTES
600 CLS
610 PRINT TAB 9;"## BACCARA ##"
630 PRINT AT 3,11;"* PONTE *"
640 PRINT AT 5,6;"-";P#(1)
650 PRINT " -";P#(3)
660 PRINT AT 10,11;"* BANQUE *"
670 PRINT AT 12,6;"-#####"
680 PRINT " -#####"
690 PRINT AT 19,2;"PONTE->";G;A
T 19,20;"MISE:";M

```

ETUDES POUR ZX81

```

770 REM -----
780 REM CHOIX DU PONTE
790 IF T=9 OR T=8 THEN GOTO 970
800 IF T<=5 THEN GOTO 850
810 LET P<5>=10
820 LET M#="*PONTE:NON*"
830 GOSUB 2000
840 GOTO 1000
850 IF T=5 THEN GOTO 910
860 LET M#="*PONTE:CARTE*"
870 GOSUB 2000
880 LET T=T+P<5>
890 IF T>9 THEN LET T=T-10
900 GOTO 950
910 PRINT AT 21,0;"CARTE ? (0/1
)"
920 INPUT R#
930 IF R#="0" THEN GOTO 880
940 LET P<5>=10
950 PRINT AT 21,0;"
"
960 GOTO 1000
970 PRINT AT 12,7;P#<2>;AT 13,7
;P#<4>
980 GOTO 1300
990 REM -----
995 REM CHOIX DE LA BANQUE
1000 IF P<5><>10 THEN PRINT AT 7
,6;"-";P#<5>
1010 PRINT AT 12,7;P#<2>;AT 13,7
;P#<4>
1100 IF T<3 THEN GOTO 1200
1110 IF T>6 THEN GOTO 1130
1120 IF T#<(T,P<5>+1)="T" THEN G
OTO 1200
1130 LET M#="*BANQUE:NON*"
1140 GOSUB 2000
1150 GOTO 1300
1200 LET M#="*BANQUE:CARTE*"
1210 GOSUB 2000
1220 PRINT AT 14,6;"-";P#<6>
1230 LET T=T+P<6>
1240 IF T>9 THEN LET T=T-10
1280 REM -----
1290 REM RESULTATS
1300 IF T<>T THEN GOTO 1350
1310 LET M#="*EXAEQUO*"
1320 GOSUB 2000
1330 GOTO 1500
1350 IF T>T THEN GOTO 1400
1360 LET M#="*BANQUE GAGNE*"
1370 GOSUB 2000
1380 LET G=G-M
1390 GOTO 1500
1400 LET M#="*PONTE GAGNE "+STR#
(M)+ "F*"
1410 GOSUB 2000
1420 LET G=G+M
1480 REM -----
1490 REM FIN DE PARTIE
1500 PRINT AT 19,2;"PONTE->";G;"
";AT 19,20;"MISE:";M
1510 PRINT AT 21,8;"TAPEZ UNE TO
UCHE"
1520 LET NC=NC+6
1600 IF INKEY#="" THEN GOTO 1600
1610 CLS
1620 IF G<=0 THEN GOTO 1650
1630 IF NC>=204 THEN GOTO 190
1640 GOTO 300
1650 CLS
1660 PRINT AT 11,0;"MERCI POUR L
ES DIX MILLE FRANCS"
1670 PRINT " LE CASINO ACCEPTE L
ES CHEQUES"
1680 STOP
1980 REM -----
1990 REM MESSAGE
2000 LET A=(31-LEN (M#))/2
2010 PRINT AT 21,A;M#
2020 FOR I=1 TO 100
2030 NEXT I
2040 PRINT AT 21,A;"
"
2050 RETURN

```

LE PROGRAMME**Lignes 30 à 260**

On place dans J\$ la liste des noms des cartes et dans C\$ la liste des couleurs.

Le tableau T\$ contient les décisions que prendra le banquier. Chaque liste du tableau contient onze informations. Les dix premières correspondent à la valeur de la carte tirée par le ponte (entre 0 et 9). La onzième précise que le ponte n'a pas pris de carte. La lettre T conseille au banquier de tirer, alors que R l'incite à rester. Chaque ligne correspond au jeu du banquier. Seules sont représentées les solutions pour des valeurs comprises entre 3 et 6 puisque le banquier tire toujours une carte quand il a moins de 3 points et refuse quand il a plus de 6 points.

Le tableau J correspond au sabot. Il est remis à zéro quand les cartes sont toutes tirées.

Lignes 300 à 320

On saisit la mise du ponte à condition qu'elle ne soit pas supérieure au capital et qu'elle ne soit pas nulle.

Lignes 400 à 690

On tire 6 cartes dans le sabot, J correspondant à la valeur de la carte et C à sa couleur. On incrémente le tableau J de 1; celui-ci ne doit jamais dépasser 8 puisqu'il n'y a que 8 jeux de 52 cartes.

Aux lignes 600 on affiche les cartes du ponte.

Lignes 790 à 980

Le choix du ponte est saisi au clavier seulement si la valeur de ses cartes atteint 5. Dans les autres cas, on suit les règles exposées dans l'introduction de ce chapitre.

Lignes 1000 à 1240

Le banquier tire ou non une carte suivant les données du tableau T\$ (voir explications des lignes 30 à 260).

Lignes 1300 à 1680

Pour les résultats, il suffit de comparer les jeux du ponte et du banquier (variables T et TO). On les affiche à l'écran ainsi que les gains du ponte. Si ce dernier a toujours de l'argent, on continue la partie.

Dans le cas où le nombre de cartes déjà tirées dépasse 204, on remet le tableau J à zéro avant une nouvelle partie.

Lignes 2000 à 2050 (sous-programme)

Le message contenu dans la variable M\$ est affiché en bas de l'écran pendant un court instant.

IDENTIFICATION des VARIABLES

NOM SYMBOLIQUE	TYPE (*)	ORIGINE			LONG.	OBSERVATIONS
		Clavier	Programme	Cassette		
A ₁	VARIABLE	X				Position du message
C	VARIABLE	X				Nombre aléatoire donnant la couleur
C \$	VARIABLE	X				Liste des couleurs
G ₁	COMPTEUR	X				Gains du pont
I	INDICE	X				Boucle FOR/NEXT
J	VARIABLE	X				Nombre aléatoire donnant le numéro de la carte dans le jeu
J (,)	TABLEAU	X				Cartes du jeu : 13 cartes x 4 couleur
J \$	VARIABLE	X				Liste des cartes du jeu
M	VARIABLE	X				Mise du pont
M \$	VARIABLE	X				Message sur l'écran
N C	COMPTEUR	X				Nombre de cartes utilisées
P (,)	TABLEAU	X				Valeur des cartes de la partie
P \$ (,)	TABLEAU	X				Cartes de la partie sous forme alphanumérique
R \$	VARIABLE	X				Réponse du joueur
T	VARIABLE	X				Total des cartes du pont
T 0	VARIABLE	X				Total des cartes du banquier (ordinateur)
T \$	VARIABLE	X				Liste des choix du banquier :
						T = conseille de tirer une carte
						R = incite à rester à deux cartes

6/1/1980

(*) Légende : aiguillage, compteur, constante, index, indice, tableau, variable. Reproduction interdite (c) Editions du P.S

Générateur de caractères

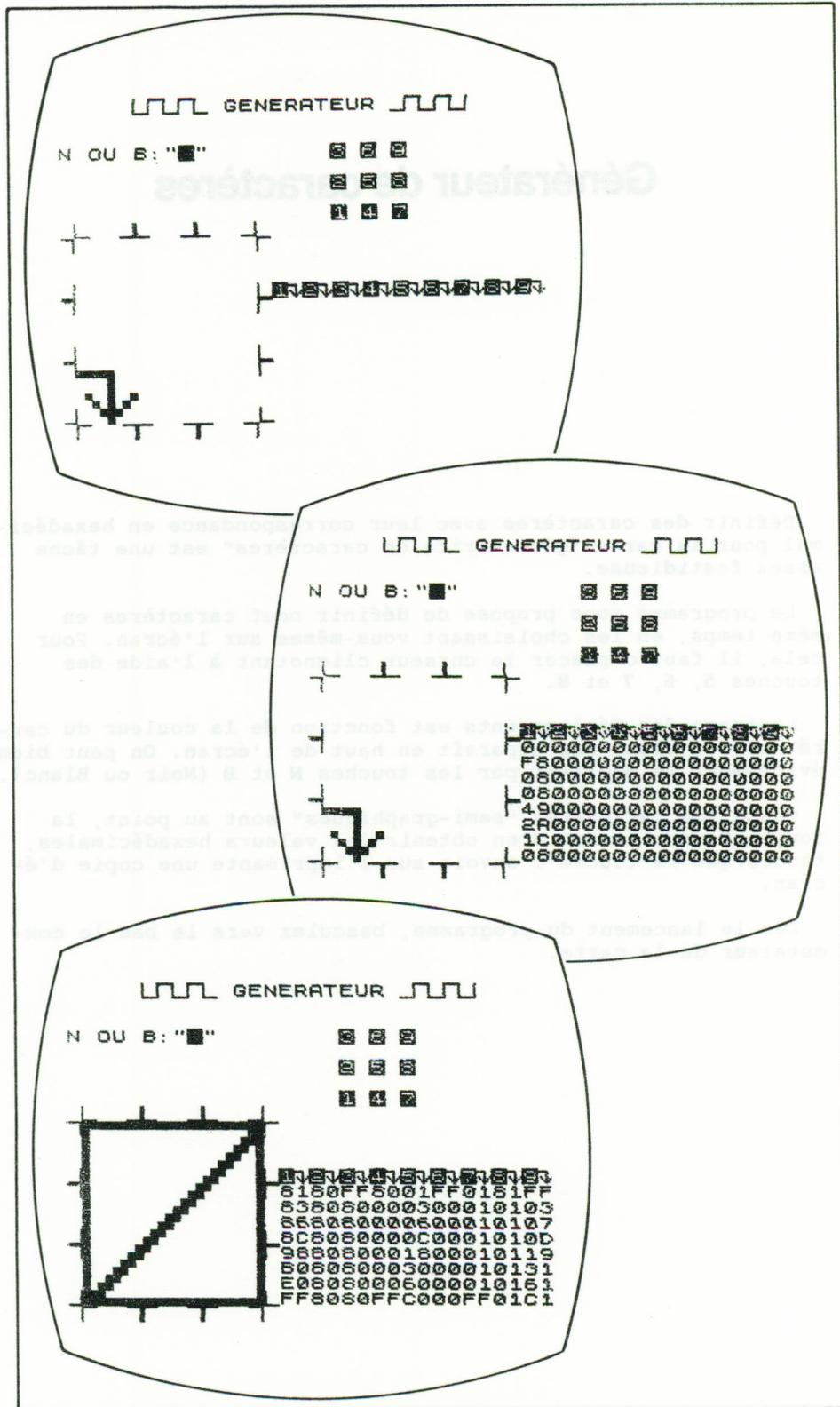
Définir des caractères avec leur correspondance en hexadécimal pour la carte "génératrice de caractères" est une tâche assez fastidieuse.

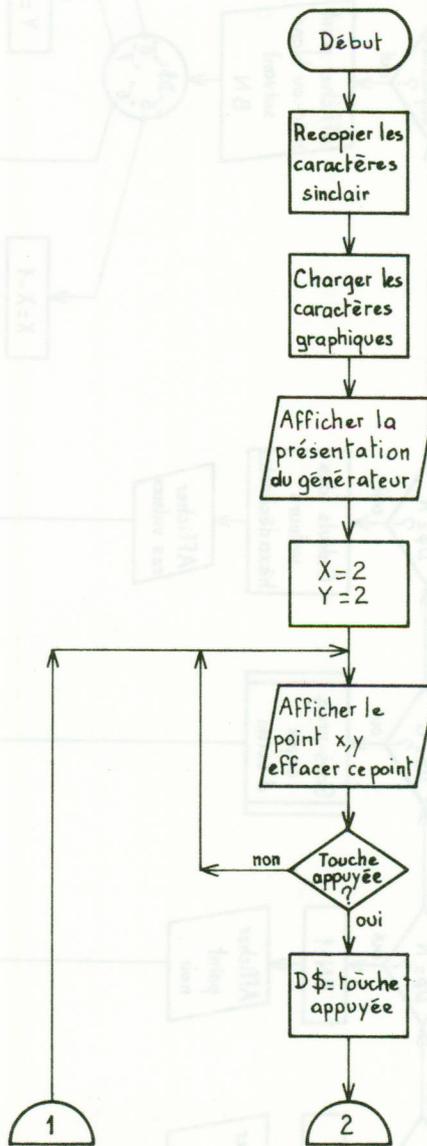
Le programme vous propose de définir neuf caractères en même temps, en les choisissant vous-mêmes sur l'écran. Pour cela, il faut déplacer le curseur clignotant à l'aide des touches 5, 6, 7 et 8.

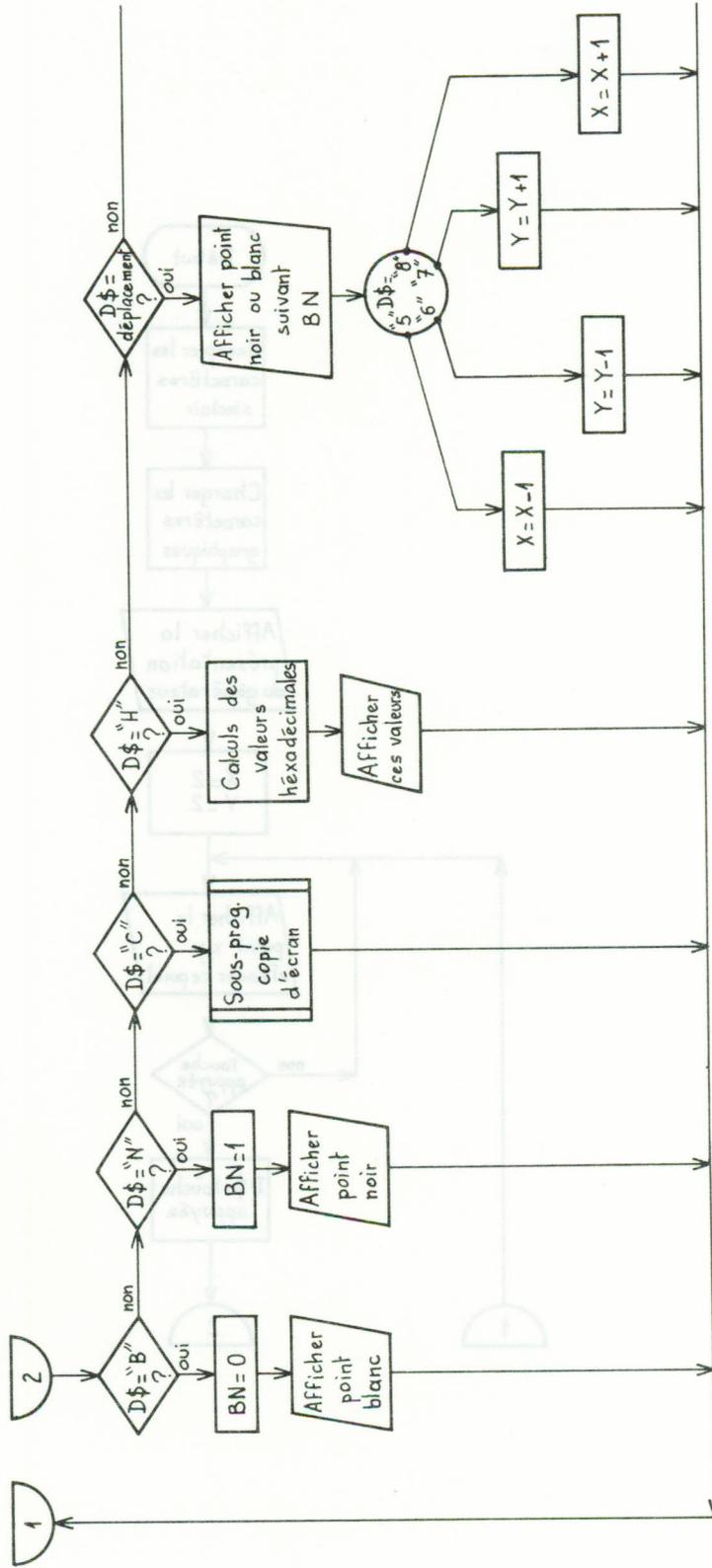
La trace des déplacements est fonction de la couleur du carré noir ou blanc qui apparaît en haut de l'écran. On peut bien évidemment le modifier par les touches **N** et **B** (Noir ou Blanc).

Quand vos caractères "semi-graphiques" sont au point, la touche **H** vous permet d'en obtenir les valeurs hexadécimales, tandis que la touche **C** envoie sur l'imprimante une copie d'écran.

Dès le lancement du programme, basculez vers le bas le commutateur de la carte.








```

980 PLOT X,Y
990 UNPLOT X,Y
1000 IF INKEY#="" THEN GOTO 980
1010 LET D#=INKEY#
1015 IF D#="B" THEN GOTO 1600
1016 IF D#="N" THEN GOTO 1700
1017 IF D#="C" THEN GOTO 1800
1018 IF D#="H" THEN GOTO 1900
1020 IF CODE D#<33 OR CODE D#>36
  THEN GOTO 1000
1030 LET D=VAL D#
1040 GOTO 1500
1050 IF X-1<2 THEN GOTO 1000
1060 LET X=X-1
1070 GOTO 980
1100 IF Y-1<2 THEN GOTO 1000
1110 LET Y=Y-1
1120 GOTO 980
1150 IF Y+1>25 THEN GOTO 1000
1160 LET Y=Y+1
1170 GOTO 980
1200 IF X+1>25 THEN GOTO 1000
1210 LET X=X+1
1220 GOTO 980
1500 IF BN=0 THEN GOTO 1530
1510 PLOT X,Y
1515 LET A#(Y-1,X-1)="1"
1520 GOTO 1540
1530 UNPLOT X,Y
1535 LET A#(Y-1,X-1)=" "
1540 GOTO 1000+(D-4)*50
1600 LET BN=0
1610 PRINT AT 3,8;" "
1620 GOTO 980
1700 LET BN=1
1710 PRINT AT 3,8;"■"
1720 GOTO 980
1800 LET L=USR QCOPY
1810 GOTO 980
1900 FOR I=1 TO 24
1910 DIM R(6)
1920 LET Y1=INT (I/8.5)*8
1930 LET X1=INT (I/8.5)*2
1940 FOR J=1 TO 4
1950 FOR K=0 TO 5
1960 IF A#(I,J+K*4)=" " THEN GOT
0 1980
1970 LET R(K+1)=R(K+1)+2*(4-J)
1980 NEXT K
1990 NEXT J
2000 FOR K=1 TO 5 STEP 2
2010 PRINT AT 21-I+Y1,X1+11+K*3;
H#(R(K)+1)
2020 PRINT AT 21-I+Y1,X1+12+K*3;
H#(R(K+1)+1)
2030 NEXT K
2040 NEXT I
2050 GOTO 980

```

LE PROGRAMME

Lignes 1 à 14

Il s'agit du programme "CHR\$" livré en même temps que la carte génératrice de caractères. Vous devez charger ce programme et écrire les autres lignes à la suite.

Lignes 20 à 80

On recopie tous les caractères du ZX dans la mémoire de la carte, puis l'on charge les caractères graphiques définis dans les lignes 40 à 75.

Lignes 100 à 540

On affiche à l'écran la présentation du générateur en utilisant les caractères semi-graphiques définis ci-dessus. La chaîne H\$ permettra ultérieurement les correspondances en hexadécimal.

Lignes 980 à 1020

On affiche puis on efface le point où se trouve le curseur (coordonnées x et y) pour l'effet de clignotement. On teste le clavier pour connaître la commande suivante. Si aucune touche n'est appuyée, on revient à la ligne 980 pour que l'effet de clignotement soit continu. On choisit le programme entre les lignes 1015 à 1020 suivant la valeur de la touche appuyée.

Lignes 1030 à 1540

Suivant le déplacement demandé, on ajoute ou on retranche un aux coordonnées x et y, ceci après avoir affiché un point noir ou blanc suivant la valeur de BN (lignes 1500 à 1540).

Lignes 1600 à 1720

On modifie la valeur de BN suivant la touche enfoncée.

Lignes 1800 à 1810

C'est la commande COPY. On utilise un sous-programme en assembleur pour obtenir sur l'imprimante les caractères semi-graphiques.

Lignes 1900 à 2050

On calcule puis on affiche toutes les valeurs en hexadécimal suivant le contenu du tableau A\$.

IDENTIFICATION des VARIABLES

NOM SYMBOLIQUE	TYPE (*)	ORIGINE				LONG.	OBSERVATIONS
		Clavier	Programme	Cassette	Disquette		
A, \$ ()	TABLEAU	X					Contenu de l'écran
B, N	VARIABLE	X					Noir ou blanc ?
C, L, R	CONSTANTE	X					Point d'entrée pour l'effacement du générateur de caractères
D, \$	VARIABLE	X					Commande du déplacement du curseur
H, \$	VARIABLE	X					Contient la liste des caractères hexadécimaux
I	INDICE	X					Boucle FOR/NEXT
J	INDICE	X					Boucle FOR/NEXT
K	INDICE	X					Boucle FOR/NEXT
L	VARIABLE	X					Variable d'appel des sous-programmes du générateur de caractères
Q, C, O, P, Y	CONSTANTE	X					Point d'entrée de recopie d'écran
Q, L, O, A, D	CONSTANTE						Point d'entrée de chargement d'un caractère
Q, P, R, N, T	CONSTANTE	X					Point d'entrée identique à LPRINT
R, ()	TABLEAU	X					Calcul des valeurs hexadécimales
S, I, N, C	CONSTANTE	X					Point d'entrée pour recopier le générateur de caractères du ZX81
X	VARIABLE	X					Position du curseur
X 1	VARIABLE	X					Position de la prochaine valeur hexadécimale

6/1980

(*) Légende : aiguillage, compteur, constante, index, indice, tableau, variable. Reproduction interdite © Editions du F

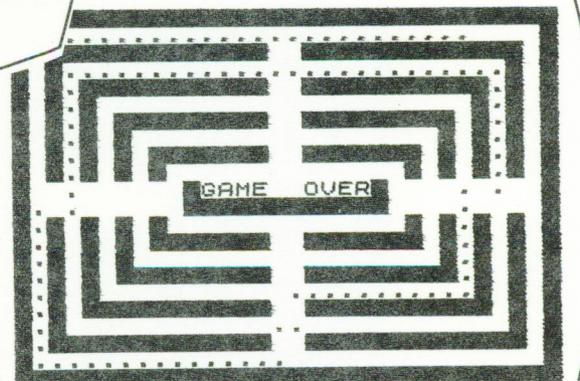
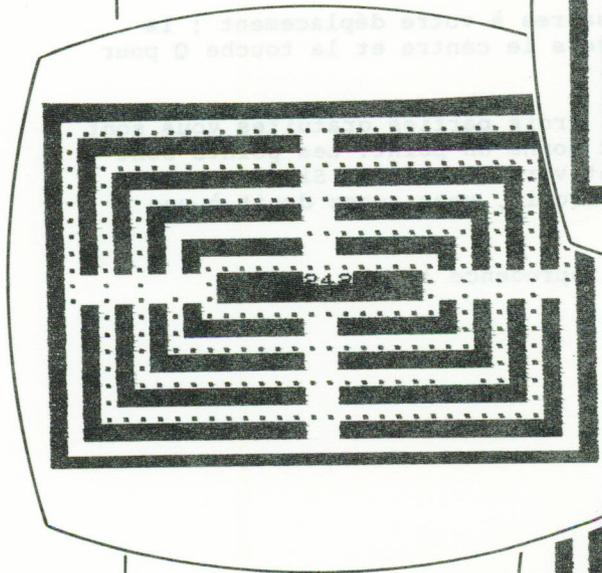
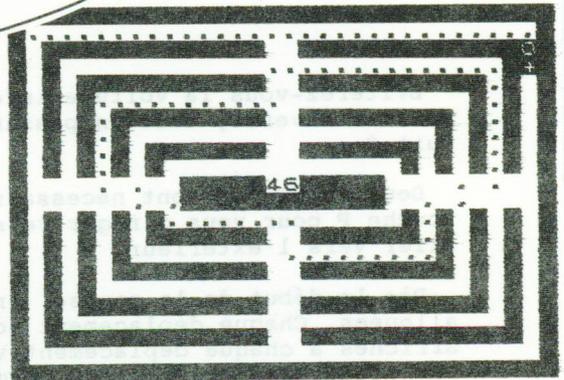
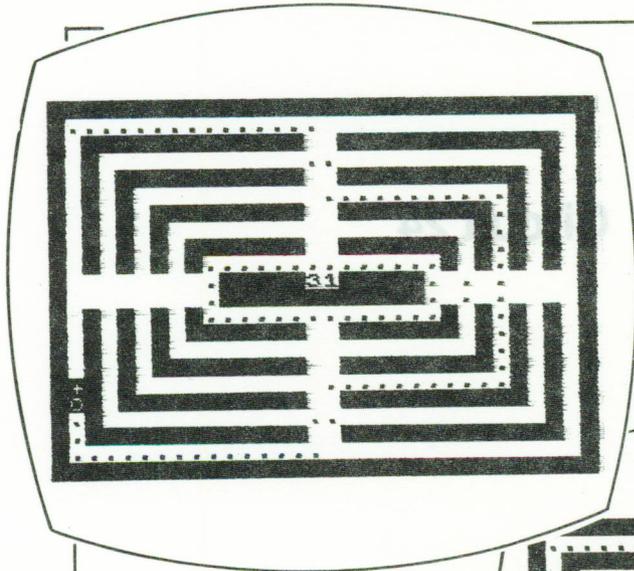
Circuit 24

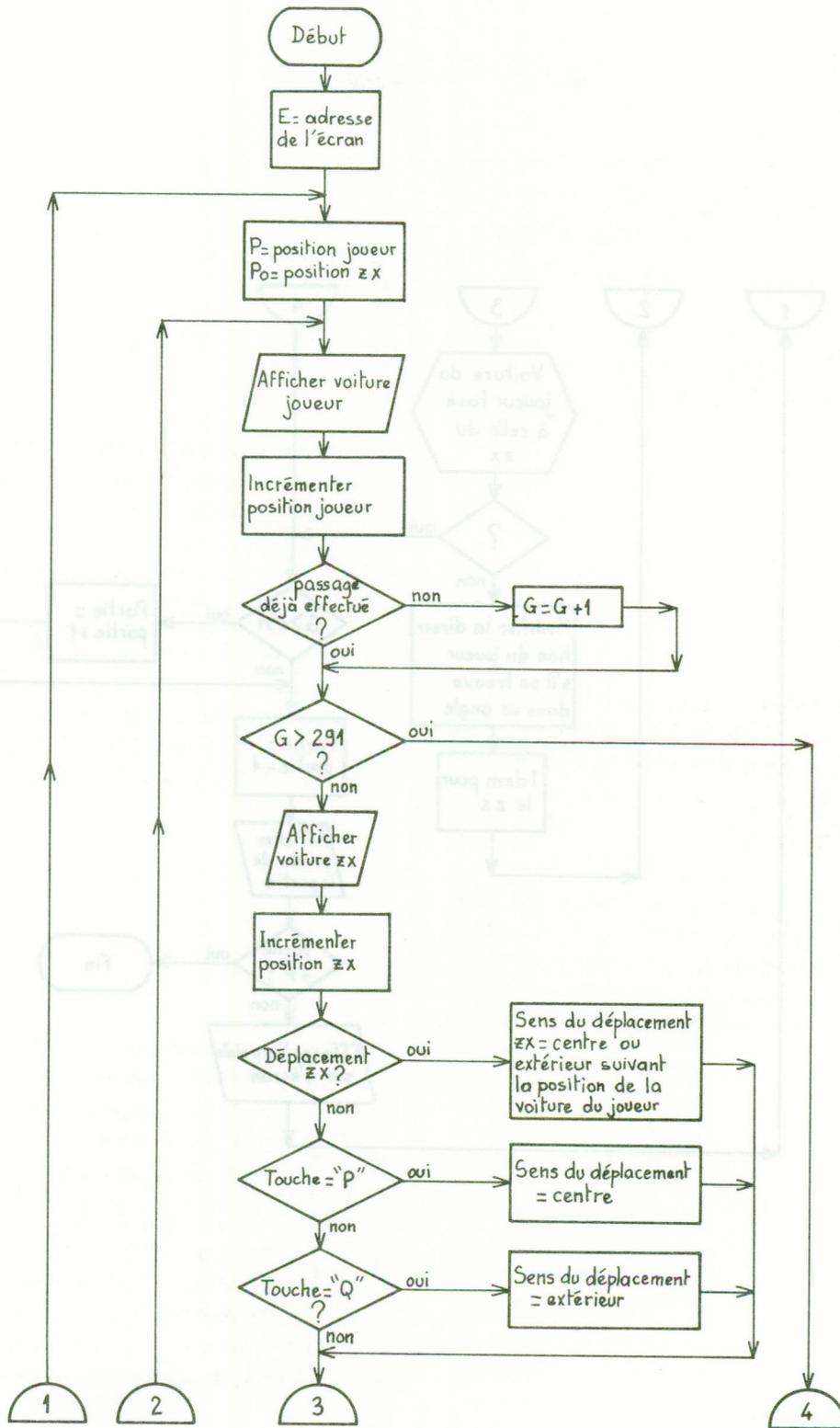
Évitez-vous la voiture infernale de l'ordinateur qui roule en sens inverse, tout en passant sur chaque portion du circuit ?

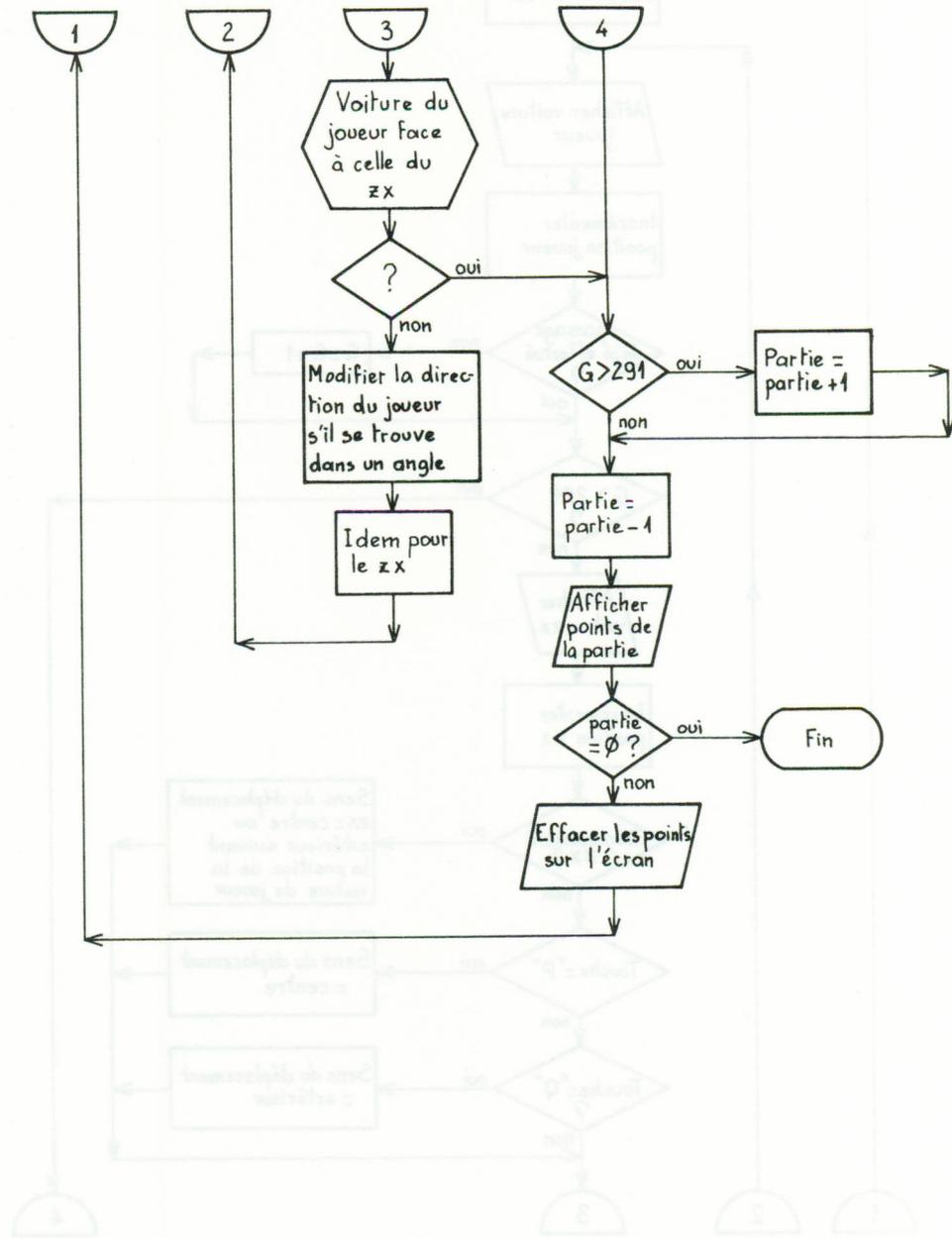
Deux commandes sont nécessaires à votre déplacement ; la touche P pour vous diriger vers le centre et la touche Q pour aller vers l'extérieur.

Dès le début de la course, trois parties gratuites vous sont allouées. Chaque déplacement donne un point. Les points sont affichés à chaque déplacement vers le centre. Si vous passez sur toutes les portions du circuit, vous aurez droit à une partie gratuite.

Bonne chance ... et pas d'imprudence !







```

10 REM CIRCUIT 24
20 REM -----
30 LET E=PEEK 16398+256*PEEK 1
6399
40 CLS
50 GOSUB 4000
60 IF INKEY#="" THEN GOTO 60
70 LET PG=3
80 LET TP=0
170 REM -----
180 REM INITIALISATION
190 LET P=E+34
200 LET PO=P+557
210 DIM T(6)
220 LET T(1)=-33
230 LET T(2)=1
240 LET T(3)=33
250 LET T(4)=-1
260 LET T(5)=-33
270 LET T(6)=1
280 LET G=0
300 LET T=1
310 LET TO=-33
320 LET A1=0
350 LET C=2
360 LET CO=5
370 LET L=5
380 LET LO=5
390 LET S=0
395 LET A=R1
397 REM -----
399 REM LE JEU ...
400 POKE P,27
410 LET P=P+T
415 IF PEEK (P)=0 THEN LET G=G+
1
416 IF G>291 THEN GOTO 3000
420 POKE P,180
430 POKE PO,A
435 LET PO=PO+TO
440 LET A1=PEEK (PO)
450 POKE PO,149
460 IF L<LO THEN LET S=-1
470 IF L>LO THEN LET S=1
490 IF S<>0 AND PEEK (PO+T(CO+S
))=0 AND (PEEK (PO+T(CO))=0 OR P
EEK (PO+TO)=27) THEN GOTO 2000
500 IF INKEY#="P" THEN GOTO 100
0
510 IF INKEY#="Q" THEN GOTO 150
0
580 IF PEEK (P+T)=149 THEN GOTO
3000
600 IF PEEK (P+T)=0 OR PEEK (P+
T)=27 THEN GOTO 635
610 LET C=C+1
620 IF C=6 THEN LET C=2
630 LET T=T(C)
635 IF PEEK (PO+TO)=0 OR PEEK (
PO+TO)=27 THEN GOTO 395
640 LET CO=CO-1
650 IF CO=1 THEN LET CO=5
660 LET TO=T(CO)
670 GOTO 400
960 REM -----
990 REM DEPLACEMENT CENTRE
1000 IF PEEK (P+T(C+1))<>0 THEN
GOTO 580
1010 POKE P,27
1020 LET P=P+T(C+1)+T(C+1)
1030 LET L=L-1
1040 IF L<1 THEN LET L=1
1045 PRINT AT 10,15;G
1050 GOTO 580

```

```

1480 REM -----
1490 REM DEPLACEMENT EXTERIEUR
1500 IF PEEK (P+T(C-1))<>0 THEN
GOTO 580
1510 POKE P,27
1520 LET P=P+T(C-1)+T(C-1)
1530 LET L=L+1
1540 GOTO 580
1980 REM -----
1990 REM DEPLACEMENT ZX
2000 POKE PO,A1
2010 LET PO=PO+T(CO+S)+T(CO+S)
2020 LET LO=LO+S
2030 LET S=0
2040 LET A1=PEEK (PO)
2050 GOTO 500
2980 REM -----
2990 REM FIN DE PARTIE
3000 POKE P,0
3010 POKE PO,0
3020 LET P=E+34
3030 LET PO=P+557
3040 IF G>291 THEN LET PG=PG+1
3050 LET PG=PG-1
3060 LET TP=TP+G
3070 PRINT AT 10,14;"██████"
3080 PRINT AT 21,31;"■"
3090 POKE 16442,4
3100 PRINT "PART.GRATIS:";PG;"
PTS:";TP
3110 IF PG=0 THEN GOTO 5000
3500 FOR I=1 TO 9 STEP 2
3510 FOR J=I TO 31-I
3520 POKE E+J+(I*33),0
3530 POKE E+693+J-(I*33),0
3540 NEXT J
3550 FOR K=I+1 TO 20-I
3560 POKE E+(K*33)+I,0
3570 POKE E+31+(K*33)-I,0
3580 NEXT K
3590 NEXT I
3600 GOTO 280
3980 REM -----
3990 REM DESSIN DU JEU
4000 FOR I=0 TO 8 STEP 2
4010 FOR J=I TO 31-I
4020 POKE E+J+(I*33),128
4030 POKE E+693+J-(I*33),128
4040 NEXT J
4050 FOR K=I+1 TO 20-I
4060 POKE E+(K*33)+I,128
4070 POKE E+31+(K*33)-I,128
4080 NEXT K
4090 NEXT I
4100 FOR I=2 TO 29
4105 IF I>19 THEN GOTO 4110
4106 POKE E+15+(I*33),0
4107 POKE E+16+(I*33),0
4110 IF I>9 AND I<22 THEN GOTO 4
140
4120 PRINT AT 10,I;" ";AT 11,I;"
"
4130 GOTO 4150
4140 PRINT AT 10,I;"■";AT 11,I;"
■"
4150 NEXT I
4160 RETURN
4980 REM -----
4990 REM FIN DU JEU
5000 FOR I=1 TO 60
5010 PRINT AT 10,11;"GAME OVER"
5020 PRINT AT 10,11;" "
5030 NEXT I
5040 GOTO 40

```

LE PROGRAMME

Lignes 300 à 395 :

La variable E contient après calcul l'adresse de début d'écran (voir le programme "piranhas" page 73)

Le tableau T correspond aux déplacements des voitures.

Une ligne contient 33 caractères (32 + NEWLINE). Pour remonter d'une ligne, il faut donc retrancher 33 à la position actuelle ; pour descendre d'une ligne, il faut lui ajouter 33. De la même manière, en ajoutant ou en retranchant 1, on avance ou on recule d'une case.

Lignes 400 à 670

Les variables I et O correspondent aux prochains déplacements du joueur et de l'ordinateur.

Au début du jeu, le joueur va vers la droite, donc T = 1, alors que l'ordinateur monte, donc TO = 33.

Le principe de déplacement des voitures est simple ; il suffit de tester le caractère qui se trouve devant chaque voiture. S'il est égal à 0 ou à un point, on peut avancer. Sinon on doit changer de direction.

Le circuit étant défini au départ, les voitures ne peuvent jamais sortir de l'écran.

Le reste du programme s'explique en consultant ensemble l'organigramme et la liste des variables.

INDICE	DESCRIPTION	TYPE
1	Direction du déplacement	TABLÉAU
2	Total des points	COMPTEUR
3	Valeur du déplacement du joueur	VARIABLE
4	Valeur du déplacement de l'X	VARIABLE
5	Différence entre le joueur et le X	VARIABLE
6	Position du X	VARIABLE
7	Parties gratuites	COMPTEUR
8	Position du joueur	VARIABLE
9	en plusieurs fois	VARIABLE
10	Compteur IX pour les déplacements	VARIABLE
11	déplacements en plusieurs fois	VARIABLE
12	Compteur du joueur pour les	VARIABLE
13	boucle FOR/NEXT	VARIABLE
14	INDICE	X

IDENTIFICATION des VARIABLES

NOM SYMBOLIQUE	TYPE (*)	ORIGINE				LONG.	OBSERVATIONS
		Clavier	Programme	Cassette	Disquette		
A	VARIABLE	X					Contient le caractère qui effacera la voiture du ZX au prochain déplacement
A 1	VARIABLE	X					Caractère sur l'écran avant déplacement du ZX
C	VARIABLE	X					Sens du déplacement du joueur
C 0	VARIABLE	X					Sens du déplacement du ZX
E	CONSTANTE	X					Début du fichier écran
G	COMPTEUR	X					Nombre de points dans une partie
I	INDICE	X					Boucle FOR/NEXT
J	INDICE	X					Boucle FOR/NEXT
K	INDICE	X					Boucle FOR/NEXT
L	VARIABLE	X					Compteur du joueur pour les déplacements en plusieurs fois
L 0	VARIABLE	X					Compteur ZX pour les déplacements en plusieurs fois
P	VARIABLE	X					Position du joueur
P G	COMPTEUR	X					Parties gratuites
P 0	VARIABLE	X					Position du ZX
S	VARIABLE	X					Différence entre le joueur et le ZX
T	VARIABLE	X					Valeur du déplacement du joueur
T 0	VARIABLE	X					Valeur du déplacement du ZX
T P	COMPTEUR	X					Total des points
T ()	TABLEAU	X					Direction du déplacement

6/1983

(*) Légende : aiguillage, compteur, constante, index, indice, tableau, variable. Reproduction interdite (c) Editions du P.S.I

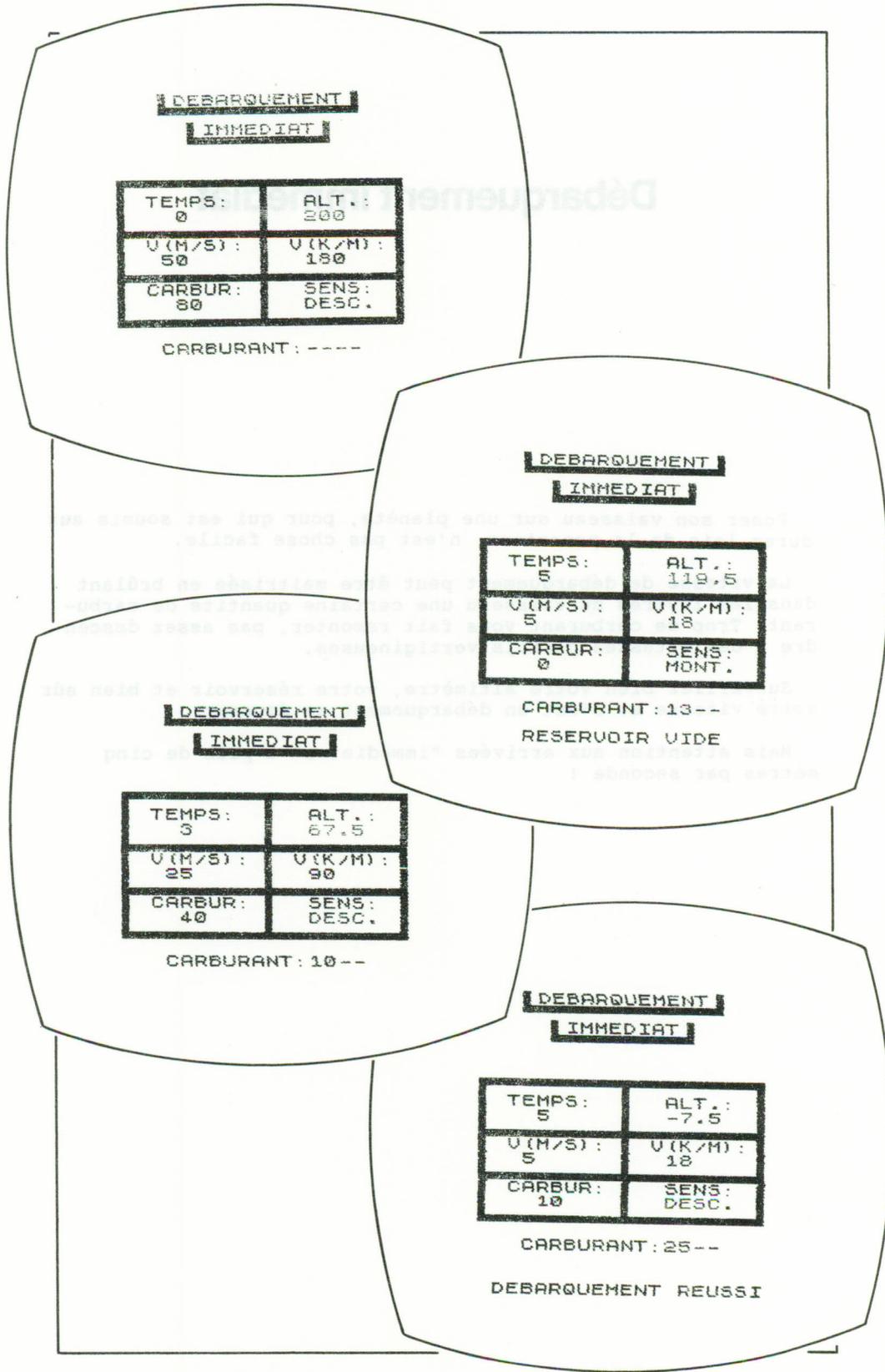
Débarquement immédiat

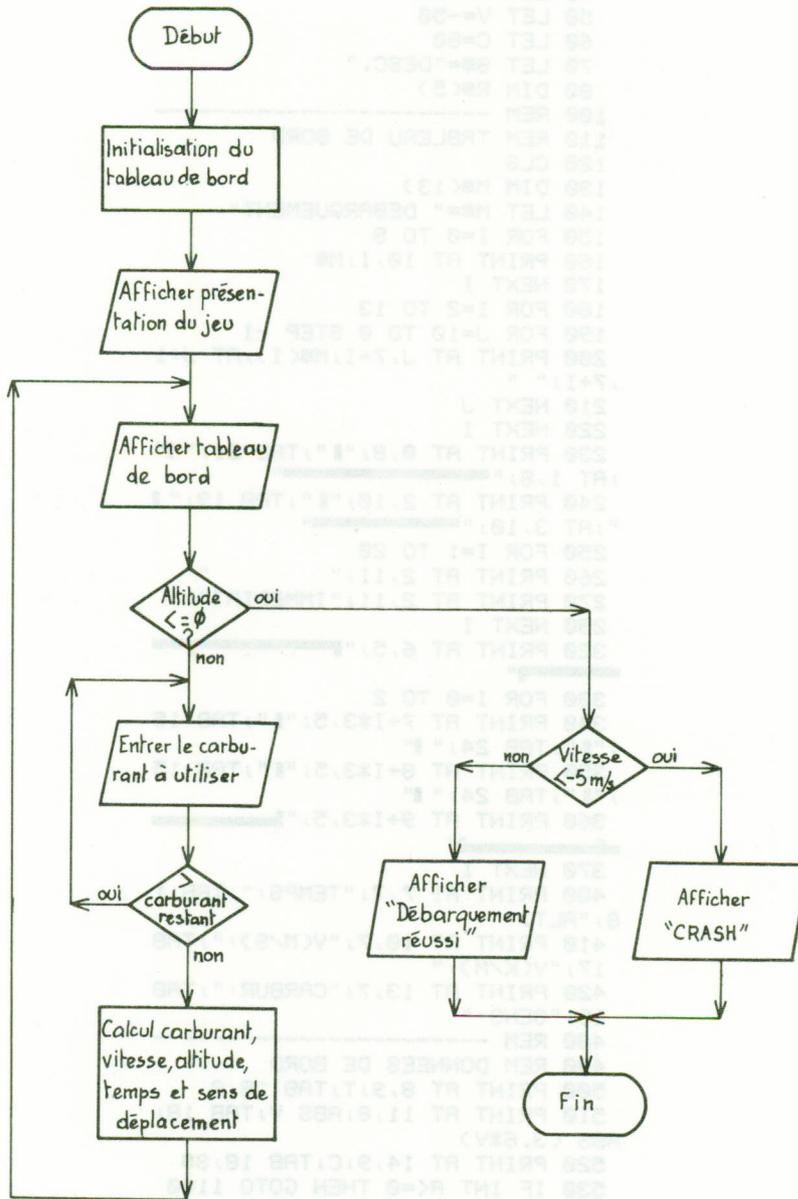
Poser son vaisseau sur une planète, pour qui est soumis aux dures lois de la pesanteur, n'est pas chose facile.

La vitesse de débarquement peut être maîtrisée en brûlant dans les tuyères du vaisseau une certaine quantité de carburant. Trop de carburant vous fait remonter, pas assez descendre à des vitesses parfois vertigineuses.

Surveillez bien votre altimètre, votre réservoir et bien sûr votre vitesse et c'est un débarquement en douceur.

Mais attention aux arrivées "immédiates" à plus de cinq mètres par seconde !





```

10 REM DEBARQUEMENT IMMEDIAT
20 REM -----
30 LET T=0
40 LET A=200
50 LET V=-50
60 LET C=80
70 LET S#="DESC."
80 DIM R#(5)
100 REM -----
110 REM TABLEAU DE BORD
120 CLS
130 DIM M#(13)
140 LET M#=" DEBARQUEMENT"
150 FOR I=0 TO 8
160 PRINT AT 10,I;M#
170 NEXT I
180 FOR I=2 TO 13
190 FOR J=10 TO 0 STEP -1
200 PRINT AT J,7+I;M#(I);AT J+1
,7+I;" "
210 NEXT J
220 NEXT I
230 PRINT AT 0,8;"|";TAB 21;"|"
;AT 1,8;"|-----|"
240 PRINT AT 2,10;"|";TAB 19;"|"
;AT 3,10;"|-----|"
250 FOR I=1 TO 20
260 PRINT AT 2,11;" "
270 PRINT AT 2,11;"IMMEDIAT"
280 NEXT I
320 PRINT AT 6,5;"|-----|"
|-----|"
330 FOR I=0 TO 2
340 PRINT AT 7+I*3,5;"|";TAB 15
;"|";TAB 24;"|"
350 PRINT AT 8+I*3,5;"|";TAB 15
;"|";TAB 24;"|"
360 PRINT AT 9+I*3,5;"|-----|"
|-----|"
370 NEXT I
400 PRINT AT 7,7;"TEMPS:";TAB 1
8;"ALT.:"
410 PRINT AT 10,7;"V(M/S):";TAB
17;"V(K/M):"
420 PRINT AT 13,7;"CARBUR:";TAB
18;"SENS:"
480 REM -----
490 REM DONNEES DE BORD
500 PRINT AT 8,9;T;TAB 18;A
510 PRINT AT 11,8;ABS V;TAB 18;
ABS (3.6*V)
520 PRINT AT 14,9;C;TAB 18;S#
530 IF INT A<=0 THEN GOTO 1100
540 IF C>0 THEN GOTO 570
550 LET R=0
560 GOTO 950
570 PRINT AT 17,8;"CARBURANT:--
--";

```

```

580 REM -----
590 REM CARBURANT A UTILISER
600 LET P=1
610 LET R#=""
620 LET C#=INKEY#
630 IF C#="" THEN GOTO 620
640 IF CODE C#=118 AND R#<>"
   " THEN GOTO 900
650 IF CODE C#=119 OR CODE C#=1
14 THEN GOTO 800
660 IF P>4 THEN GOTO 620
670 IF CODE C#<27 OR CODE C#>37
AND C#<>". " THEN GOTO 620
680 PRINT AT 17,17+P;C#;
690 LET R#(P TO P)=C#
700 LET P=P+1
710 GOTO 620
800 IF P=1 THEN GOTO 620
810 LET P=P-1
820 PRINT AT 17,17+P;"-";AT 17,
17+P;
830 LET R#(P TO P)=" "
840 GOTO 620
880 REM -----
890 REM CALCUL
900 LET R=VAL R#
910 IF R>C THEN GOTO 570
920 IF R>25 THEN LET R=25
930 LET C=C-R
940 IF C=0 THEN PRINT AT 19,8;"
RESERVOIR VIDE"
950 IF C=0 THEN PAUSE 200
960 LET T=T+1
970 LET A=A+V-((5-R)/2)
980 LET V=V-(5-R)
990 PRINT AT 08,8;"      ";TAB
18;"      "
1000 PRINT AT 11,8;"      ";TAB
18;"      "
1010 PRINT AT 14,8;"      ";TAB
18;"      "
1020 IF V>=0 THEN LET S#="MONT."
1030 IF V<0 THEN LET S#="DESC."
1040 GOTO 500
1080 REM -----
1090 REM FIN DE PARTIE
1100 PRINT AT 19,6;"
"
1110 IF V<-5 THEN GOTO 1200
1120 FOR I=1 TO 40
1130 PRINT AT 20,6;"DEBARQUEMENT
REUSSI"
1140 PRINT AT 20,6;"
"
1150 NEXT I
1160 GOTO 30
1180 REM -----
1190 REM CRASH..
1200 CLS
1210 FOR I=1 TO 100
1220 PLOT RND*38+10,RND*18+10
1230 PRINT AT 11,12;"CRASH";AT 1
1,12;"      "
1240 NEXT I
1250 GOTO 30

```

LE PROGRAMME

Lignes 30 à 80

On initialise dans ces lignes toutes les variables du tableau de bord (temps, altitude, vitesse, carburant et sens du déplacement).

Lignes 120 à 420

Pour la présentation du jeu, on déplace le mot "DEBARQUEMENT" sur l'écran (lignes 140 à 220), puis l'on fait clignoter vingt fois le mot "IMMEDIAT". Le reste du programme dessine le tableau de bord.

Lignes 490 à 570

On affiche dans ces lignes les valeurs du tableau de bord. Dans le cas où l'altitude est inférieure à 0, on saute directement à la fin du jeu.

Lignes 600 à 840

Le carburant à utiliser est saisi caractère par caractère avec vérification. Cette saisie remplace une fonction INPUT avec l'avantage du choix d'emplacement (voir GESTION DE FICHER du TOME I d'Etudes pour ZX81).

Lignes 900 à 1040

Cette partie du programme calcule les valeurs du nouveau tableau de bord. Les anciennes valeurs sont effacées sur l'écran pour éviter toute erreur dans l'affichage.

Lignes 1100 à 1250

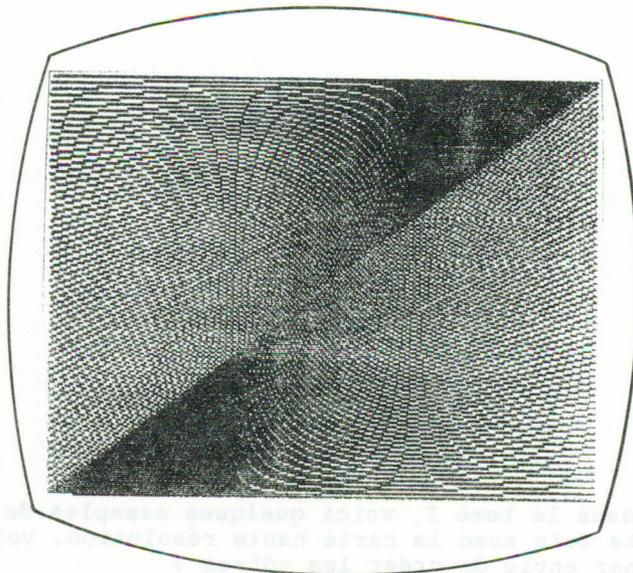
Suivant la vitesse du vaisseau, on affiche un des deux messages : "DEBARQUEMENT REUSSI" ou "CRASH".

Graphiques

Comme dans le tome I, voici quelques exemples de graphiques, mais cette fois avec la carte haute résolution. Voilà de quoi vous donner envie de créer les vôtres !

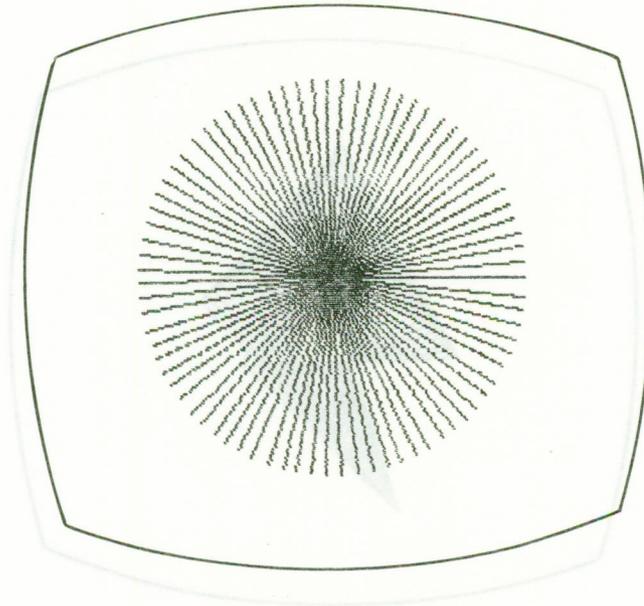
Les résultats obtenus sont meilleurs sur l'écran que les copies de ce même écran données comme illustrations des programmes.

```
10 REM GRAPHICS
11 REM
12 LET C=100000
13 LET L=USR C
14 REM BL CL
15 LET A=0
16 LET B=101
17 LET X=200
18 LET Y=101
19 LET C=2000
20 LET D=0
21 LET V=0
22 LET W=0
23 FOR I=1 TO 64
24 LET L=USR C
25 REM HD C D DR V W
26 LET W=W+3
27 NEXT I
28 IF INKEY="" THEN GOTO 400
29 IF INKEY="P" THEN GOTO 410
410 LET L=USR C
420 REM COPY
430 FOR K=0
```



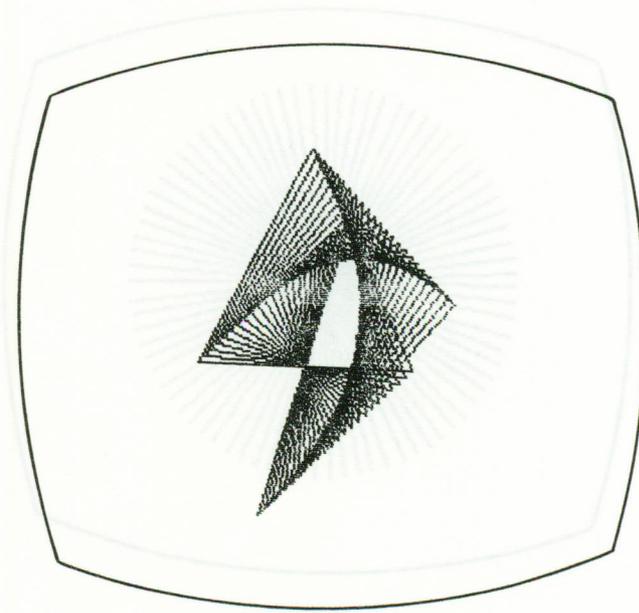
```

10 REM GRAPHIQUES
20 REM -----
30 LET G=10240
40 LET L=USR G
50 REM BL CL
60 LET A=0
70 LET B=191
80 LET X=255
90 LET Y=191
100 LET C=255
110 LET D=0
120 LET V=0
130 LET W=0
200 FOR I=1 TO 64
210 LET L=USR G
220 REM MO A B DR X Y
230 LET Y=Y-3
240 LET L=USR G
250 REM MO C D DR V W
260 LET W=W+3
300 NEXT I
400 IF INKEY#="" THEN GOTO 400
410 IF INKEY#(">") THEN GOTO 44
0
420 LET L=USR G
430 REM COPY
440 POKE G,0
    
```



```

10 REM GRAPHIQUES
20 REM -----
30 LET G=10240
40 LET L=USR G
50 REM BL CL
60 LET A=255/2
70 LET B=191/2
200 FOR I=0 TO 234 STEP 3
210 LET I1=I/(2*PI)
220 LET X=COS (I1)*90+A
230 LET Y=SIN (I1)*90+B
240 LET L=USR G
250 REM MO A B DR X Y
300 NEXT I
400 IF INKEY#="" THEN GOTO 400
410 IF INKEY#("<")="P" THEN GOTO 44
0
420 LET L=USR G
430 REM COPY
440 POKE G,0
    
```



```

10 REM GRAPHIQUES
20 REM -----
30 LET G=10240
40 LET L=USR G
50 REM BL CL
200 FOR I=0 TO 70 STEP 3
210 LET J=I+70
215 LET K=100-J
220 LET X=120+I
230 LET Y=I
240 LET V=170-I
250 LET W=100+I
280 LET L=USR G
290 REM MO J K DR X Y DR V W DR
J K
300 NEXT I
400 IF INKEY#="" THEN GOTO 400
410 IF INKEY#<>"P" THEN GOTO 44
0
420 LET L=USR G
430 REM COPY
440 POKE G,0
    
```

Taquin

Le taquin ou puzzle à 15 a été inventé par Sam LLOYD aux Etats-Unis dans les années 1880. Il se compose d'un plateau et de quinze plaquettes numérotées de 1 à 15 que l'on dispose en désordre (plus de vingt mille milliards de possibilités). Le vide sur le plateau permet de bouger les plaquettes pour les remettre dans leur ordre croissant.

Lancez le programme en tapant RUN. Après quelques instants - le temps de mélanger le taquin - le plateau s'affiche à l'écran.

Seules les plaquettes voisines de la case vide peuvent se déplacer.

Il ne s'agit pas du jeu original de Sam LLOYD. En conséquence, toutes les positions de départ permettent une reconstitution de l'ensemble du jeu.

--= TAQUIN =--

11	2	13	4
5	12	14	1
6	3	10	9
15	7	8	

VOUS DEPLACEZ QUEL NUMERO -->

--= TAQUIN =--

1	2	3	4
12	15		11
6	13	14	5
7	10	9	8

VOUS DEPLACEZ QUEL NUMERO -->

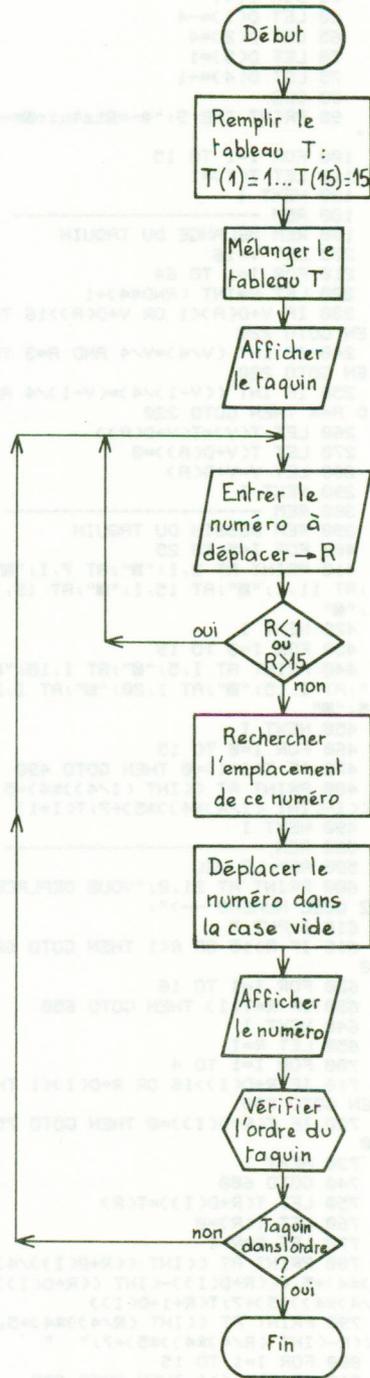
--= TAQUIN =--

1	2	3	4
5	6	7	8
12		10	9
15	13	11	14

VOUS DEPLACEZ QUEL NUMERO -->

--= TAQUIN =--

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	



```

10 REM TAQUIN
20 REM -----
30 DIM T(16)
40 DIM R$(2)
50 DIM D(4)
60 LET D(1)=-4
65 LET D(2)=4
70 LET D(3)=1
75 LET D(4)=-1
80 CLS
90 PRINT TAB 9;"#--■taquin■--#"
"
100 FOR I=1 TO 15
110 LET T(I)=I
120 NEXT I
180 REM -----
190 REM MELANGE DU TAQUIN
200 LET V=16
210 FOR I=1 TO 64
220 LET A=INT (RND*4)+1
230 IF V+D(A)<1 OR V+D(A)>16 TH
EN GOTO 220
240 IF INT (V/4)=V/4 AND A=3 TH
EN GOTO 220
250 IF INT ((V-1)/4)=(V-1)/4 AN
D A=4 THEN GOTO 220
260 LET T(V)=T(V+D(A))
270 LET T(V+D(A))=0
280 LET V=V+D(A)
290 NEXT I
380 REM -----
390 REM DESSIN DU TAQUIN
400 FOR I=5 TO 25
410 PRINT AT 3,I;"■";AT 7,I;"■"
;AT 11,I;"■";AT 15,I;"■";AT 19,I
;"■"
420 NEXT I
430 FOR I=3 TO 19
440 PRINT AT I,5;"■";AT I,10;"■"
;AT I,15;"■";AT I,20;"■";AT I,2
5;"■"
450 NEXT I
460 FOR I=0 TO 15
470 IF T(I+1)=0 THEN GOTO 490
480 PRINT AT ((INT (I/4))*4)+5,
((I-(INT (I/4))*4))*5+7;T(I+1)
490 NEXT I
580 REM -----
590 REM LE JEU
600 PRINT AT 21,0;"VOUS DEPLACE
Z QUEL NUMERO -->";
610 INPUT R
615 IF R>15 OR R<1 THEN GOTO 60
0
620 FOR I=1 TO 16
630 IF R=T(I) THEN GOTO 650
640 NEXT I
650 LET R=I
700 FOR I=1 TO 4
710 IF R+D(I)>16 OR R+D(I)<1 TH
EN GOTO 730
720 IF T(R+D(I))=0 THEN GOTO 75
0
730 NEXT I
740 GOTO 600
750 LET T(R+D(I))=T(R)
760 LET T(R)=0
770 LET R=R-1
780 PRINT AT ((INT ((R+D(I))/4)
)*4)+5,((R+D(I))-(INT ((R+D(I)
)/4)*4))*5+7;T(R+D(I))
790 PRINT AT ((INT (R/4))*4)+5,
((R-(INT (R/4))*4))*5+7;" "
800 FOR I=1 TO 15
810 IF T(I)<>I THEN GOTO 600
820 NEXT I
900 PRINT AT 21,0;"O.K , VOUS A
VEZ TROUVE ... "
910 IF INKEY#="" THEN GOTO 910
920 RUN

```

LE PROGRAMME

Lignes 30 à 120

Le tableau T correspond aux seize cases du taquin. On y place donc les numéros de 1 à 15 (ligne 100). Le tableau D correspond au déplacement des plaquettes.

1	2	3	4	-4
5	6	7	8	↑
9	10	11	12	-1 ← X → +1
13	14	15	/	↓
				+4

Lignes 200 à 260

On mélange les numéros en permutant deux cases désignées par un nombre aléatoire et le pointeur de la case vide. Soixante quatre permutations suffisent à mélanger correctement le taquin.

Lignes 400 à 480

On affiche le jeu à l'écran dans le nouvel ordre (ou plutôt désordre !).

Lignes 600 à 790

On déplace le numéro tapé par le joueur en vérifiant que celui-ci est placé à côté de la case libre. Le déplacement est calculé avec le tableau D.

Lignes 800 à 820

A chaque essai, on vérifie qu'il reste des plaquettes dans le désordre. Si c'est le cas, on retourne à la ligne 600 pour un nouvel essai.

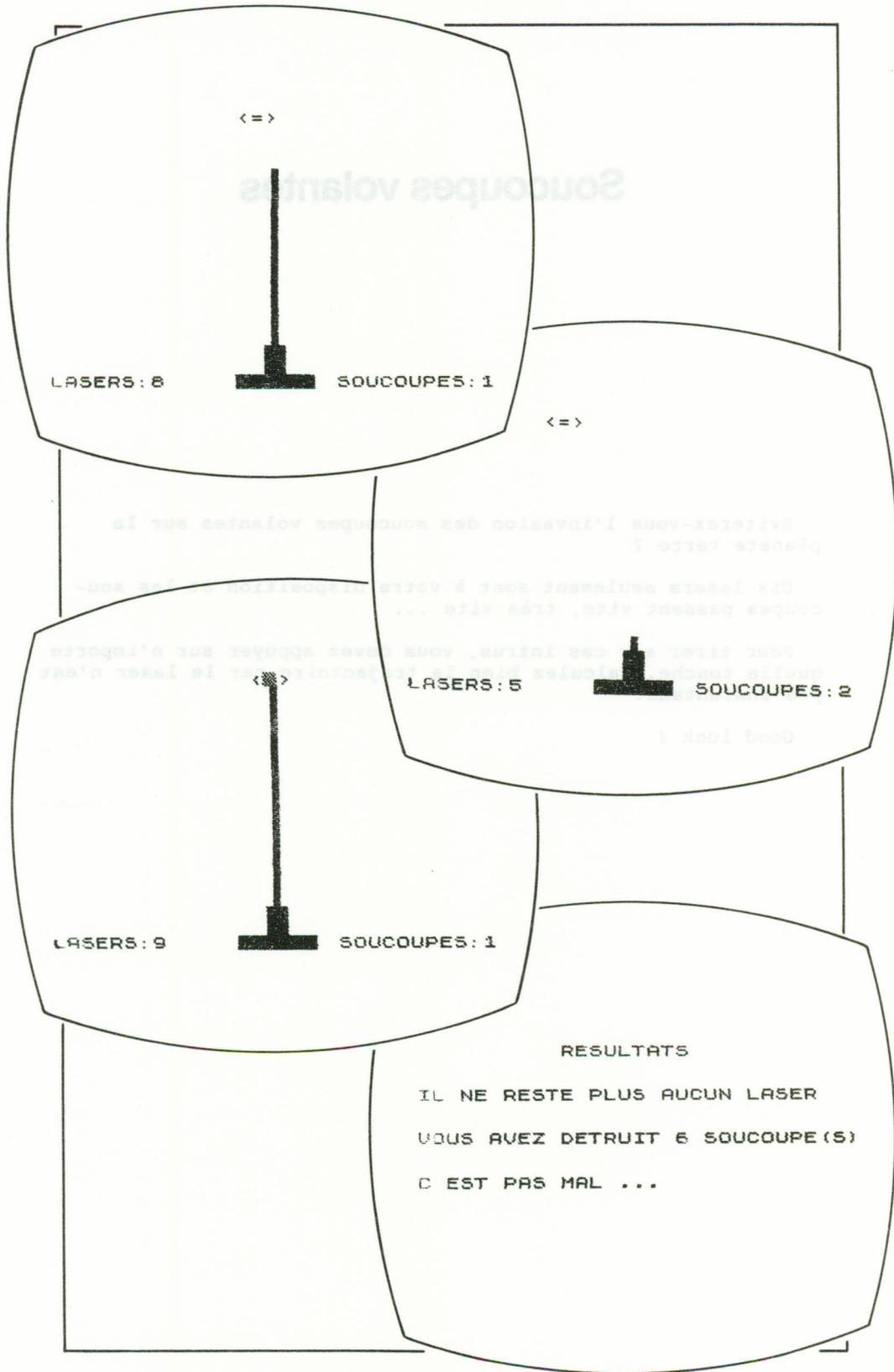
Soucoupes volantes

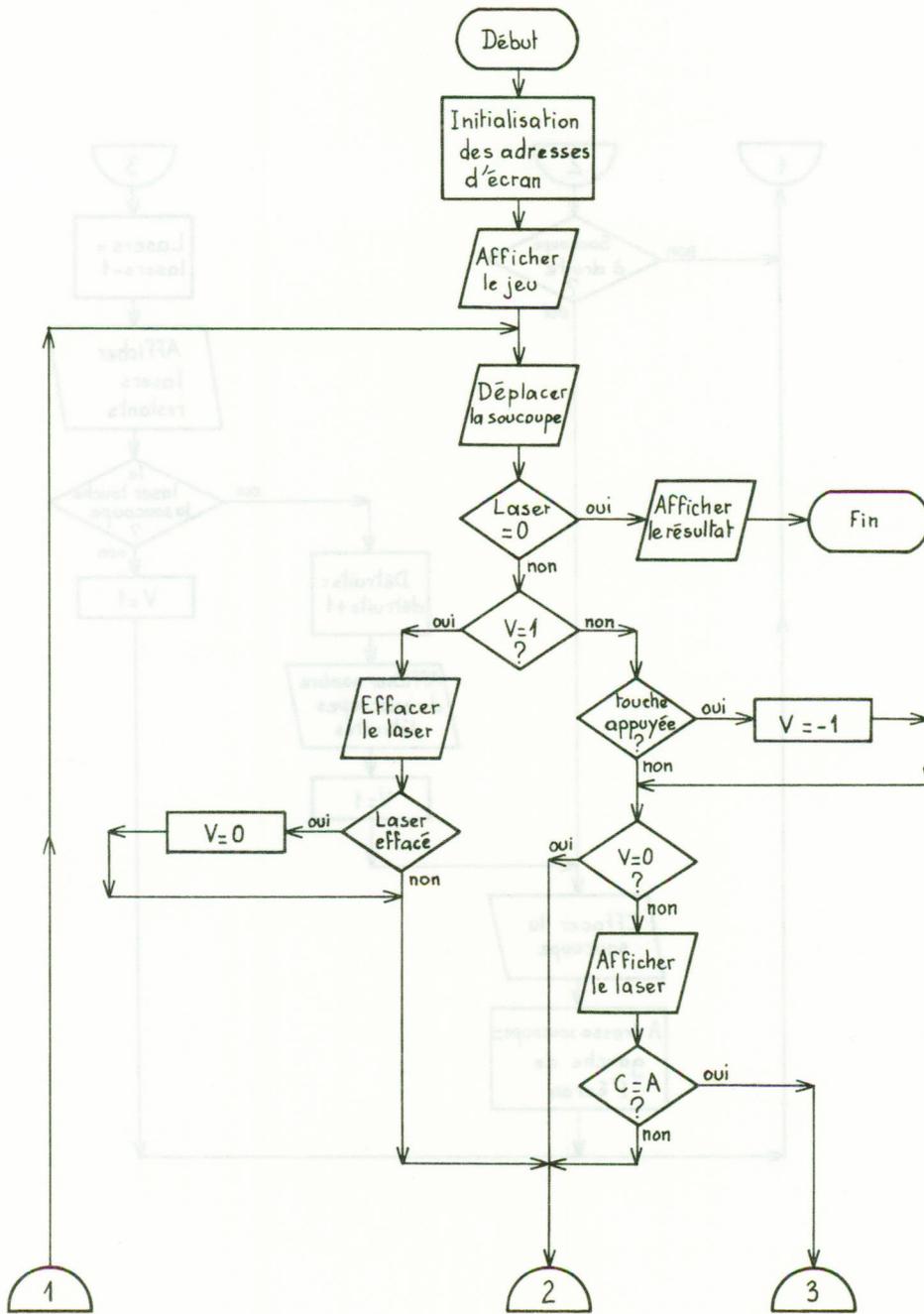
Eviterez-vous l'invasion des soucoupes volantes sur la planète terre ?

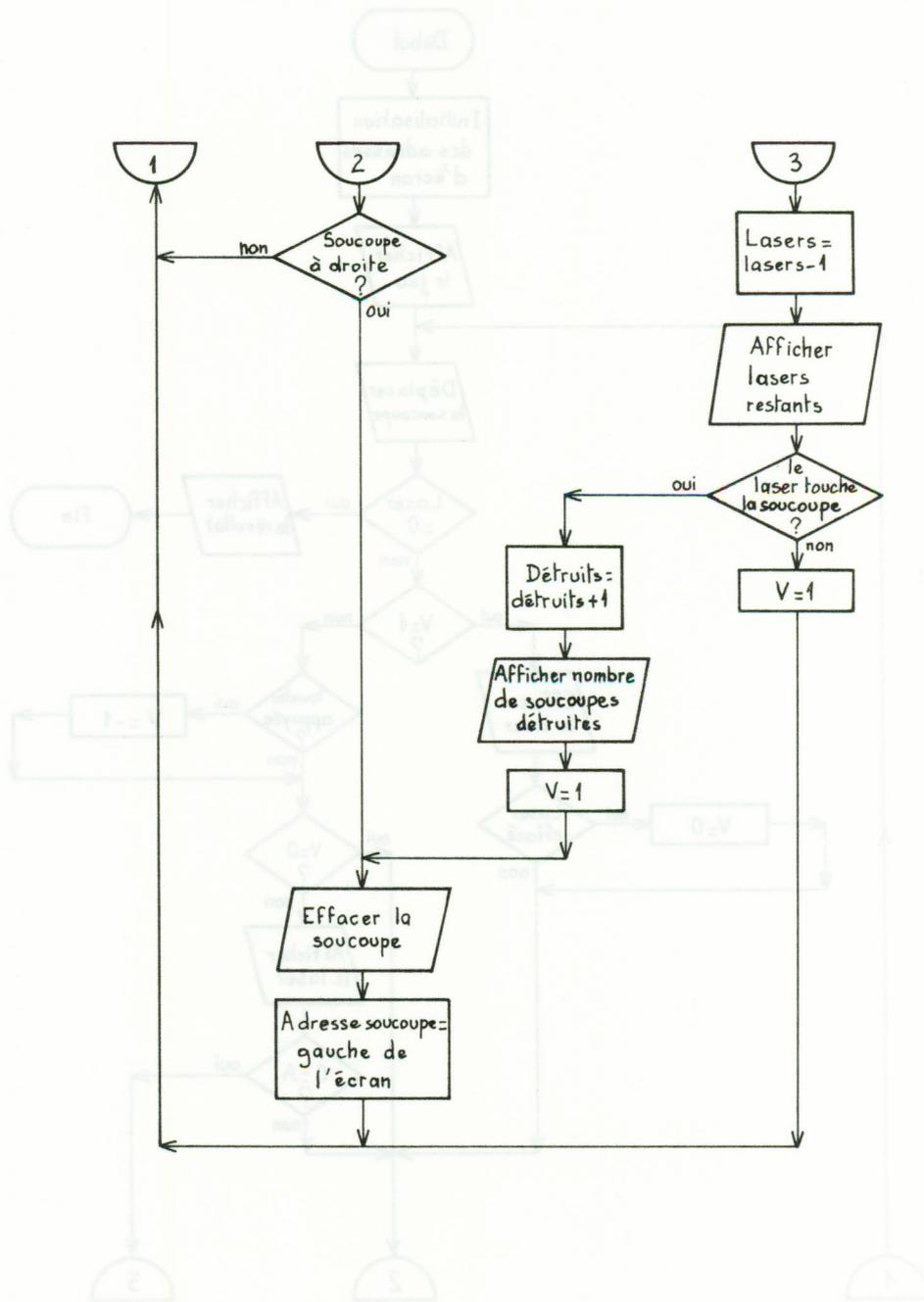
Dix lasers seulement sont à votre disposition et les soucoupes passent vite, très vite ...

Pour tirer sur ces intrus, vous devez appuyer sur n'importe quelle touche. Calculez bien la trajectoire car le laser n'est pas instantané.

Good luck !







ETUDES POUR ZX81

```

10 REM SOUCOUPES VOLANTES
20 REM -----
30 LET E=PEEK 16398+256*PEEK 1
6399
40 LET L=10
50 LET D=0
60 LET AL=E+(20*33)+7
70 LET AS=E+(20*33)+30
200 PRINT AT 20,13;" "
210 PRINT AT 19,15;" ";AT 18,15;" "
220 PRINT AT 20,0;"LASERS:"
230 PRINT AT 20,20;"SOUCOUPES:"
300 LET C=E+15+(17*33)
310 LET S=E+66
320 LET V=0
330 LET A=E+66+15
380 REM -----
390 REM LE JEU
400 POKE S,0
410 POKE S+1,19
420 POKE S+2,20
430 POKE S+3,18
440 LET S=S+1
445 IF L=0 THEN GOTO 900
450 IF V=1 THEN GOTO 530
460 IF INKEY#<>" " THEN LET V=-1
465 IF V=0 THEN GOTO 600
470 POKE C,133
480 POKE C-33,133
490 POKE C-66,133
500 LET C=C-(99)
510 IF C=A THEN GOTO 680
520 GOTO 600
530 POKE C,0
540 POKE C+33,0
550 POKE C+66,0
560 LET C=C+99
570 IF C<E+15+(17*33) THEN GOTO
600
580 LET V=0
600 IF S>E+66+27 THEN GOTO 620
610 GOTO 400
620 POKE S,0
630 POKE S+1,0
640 POKE S+2,0
650 LET S=E+66
660 GOTO 400
680 LET L=L-1
690 POKE AL,L+28
700 IF PEEK C=20 THEN GOTO 750
710 LET V=1
720 GOTO 400
750 LET D=D+1
760 POKE AS,D+28
770 LET V=1
780 FOR I=1 TO 30
790 POKE S+1,0
800 POKE S+1,0
810 NEXT I
820 GOTO 620
880 REM -----
890 REM FIN DU JEU
900 CLS
910 PRINT TAB 10;"RESULTATS"
920 PRINT
930 PRINT
940 PRINT "IL NE RESTE PLUS AUC
UN LASER"
950 PRINT
960 PRINT
970 PRINT "VOUS AVEZ DETRUIT ";
D;" SOUCOUCO(S)"
980 PRINT
990 PRINT
1000 IF D=10 THEN PRINT "VOUS ET
ES UN AS DE L ESPACE"
1010 IF D>=5 AND D<>10 THEN PRIN
T "C EST PAS MAL ..."
1020 IF D<5 THEN PRINT "JE PREFE
RE NE RIEN DIRE ..."
1030 IF INKEY#="" THEN GOTO 1030
1040 CLS
1050 RUN

```

LE PROGRAMME

Lignes 30 à 330.

La variable E contient l'adresse du fichier écran, ce qui permet d'écrire directement sur l'écran par l'instruction POKE. A partir de celle-ci, on place dans les variables S et C les adresses de départ de la soucoupe volante et du laser.

Lignes 400 à 820

La soucoupe, pointée par S, se déplace continuellement. Dès qu'elle arrive à droite, on l'efface et on replace dans S l'adresse de départ.

- La variable V peut prendre trois valeurs :
- V = 0 : le laser n'est pas déplacé.
 - V = -1 : on déplace le laser vers le haut.
 - V = 1 : on efface le laser de l'écran.

Dès que le laser arrive sur la ligne de la soucoupe, on teste le caractère qui s'y trouve (PEEK de la ligne 700). Dans le cas où ce caractère a le code 20, on incrémente le compteur des soucoupes détruites.

Facturation

Les heureux possesseurs du tome I d'"Etudes pour ZX81" reconnaîtront ici la gestion de fichier de la page 132. Elle a été remaniée en gestion d'articles et trois nouveaux modules ont été ajoutés.

La facturation : elle permet d'éditer une facture d'un nombre indéfini d'articles. Il s'agit de rentrer au clavier le code des articles et la quantité à facturer. Le programme vérifie le stock existant et refuse toute quantité supérieure à celui-ci. Le mot "FIN", à la place d'un code-article, calcule le montant hors-taxes, la T.V.A. et le montant toutes taxes.

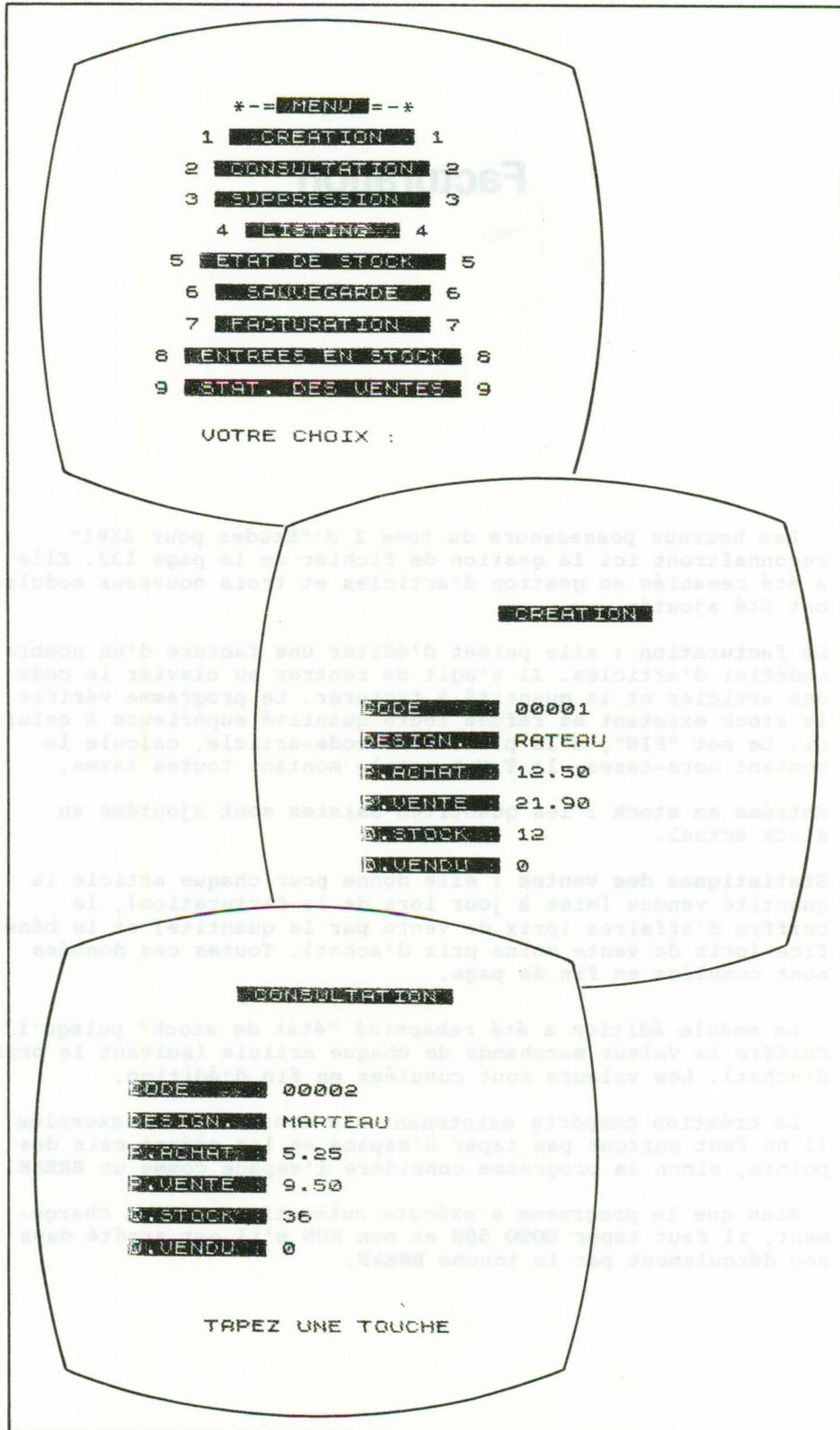
Entrées en stock : les quantités saisies sont ajoutées au stock actuel.

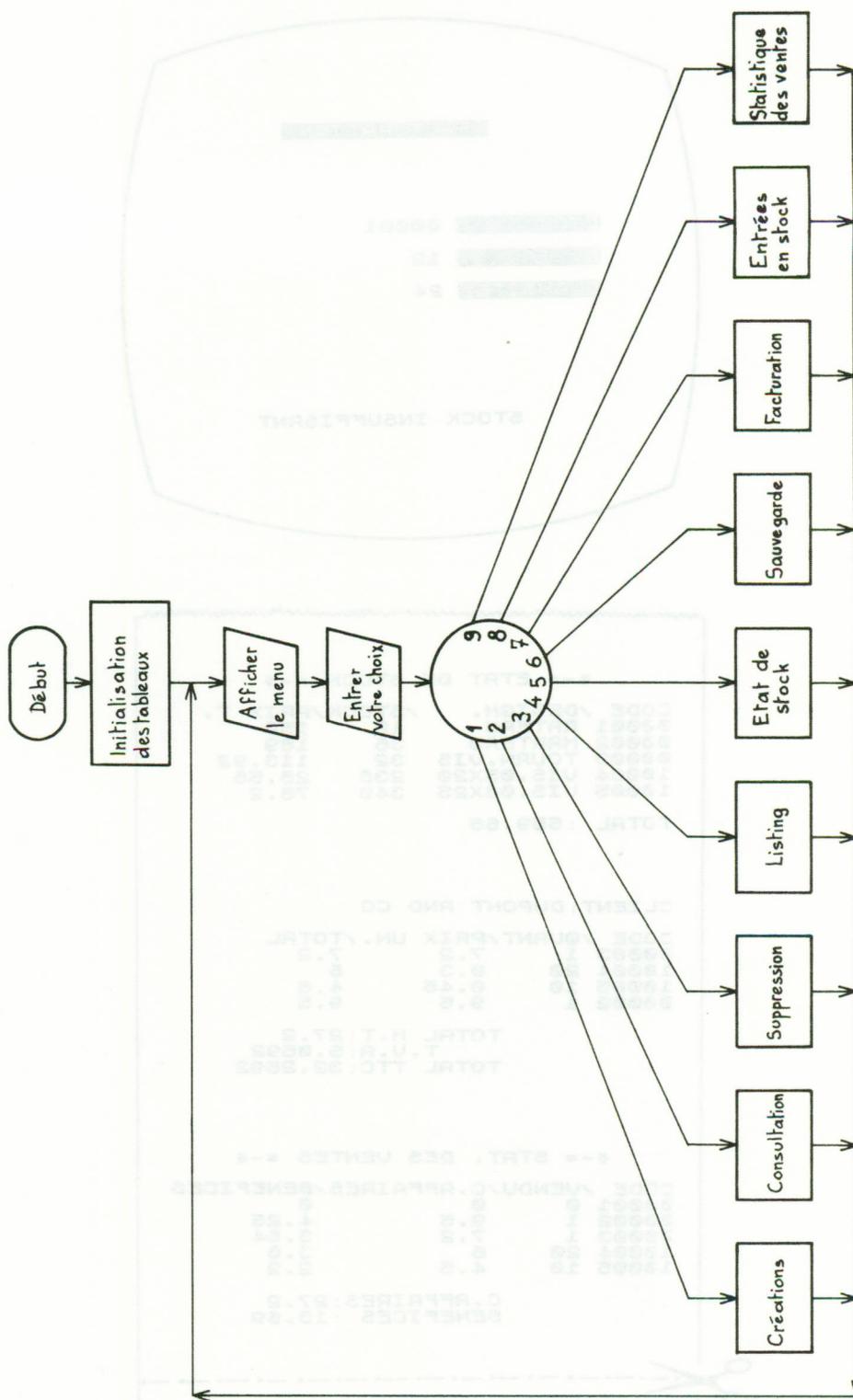
Statistiques des ventes : elle donne pour chaque article la quantité vendue (mise à jour lors de la facturation), le chiffre d'affaires (prix de vente par la quantité) et le bénéfice (prix de vente moins prix d'achat). Toutes ces données sont cumulées en fin de page.

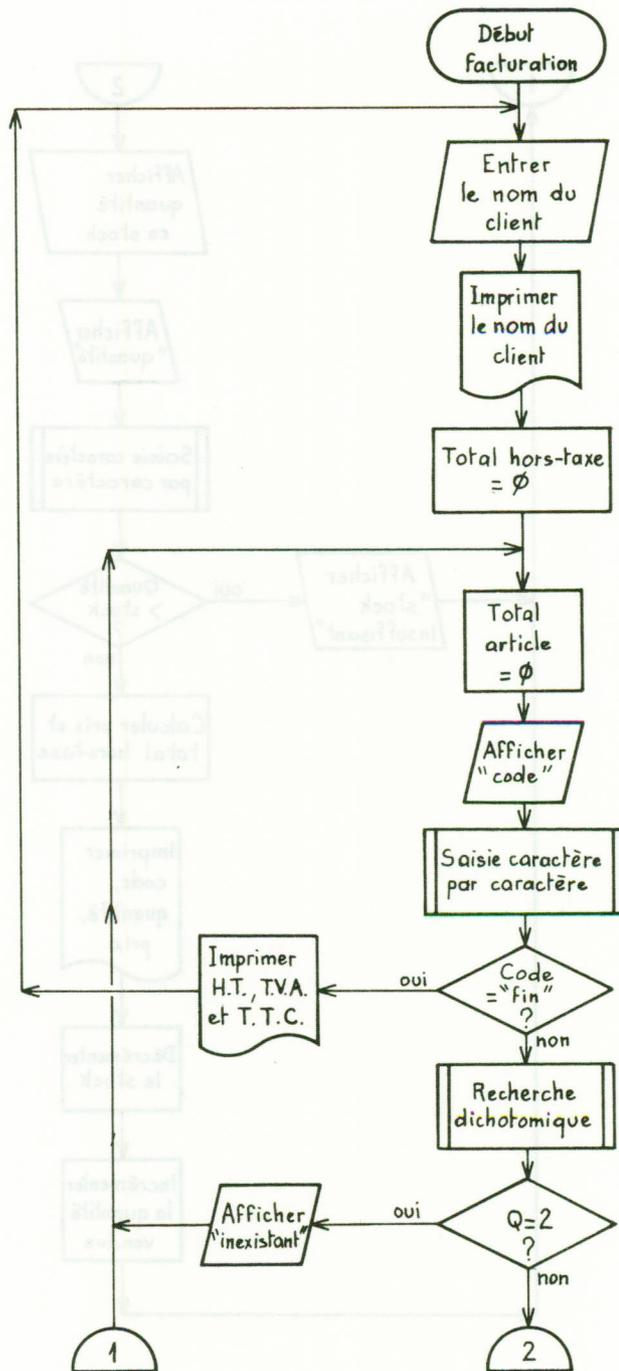
Le module édition a été rebaptisé "état de stock" puisqu'il chiffre la valeur marchande de chaque article (suivant le prix d'achat). Les valeurs sont cumulées en fin d'édition.

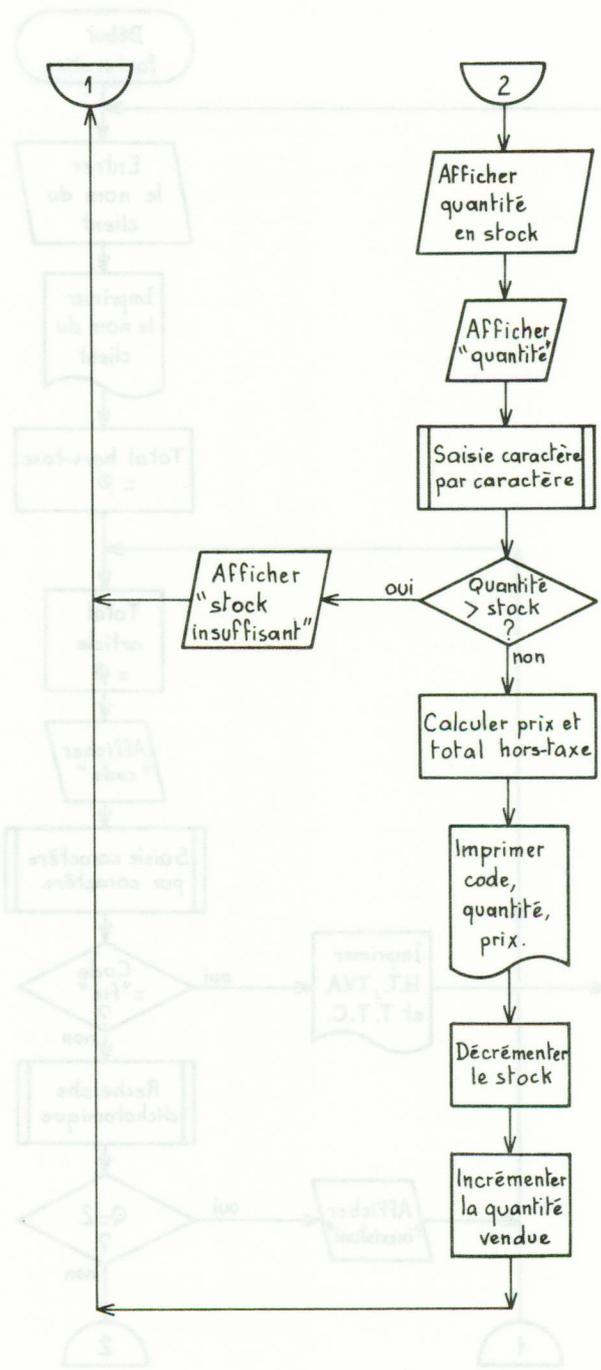
La création comporte maintenant six données (voir exemples). Il ne faut surtout pas taper d'espace en les créant mais des points, sinon le programme considère l'espace comme un **BREAK**.

Bien que le programme s'exécute automatiquement au chargement, il faut taper **GOTO 500** et non **RUN** s'il est arrêté dans son déroulement par la touche **BREAK**.









```

10 REM STOCK-FACTURATION
20 REM -----
30 DIM N$(100,5)
40 DIM D$(100,10)
50 DIM A$(100,9)
60 DIM P$(100,9)
70 DIM S$(100,5)
80 DIM V$(100,5)
90 DIM R$(5)
100 LET NBA=0
110 LET M=1
120 DIM Y$(10)
400 REM -----
490 REM MENU
500 CLS
510 PRINT TAB 10;"*--■menu■--*"
520 PRINT
540 PRINT TAB 8;"1 ■creation■"
550 PRINT
560 PRINT TAB 7;"2 ■consultatio
n■ 2"
570 PRINT
580 PRINT TAB 7;"3 ■suppression
■ 3"
590 PRINT
600 PRINT TAB 9;"4 ■listing■ 4"
610 PRINT
620 PRINT TAB 6;"5 ■etat■de■sto
ck■ 5"
630 PRINT
640 PRINT TAB 7;"6 ■sauvegarde
■ 6"
650 PRINT
660 PRINT TAB 7;"7 ■facturation
■ 7"
670 PRINT
680 PRINT TAB 5;"8 ■entrees■en■
stock■ 8"
690 PRINT
700 PRINT TAB 5;"9 ■stat.■des■v
entes■ 9"
760 PRINT
770 PRINT
780 PRINT TAB 8;"VOTRE CHOIX : "
790 IF INKEY#="" THEN GOTO 790
800 LET C=CODE INKEY#
820 IF C<29 OR C>37 THEN GOTO 7
90
830 LET C=C-27
840 IF C=10 THEN LET C=9.5
850 GOTO C*1000
980 REM -----
990 REM RECHERCHE DICHOTOMIQUE
1000 LET I=1
1010 LET S=NBA
1020 IF S<1 THEN GOTO 1110
1030 IF I>S THEN GOTO 1110
1040 LET M=INT ((I+S)/2)
1050 IF N$(M)=R# THEN GOTO 1130
1060 IF N$(M)>R# THEN GOTO 1090
1070 LET I=M+1
1080 GOTO 1020
1090 LET S=M-1
1100 GOTO 1020
1110 LET Q=2
1120 RETURN
1130 LET Q=1
1140 RETURN

```

```

1180 REM -----
1190 REM SAISIE CARACTERE PAR
      CARACTERE
1200 LET X=1
1201 LET Y#=""
1205 LET X#=INKEY#
1210 IF X#="" THEN GOTO 1205
1220 IF CODE X#=118 THEN GOTO 14
      00
1225 IF CODE X#=119 OR CODE X#=1
14 THEN GOTO 1300
1230 IF X=11 THEN GOTO 1205
1235 LET Y#(X TO X)=X#
1240 PRINT X#;
1250 LET X=X+1
1260 GOTO 1205
1300 IF X=1 THEN GOTO 1205
1310 LET X=X-1
1320 PRINT AT Y,10+X;" ";
1330 PRINT AT Y,10+X;
1340 GOTO 1205
1400 IF Y#="" " AND Y>8
THEN LET Y#=""
1410 RETURN
1980 REM -----
1990 REM CREATION
2000 CLS
2010 PRINT TAB 10;"creation";
2020 PRINT AT 6,1;"code";
2030 LET Y=6
2040 GOSUB 1200
2050 IF CODE Y#=0 OR Y#="" THEN
      GOTO 500
2060 LET R#=Y#
2100 GOSUB 1000
2110 IF Q=1 THEN GOTO 2700
2120 PRINT AT 21,10;"JE TRIE"
2130 IF NBA=0 THEN GOTO 2150
2140 IF N#(M)<R# THEN LET M=M+1
2150 LET NBA=NBA+1
2160 IF M=NBA THEN GOTO 2225
2200 FOR J=NBA TO M+1 STEP -1
2210 LET N#(J)=N#(J-1)
2212 LET P#(J)=P#(J-1)
2214 LET A#(J)=A#(J-1)
2216 LET S#(J)=S#(J-1)
2218 LET V#(J)=V#(J-1)
2219 LET V#(J)=V#(J-1)
2220 NEXT J
2225 PRINT AT 21,10;" "
2230 LET N#(M)=R#
2240 PRINT AT 8,1;"design. ";
2250 LET Y=8
2260 GOSUB 1200
2270 LET D#(M)=Y#
2280 PRINT AT 10,1;"p.achat ";
2290 LET Y=10
2300 GOSUB 1200
2310 LET A#(M)=Y#
2320 PRINT AT 12,1;"p.vente ";
2330 LET Y=12
2340 GOSUB 1200
2350 LET P#(M)=Y#
2360 PRINT AT 14,1;"q.stock ";
2370 LET Y=14
2380 GOSUB 1200
2390 LET S#(M)=Y#
2400 PRINT AT 16,1;"d.vendu ";
2410 LET Y=16
2420 GOSUB 1200
2430 LET V#(M)=Y#
2500 GOTO 2000
2700 PRINT AT 21,4;R#;" EXISTE D
      EJA"
2710 IF INKEY#="" THEN GOTO 2710
2720 GOTO 2000

```

```

2980 REM -----
2990 REM CONSULTATION
3000 CLS
3010 PRINT TAB 8;"consultation"
"
3020 PRINT AT 6,1;"code": ";
3030 LET Y=6
3035 GOSUB 1200
3036 IF CODE Y#=0 THEN GOTO 500
3040 LET R#=Y#
3045 GOSUB 1000
3050 IF Q=2 THEN GOTO 3300
3060 PRINT
3070 PRINT AT 8,1;"design.": ";D
#(M)
3080 PRINT
3090 PRINT AT 10,1;"p.achat": ";
A#(M)
3100 PRINT
3110 PRINT AT 12,1;"p.vente": ";
P#(M)
3120 PRINT
3130 PRINT AT 14,1;"q.stock": ";
S#(M)
3140 PRINT
3150 PRINT AT 16,1;"q.vendu": ";
V#(M)
3160 PRINT AT 21,6;"TAPEZ UNE TO
UCHE"
3170 IF INKEY#="" THEN GOTO 3170
3180 GOTO 3000
3300 PRINT AT 21,0;Y#;"INEXISTAN
T"
3310 IF INKEY#="" THEN GOTO 3310
3320 GOTO 3000
3980 REM -----
3990 REM SUPPRESSION
4000 CLS
4010 PRINT TAB 10;"suppression"
"
4020 PRINT AT 6,1;"code": ";
4030 LET Y=6
4040 GOSUB 1200
4050 IF CODE Y#=0 THEN GOTO 500
4060 LET R#=Y#
4070 GOSUB 1000
4080 IF Q=2 THEN GOTO 4300
4090 IF M=NBA THEN GOTO 4170
4100 FOR I=M TO NBA-1
4110 LET N#(I)=N#(I+1)
4120 LET D#(I)=D#(I+1)
4130 LET A#(I)=A#(I+1)
4140 LET P#(I)=P#(I+1)
4150 LET S#(I)=S#(I+1)
4160 LET V#(I)=V#(I+1)
4170 NEXT I
4180 LET NBA=NBA-1
4190 GOTO 4000
4300 PRINT AT 21,0;Y#;"INEXISTAN
T"
4310 IF INKEY#="" THEN GOTO 4310
4320 GOTO 4000

```

```

4980 REM -----
4990 REM LISTING
5000 CLS
5005 LET L=0
5010 FOR I=1 TO NBA
5020 LET L=L+1
5030 IF L<20 THEN GOTO 5050
5040 PRINT
5041 PRINT "TAPEZ UNE TOUCHE POU
R LA SUITE"
5042 IF INKEY#="" THEN GOTO 5042
5044 CLS
5046 LET L=0
5050 PRINT N$(I); " "; D$(I)
5060 NEXT I
5070 PRINT
5080 PRINT "TAPEZ UNE TOUCHE POU
R LE MENU"
5090 IF INKEY#="" THEN GOTO 5090
5100 GOTO 500
5980 REM -----
5990 REM EDITION
6000 CLS
6010 LPRINT TAB 5; "*-= ETAT DE S
TOCK =-*"
6012 LET T=0
6015 LPRINT
6020 LPRINT "CODE /DESIGN. /ST
OCK/PRIX T."
6030 FOR I=1 TO NBA
6040 LPRINT N$(I); " "; D$(I); " ";
S$(I); TAB 23; VAL S$(I)*VAL A$(I)
6050 LET T=T+(VAL S$(I)*VAL A$(I)
)
6060 NEXT I
6070 LPRINT
6080 LPRINT "TOTAL :"; T
6090 GOTO 500
6980 REM -----
6990 REM SAUVEGARDE
7000 CLS
7010 PRINT TAB 10; "■sauvegarde■"
7020 PRINT
7030 PRINT " QUAND VOTRE MAGNETO
PHONE"
7040 PRINT " EST EN POSITION ENR
EGISTREMENT"
7050 PRINT " TAPER SUR UNE TOUCH
E POUR"
7060 PRINT " LA SAUVEGARDE DE VO
TRE FICHIER"
7070 IF INKEY#="" THEN GOTO 7070
7080 CLS
7090 FOR I=1 TO 16
7100 PRINT TAB I; "TAPEZ ""GOTO 5
00""
7110 NEXT I
7120 PRINT
7130 PRINT "ET SURTOUT PAS ""RUN
""
7140 SAVE "FICHIER"
7150 GOTO 500

```

```

7980 REM -----
7990 REM FACTURATION
8000 CLS
8010 PRINT TAB 8;"Facturation"
8020 PRINT AT 21,0;"NOM DU CLIENT ?"
8030 INPUT C#
8040 IF C#="" THEN GOTO 500
8050 PRINT AT 21,0;"
"
8060 LPRINT "CLIENT: ";C#
8070 LPRINT
8080 LPRINT "CODE /QUANT/PRIX UN
./TOTAL"
8100 LET TH=0
8130 CLS
8140 PRINT TAB 8;"Facturation"
8150 PRINT AT 6,1;"code: ";
8160 LET TA=0
8165 LET Y=6
8170 GOSUB 1200
8180 IF Y#( TO 3)="FIN" THEN GOT
O 8400
8190 LET R#=Y#
8200 GOSUB 1000
8210 IF Q=2 THEN GOTO 8600
8215 PRINT AT 8,1;"q.stock: ";S
#(M)
8220 PRINT AT 10,1;"quantite: ";
8230 LET Y=10
8240 GOSUB 1200
8250 IF VAL Y#=0 THEN GOTO 8130
8255 IF VAL Y#>VAL S#(M) THEN GO
TO 8700
8260 LET TA=VAL P#(M)*VAL Y#
8270 LET TH=TH+TA
8280 LPRINT TAB 0;N#(M);TAB 6;VA
L Y#;TAB 12;VAL P#(M);TAB 21;TA
8290 LET S#(M)=STR# (VAL S#(M)-V
AL Y#)
8300 LET V#(M)=STR# (VAL V#(M)+V
AL Y#)
8310 GOTO 8130
8400 LPRINT
8410 LPRINT TAB 11;"TOTAL H.T.:"
TH
8420 LPRINT TAB 11;"    T.V.A.:"
TH*0.186
8430 LPRINT TAB 11;"TOTAL TTC.:"
TH*1.186
8440 LPRINT
8450 GOTO 8000
8600 PRINT AT 21,6;Y#;"  INEXIS
TANT"
8610 IF INKEY#="" THEN GOTO 8610
8620 GOTO 8130
8700 PRINT AT 18,6;"STOCK INSUFF
ISANT"
8710 IF INKEY#="" THEN GOTO 8710
8720 GOTO 8130

```

```

-----
8980 REM -----
8990 REM ENTREES EN STOCK
9000 CLS
9010 PRINT TAB 8;"entrees en stock"
9020 PRINT AT 6,1;"code: ";
9030 LET Y=6
9040 GOSUB 1200
9050 IF CODE Y#=0 THEN GOTO 500
9060 LET R#=Y#
9070 GOSUB 1000
9080 IF Q=2 THEN GOTO 9200
9090 PRINT AT 9,1;"quantite: ";
9100 LET Y=9
9110 GOSUB 1200
9120 LET S#(M)=STR# (VAL S#(M)+V
AL Y#)
9140 GOTO 9000
9200 PRINT AT 21,6;Y#;" INEXIST
ANT"
9210 IF INKEY#="" THEN GOTO 9210
9220 GOTO 9000
9400 REM -----
9490 REM STAT. DES VENTES
9500 CLS
9510 LPRINT TAB 4;"*-= STAT. DES
VENTES =-*"
9520 LPRINT
9530 LET CA=0
9540 LET BE=0
9550 LPRINT "CODE /VENDU/C.AFFAI
RES/BENEFICES"
9560 FOR I=1 TO NBA
9570 LPRINT N#(I);" ";V#(I);TAB
12;VAL P#(I)*VAL V#(I);TAB 23;(V
AL P#(I)*VAL V#(I))-(VAL A#(I)*V
AL V#(I))
9580 LET CA=CA+(VAL P#(I)*VAL V#
(I))
9590 LET BE=BE+(VAL P#(I)*VAL V#
(I))-(VAL A#(I)*VAL V#(I))
9600 NEXT I
9610 LPRINT
9620 LPRINT TAB 11;"C.AFFAIRES:"
;CA
9630 LPRINT TAB 11;"BENEFICES:"
;BE
9640 GOTO 500

```

IDENTIFICATION des VARIABLES

NOM SYMBOLIQUE	TYPE (*)	ORIGINE				LONG.	OBSERVATIONS
		Clavier	Programme	Cassette	Disquette		
\$ ()	TABLEAU	X					Tableau des prix d'achats.
E	VARIABLE	X					Total des bénéfices.
	VARIABLE	X					Choix de l'option (de 1 à 9).
A	VARIABLE	X					Chiffre d'affaires.
\$	VARIABLE	X					Nom du client.
\$ ()	TABLEAU	X					Tableau des désignations.
	VARIABLE	X					Pointeur de la partie inférieure
							du tableau des noms dans la
							recherche dichotomique.
	INDICE	X					Boucle FOR/NEXT.
	COMPTEUR	X					Nombre de lignes affichées sur
							l'écran dans l'option listing.
	VARIABLE	X					Milieu du tableau des noms dans la
							recherche dichotomique.
3 A	COMPTEUR	X					Nombre d'articles dans le fichier
\$ ()	TABLEAU	X					Tableau des codes articles.
\$ ()	TABLEAU	X					Tableau des prix de ventes.
	AIGUILLAGE	X					Si Q = 1 le nom cherché existe
							Si Q = 2 le nom cherché n'existe pas
\$	VARIABLE	X					Nom à trouver dans le tableau dans
							la recherche dichotomique.
							.../...

légende : aiguillage, compteur, constante, index, indice, tableau, variable. Reproduction interdite (c) Editions du P.S.I.

IDENTIFICATION des VARIABLES

NOM SYMBOLIQUE	TYPE (*)	ORIGINE				LONG.	OBSERVATIONS
		Clavier	Programme	Cassette	Disquette		
S ₁	VARIABLE	X					Pointeur de la partie supérieure du tableau des noms dans la recherche dichotomique.
S ₁ \$ ()	TABEAU	X					Tableau des quantités en stock.
T H	VARIABLE	X					Total du hors-tax.
T A	VARIABLE	X					Total du prix du stock pour un article.
V ₁ \$ ()	TABEAU	X					Tableau des quantités vendues.
X	VARIABLE	X					Position du prochain caractère à afficher dans la saisie caractère par caractère.
X \$	VARIABLE	X					Caractère saisi au clavier.
Y	VARIABLE	X					Numéro de la ligne en cours dans la saisie caractère par caractère
Y \$	VARIABLE	X					Variable tampon de la saisie caractère par caractère.

6/1980

(*) Légende : aiguillage, compteur, constante, index, indice, tableau, variable. Reproduction interdite (c) Editions du F

Tris...sweet tris

Le tri est en informatique l'une des opérations les plus utilisées.

Tout programme de gestion nécessite des classements de tableaux de salariés, de produits, de clients ...

Pourquoi présenter ici trois tris différents ? Parce qu'ils pourront être choisis dans des cas précis pour leurs caractéristiques propres.

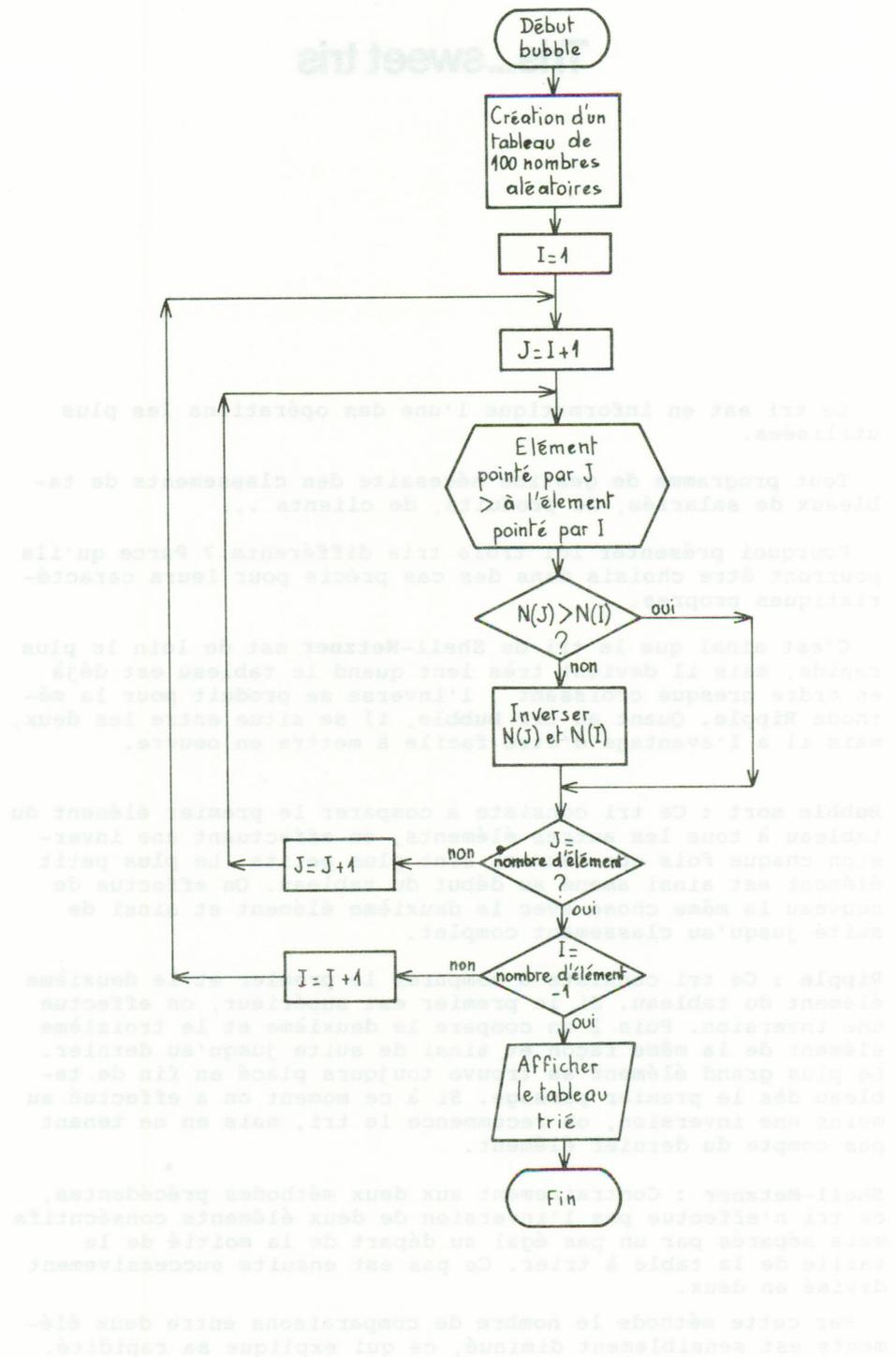
C'est ainsi que le tri de **Shell-Metzner** est de loin le plus rapide, mais il devient très lent quand le tableau est déjà en ordre presque croissant ; l'inverse se produit pour la méthode **Ripple**. Quant au tri **Bubble**, il se situe entre les deux, mais il a l'avantage d'être facile à mettre en oeuvre.

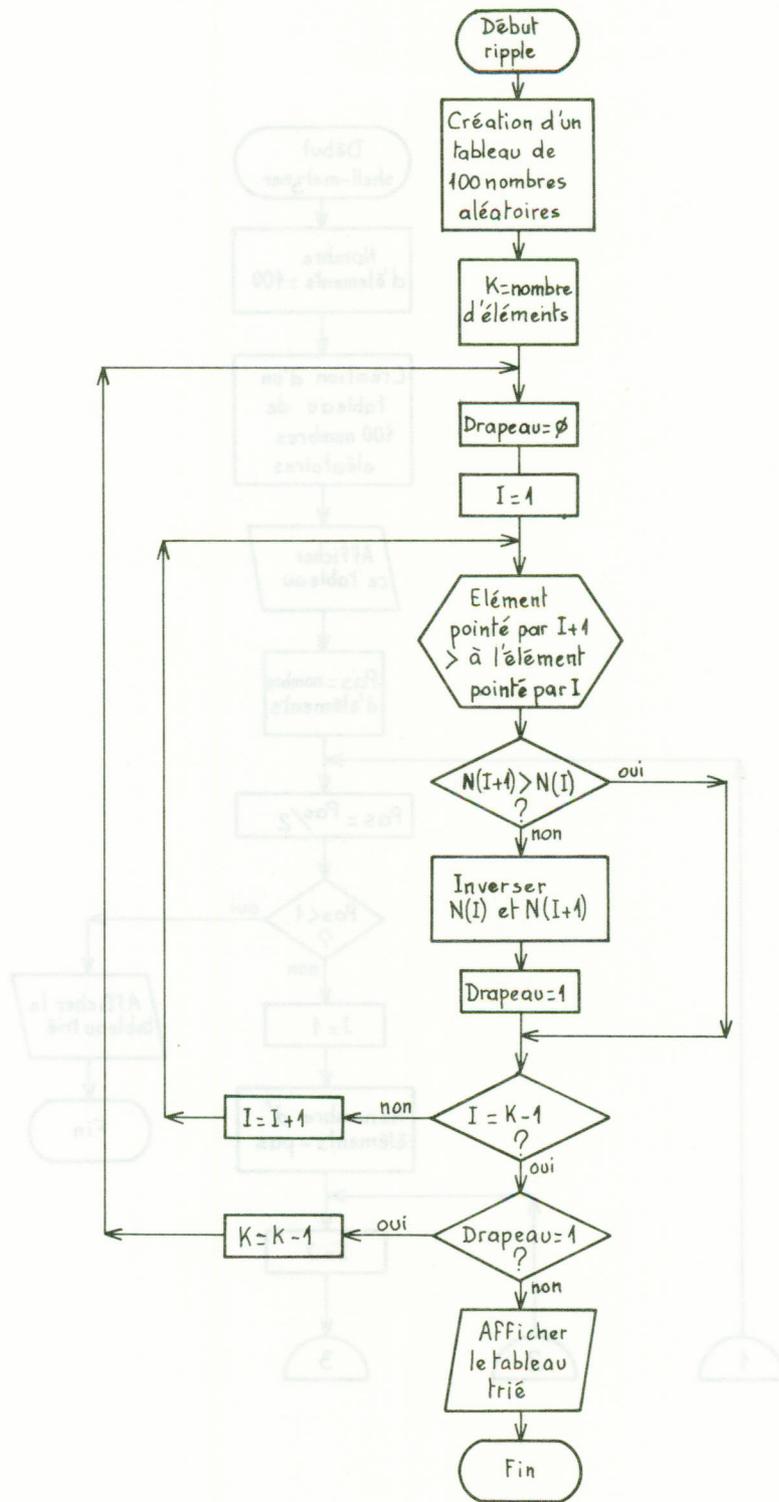
Bubble sort : Ce tri consiste à comparer le premier élément du tableau à tous les autres éléments, en effectuant une inversion chaque fois que ceux-ci sont plus petits. Le plus petit élément est ainsi amené au début du tableau. On effectue de nouveau la même chose avec le deuxième élément et ainsi de suite jusqu'au classement complet.

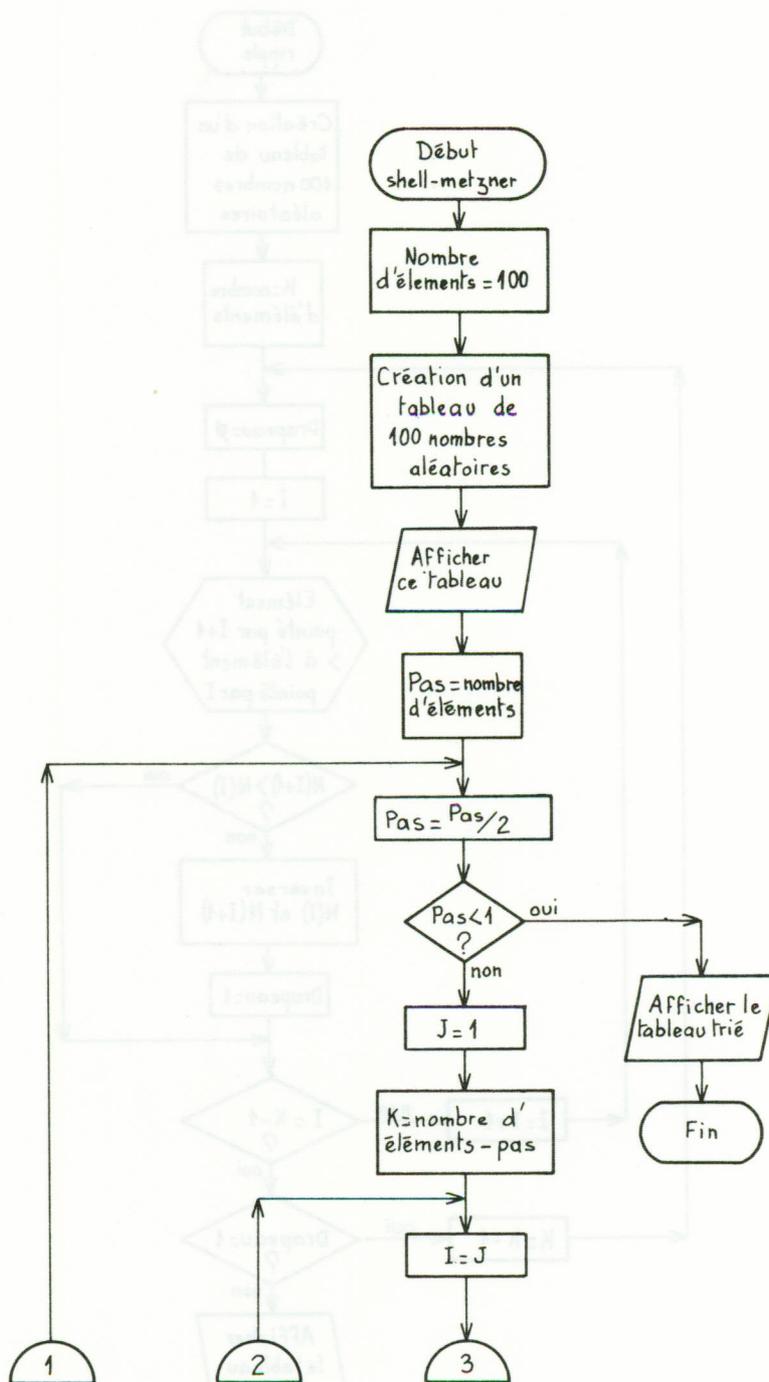
Ripple : Ce tri consiste à comparer le premier et le deuxième élément du tableau. Si le premier est supérieur, on effectue une inversion. Puis l'on compare le deuxième et le troisième élément de la même façon et ainsi de suite jusqu'au dernier. Le plus grand élément se trouve toujours placé en fin de tableau dès le premier passage. Si à ce moment on a effectué au moins une inversion, on recommence le tri, mais en ne tenant pas compte du dernier élément.

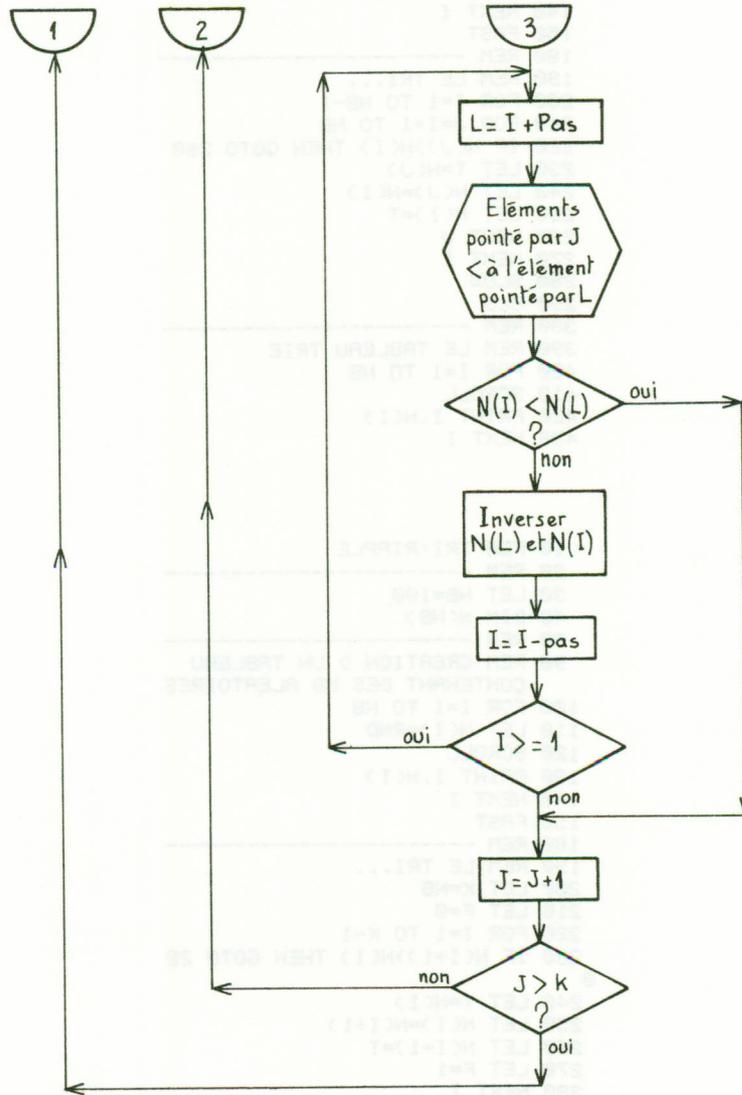
Shell-Metzner : Contrairement aux deux méthodes précédentes, ce tri n'effectue pas l'inversion de deux éléments consécutifs mais séparés par un pas égal au départ de la moitié de la taille de la table à trier. Ce pas est ensuite successivement divisé en deux.

Par cette méthode le nombre de comparaisons entre deux éléments est sensiblement diminué, ce qui explique sa rapidité.









```

10 REM TRI: BUBBLE SORT
20 REM -----
30 LET NB=100
40 DIM N(NB)
80 REM -----
90 REM CREATION D UN TABLEAU
  CONTENANT DES NB ALEATOIRES
100 FOR I=1 TO NB
110 LET N(I)=RND
120 SCROLL
130 PRINT I,N(I)
140 NEXT I
150 FAST
180 REM -----
190 REM LE TRI...
200 FOR I=1 TO NB-1
210 FOR J=I+1 TO NB
220 IF N(J)>N(I) THEN GOTO 260
230 LET T=N(J)
240 LET N(J)=N(I)
250 LET N(I)=T
260 NEXT J
270 NEXT I
280 SLOW
290 CLS
380 REM -----
390 REM LE TABLEU TRIE
400 FOR I=1 TO NB
410 SCROLL
420 PRINT I,N(I)
430 NEXT I

```

```

10 REM TRI: RIPPLE
20 REM -----
30 LET NB=100
40 DIM N(NB)
80 REM -----
90 REM CREATION D UN TABLEAU
  CONTENANT DES NB ALEATOIRES
100 FOR I=1 TO NB
110 LET N(I)=RND
120 SCROLL
130 PRINT I,N(I)
140 NEXT I
150 FAST
180 REM -----
190 REM LE TRI...
200 LET K=NB
210 LET F=0
220 FOR I=1 TO K-1
230 IF N(I+1)>N(I) THEN GOTO 28
240 LET T=N(I)
250 LET N(I)=N(I+1)
260 LET N(I+1)=T
270 LET F=1
280 NEXT I
290 IF F=0 THEN GOTO 320
300 LET K=K-1
310 GOTO 210
320 SLOW
330 CLS
380 REM -----
390 REM LE TABLEU TRIE
400 FOR I=1 TO NB
410 SCROLL
420 PRINT I,N(I)
430 NEXT I

```

1	0.13908386
2	0.43308624
3	0.40066524
4	0.50409517
5	0.07000000
6	0.40000000
7	0.00000000
8	0.10374085
9	0.03147888
10	0.00145002
11	0.00145002
12	0.00917224
1	0.13908386
2	0.43308624
3	0.40066524
4	0.50409517
5	0.07000000
6	0.40000000
7	0.00000000
8	0.10374085
9	0.03145002
10	0.00145002
11	0.00145002
12	0.00917224

```

10 REM TRI: SHELL-METZNER
20 REM -----
30 LET NB=100
40 DIM N(NB)
80 REM -----
90 REM CREATION D UN TABLEAU
  CONTENANT DES NB ALEATOIRES
100 FOR I=1 TO NB
110 LET N(I)=RND
120 SCROLL
130 PRINT I,N(I)
140 NEXT I
150 FAST
180 REM -----
190 REM LE TRI...
200 LET P=NB
210 LET P=INT (P/2)
220 IF P<1 THEN GOTO 360
230 LET J=1
240 LET K=NB-P
250 LET I=J
260 LET L=I+P
270 IF N(I)<N(L) THEN GOTO 330
280 LET T=N(L)
290 LET N(L)=N(I)
300 LET N(I)=T
310 LET I=I-P
320 IF I>=1 THEN GOTO 260
330 LET J=J+1
340 IF J>K THEN GOTO 210
350 GOTO 250
360 SLOW
370 CLS
380 REM -----
390 REM LE TABLEAU TRIE
400 FOR I=1 TO NB
410 SCROLL
420 PRINT I,N(I)
430 NEXT I
  
```


Test mémoire

Ce programme en assembleur va tester pour vous votre extension 16 K.

Mais pour rentrer ce programme quelques manipulations sont nécessaires. Il faut d'abord taper au clavier en mode direct :

- POKE 16 388,0
- POKE 16 389,68
- NEW.

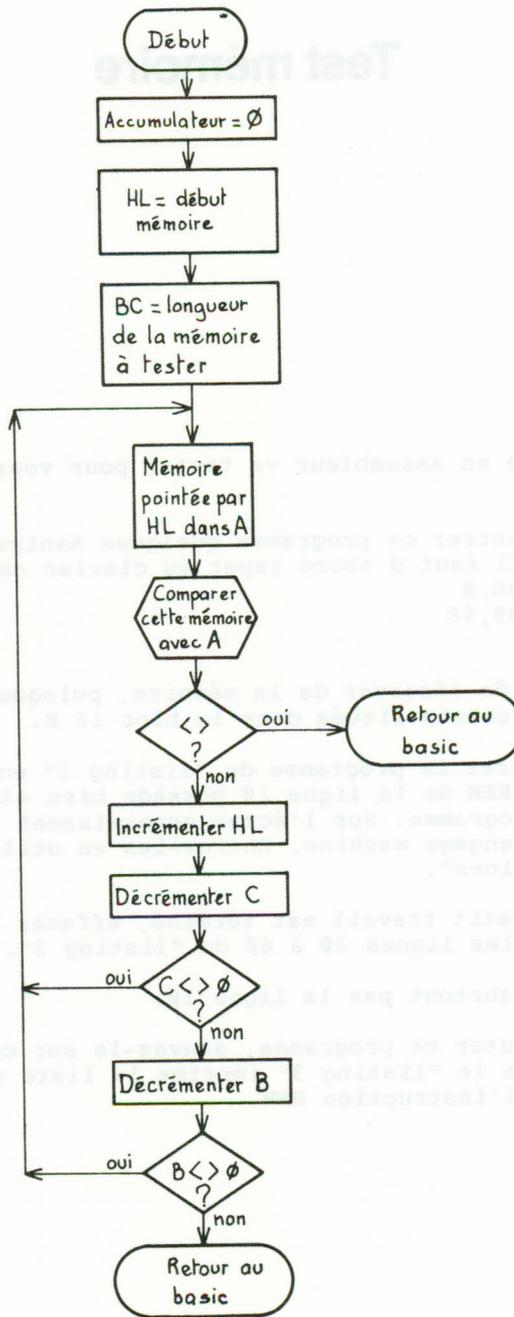
Ceci permet de réserver de la mémoire, puisque l'on va modifier tous les octets situés dans le bloc 16 K.

Ensuite, entrez le programme du "listing 1" en vérifiant que l'instruction **REM** de la ligne 10 possède bien 44 caractères. Exécutez ce programme. Sur l'écran apparaissent les adresses des codes en langage machine. Entrez-les en utilisant "la liste des codes machines".

Dès que ce petit travail est terminé, effacez les lignes 20 à 70 et tapez les lignes 20 à 60 du "listing 2".

Ne modifiez surtout pas la ligne 10.

Avant d'exécuter ce programme, sauvez-le sur cassette. En cas de problème le "listing 3" imprime la liste des codes contenus dans l'instruction **REM**.



```

10 REM 12345678901234567890123
456789012345678901234
20 FOR I=0 TO 43
30 SCROLL
40 PRINT 16514+I
50 INPUT D
60 POKE 16514+I,D
70 NEXT I

```

```

10 REM Y LN #RNDY?LN #RNDYeLN
#RNDY COPY LN #RNDTAN 5 ?* W?yTA
B @RND7#TAB @RND#TAB @RNDTAN LPR
INT TAN
20 LET E=USR 16514
30 IF E=0 THEN GOTO 60
40 PRINT "ERREUR EN ";32767-E
50 STOP
60 PRINT "MEMOIRE O.K"

```

```

10 REM Y LN #RNDY?LN #RNDYeLN
#RNDY COPY LN #RNDTAN 5 ?* W?yTA
B @RND7#TAB @RND#TAB @RNDTAN LPR
INT TAN
20 FOR I=0 TO 43
30 LPRINT 16514+I,PEEK (16514+
I)
40 NEXT I

```

16514	62
16515	0
16516	205
16517	151
16518	64
16519	62
16520	85
16521	205
16522	151
16523	64
16524	62
16525	170
16526	205
16527	151
16528	64
16529	62
16530	255
16531	205
16532	151
16533	64
16534	201
16535	33
16536	0
16537	68
16538	1
16539	0
16540	60
16541	119
16542	190
16543	194
16544	172
16545	64
16546	35
16547	13
16548	194
16549	157
16550	64
16551	5
16552	194
16553	157
16554	64
16555	201
16556	225
16557	201

EMPLACEMENTS MEMOIRE

16514 à 16534

L'accumulateur est chargé successivement des valeurs 0,55, AA et FF avant l'appel du sous-programme de Test.

Ces valeurs ont la particularité d'être quatre combinaisons binaires intéressantes pour un test mémoire (voir listing assembleur). Le RETURN en 16534 étant le dernier de la pile, on revient au Basic pour vérifier la valeur contenue dans les registres B et C.

16535 à 16557

Les registres HL et BC contiennent respectivement l'adresse de début et le nombre d'octets à tester.

On place dans la mémoire pointée par HL la valeur contenue dans l'Accumulateur (adressage indirect registre). Puis l'on teste le contenu de cette même mémoire avec A. Si la mémoire est défectueuse, on obtient quelque chose de différent. Une telle erreur envoie à l'adresse 16556 (dans le cas contraire, on décrémente C et D).

Si on utilisait une instruction RET directement en cas d'erreur, on reviendrait au test pour une autre valeur de l'accumulateur.

L'astuce ici consiste à diminuer la pile de deux octets en les plaçant dans HL (instruction POP), de manière à retourner directement au Basic. On utilise HL, puisque les registres B et C sont utilisés par le Basic pour donner l'adresse de l'erreur. On aurait pu tout aussi bien décrémente deux fois le pointeur de pile (SP).

Espérance de vie

Vous ne le savez peut-être pas, mais les voyantes modernes ne vous répondent plus à la légère quand vous leur demandez combien de temps vous vivrez.

Elles calculent consciencieusement votre espérance de vie grâce à un ordinateur discrètement relié à leur boule de cristal.

Lequel ordinateur utilise le test très sérieux qui vous est proposé dans les pages suivantes. Il a été mis au point par un médecin anglais d'après les renseignements d'ordre privé demandé par les assureurs-vie à leurs clients.

Ce test est basé sur le principe d'une espérance de vie correspondant à chaque âge (71 ans si l'on a 30 ans, 73 ans si l'on a 50 ans ...). Mais des critères négatifs (abus du tabac, de l'alcool ...) ou positifs (bon sommeil, longévité des parents) peuvent rallonger ou raccourcir cette espérance de vie de plusieurs années. Bien sûr, cela reste du domaine des statistiques, chaque histoire humaine étant unique et imprévisible! Mais on peut méditer ...

ESPERANCE DE VIE

Esperance de vie

- 1-VOUS AVEZ UN PARENT DECEDE
- 0-UNE CRISE CARDIAQUE, EMBOLIE OU ARTERIOSCLEROSE AVANT 50 ANS
- 2-DECEDE ENTRE 50 ET 60 ANS
- 3-DECEDE DE DIABETE AVANT 60 ANS
- 4-DECEDE DE CANCER DE L'ETOMAC AVANT 60 ANS
- 5-AUCUN

REPONDEZ PAR UN CHIFFRE COMPRIS ENTRE 1 ET 5

ESPERANCE DE VIE

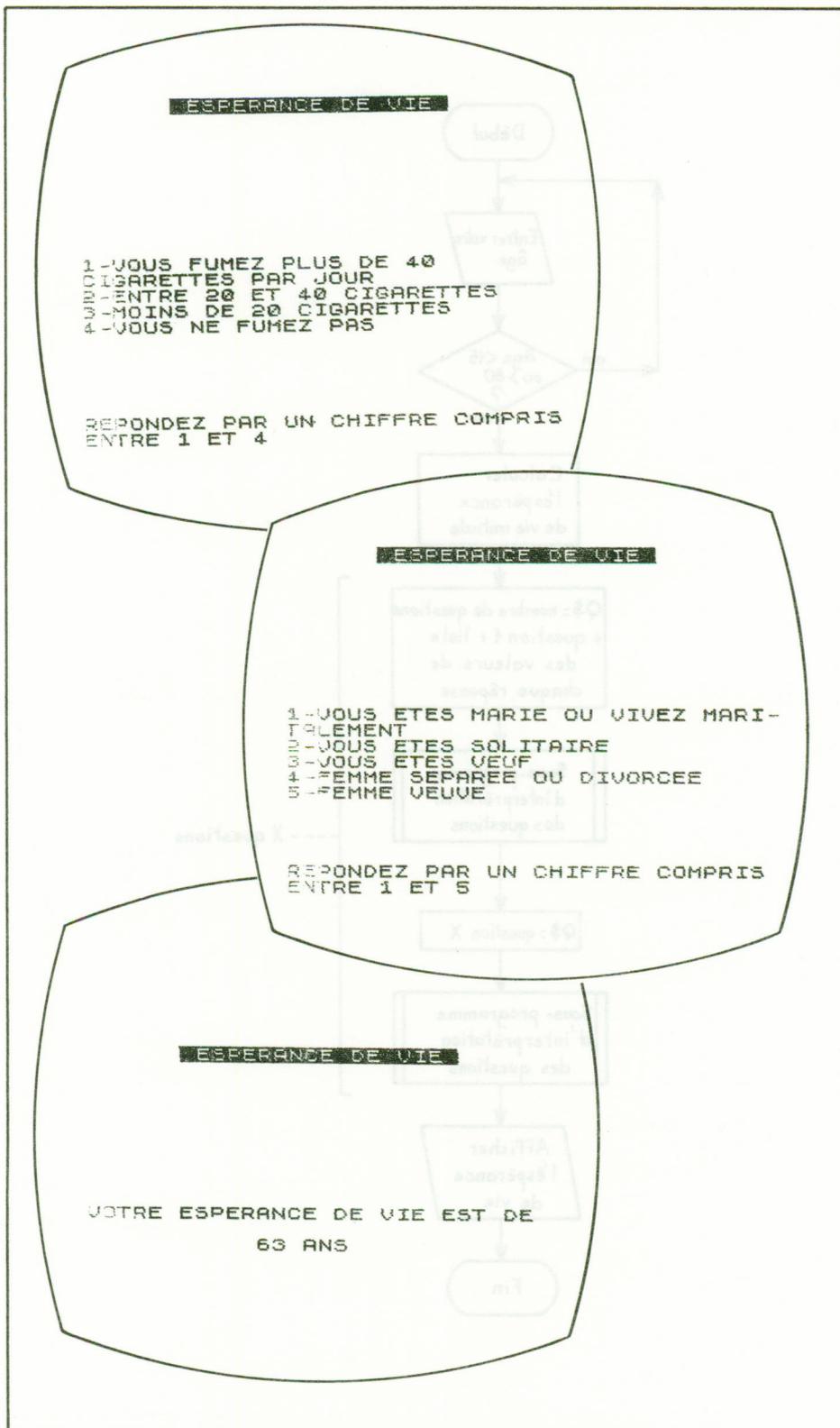
VOUS DORMEZ PLUS DE 10 HEURES PAR NUIT OU MOINS DE 5 HEURES

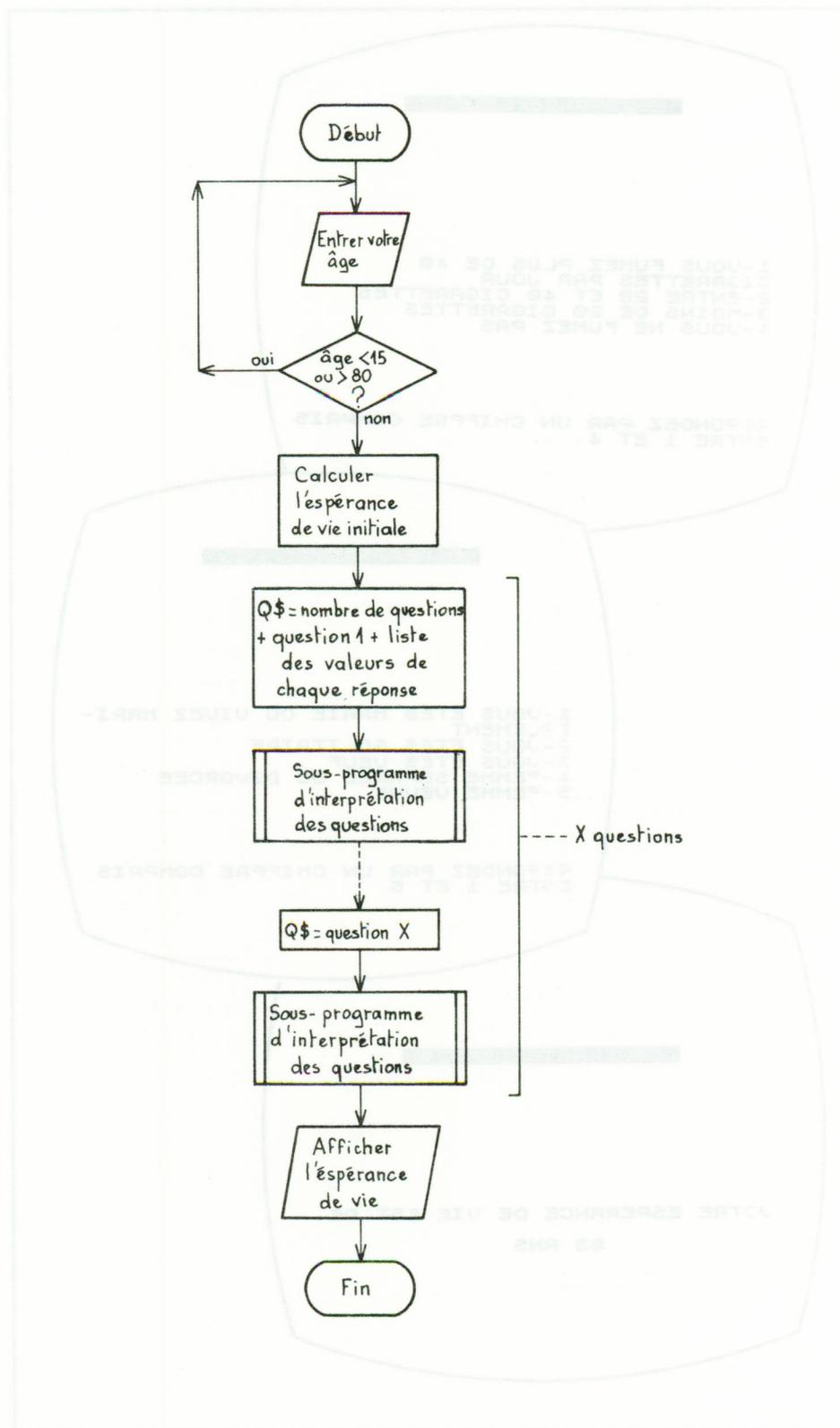
REPONDEZ PAR O OU N (OUI/NON) ?

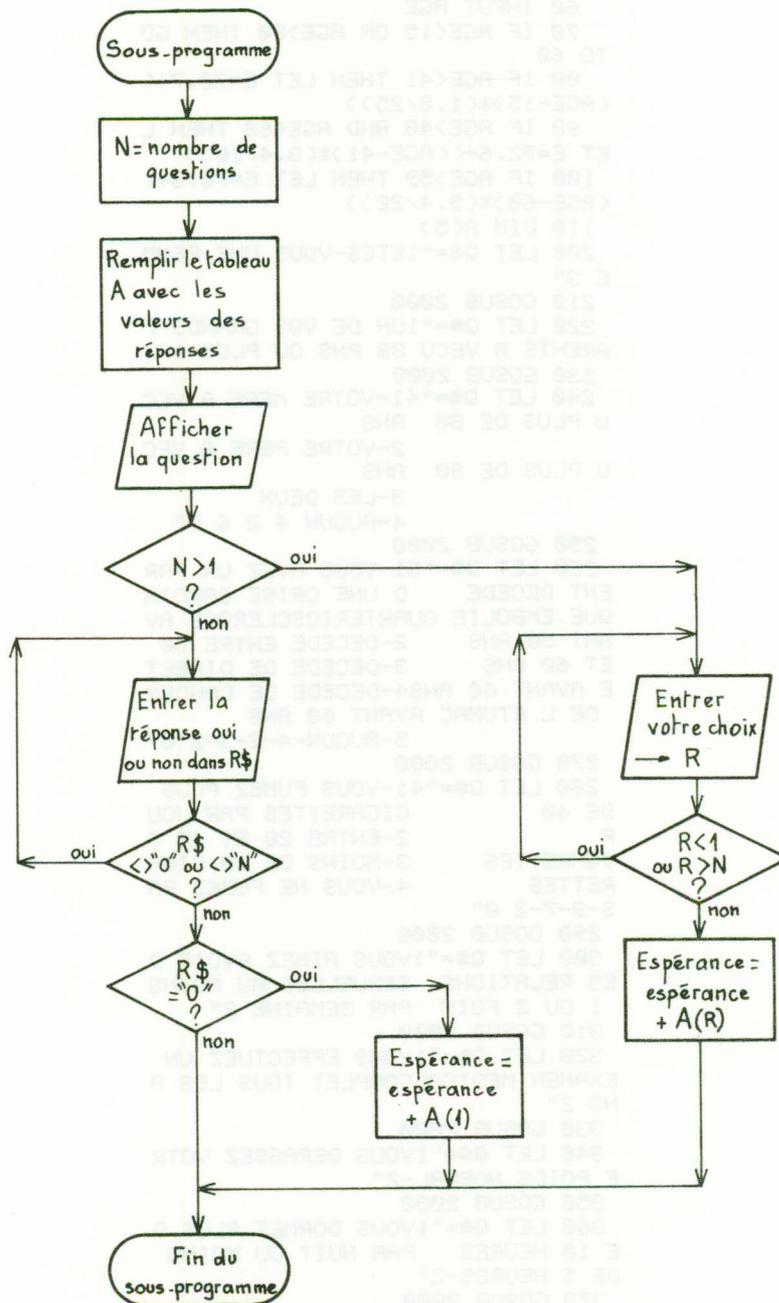
ESPERANCE DE VIE

- 1-VOUS BUVEZ MODEREMENT (1/2 LITRE DE VIN OU 4 VERRES DE BIERE)
- 2-VOUS NE BUVEZ PAS
- 3-VOUS BUVEZ BEAUCOUP

REPONDEZ PAR UN CHIFFRE COMPRIS ENTRE 1 ET 3







```

10 REM ESPERANCE DE VIE
20 REM -----
30 LET T#="Esperance de vie"
40 PRINT TAB 5;T#
50 PRINT AT 21,0;"VOTRE AGE ?"
60 INPUT AGE
70 IF AGE<15 OR AGE>80 THEN GO
TO 60
80 IF AGE<41 THEN LET E=70.7+(
<AGE-15>*(1.8/25))
90 IF AGE>40 AND AGE<60 THEN L
ET E=72.6+((AGE-41)*(3.4/18))
100 IF AGE>59 THEN LET E=76.3+(
<AGE-60>*(9.4/20))
110 DIM A(5)
200 LET Q#="1ETES-VOUS UNE FEMM
E 3"
210 GOSUB 2000
220 LET Q#="1UN DE VOS GRANDS-P
ARENTS A VECU 80 ANS OU PLUS 1"
230 GOSUB 2000
240 LET Q#="41-VOTRE MERE A VEC
U PLUS DE 80 ANS
2-VOTRE PERE A VEC
U PLUS DE 80 ANS
3-LES DEUX
4-AUCUN 4 2 6 0"
250 GOSUB 2000
260 LET Q#="51-VOUS AVEZ UN PAR
ENT DECEDE D UNE CRISE CARDIA
QUE,EMBOLIE QUARTERIOSCLEROSE AV
ANT 50 ANS 2-DECEDE ENTRE 50
ET 60 ANS 3-DECEDE DE DIABET
E AVANT.60 ANS4-DECEDE DE CANCER
DE L ETOMAC AVANT 60 ANS
5-AUCUN-4-2-3-2 0"
270 GOSUB 2000
280 LET Q#="41-VOUS FUMEZ PLUS
DE 40 CIGARETTES PAR JOU
R 2-ENTRE 20 ET 40 C
IGARETTES 3-MOINS DE 20 CIGA
RETTES 4-VOUS NE FUMEZ PA
S-9-7-2 0"
290 GOSUB 2000
300 LET Q#="1VOUS AIMEZ AVOIR D
ES RELATIONS SEXUELLES AU MOINS
1 OU 2 FOIS PAR SEMAINE 2"
310 GOSUB 2000
320 LET Q#="1VOUS EFFECTUEZ UN
EXAMEN MEDICALCOMPLET TOUS LES A
NS 2"
330 GOSUB 2000
340 LET Q#="1VOUS DEPASSEZ VOTR
E POIDS NORMAL-2"
350 GOSUB 2000
360 LET Q#="1VOUS DORMEZ PLUS D
E 10 HEURES PAR NUIT OU MOINS
DE 5 HEURES-2"
370 GOSUB 2000
380 LET Q#="31-VOUS BUVEZ MODER
EMENT (1/2 LI-TRE DE VIN OU 4 VE
RRRES DE BIERE)2-VOUS NE BUVEZ PA
S 3-VOUS BUVEZ BEAUC
OUP 3 0-8"
390 GOSUB 2000

```

```

400 LET Q#="1VOUS PRENEZ DE L E
XERCICE TROIS FOIS PAR SEMAINE: C
YCLISME, MARCHÉ RAPIDE, NAT
ATION, DANSE 3"
410 GOSUB 2000
420 LET Q#="31-VOUS ETES EX-UNI
VERSITAIRE 2-VOUS ETES BACHEL
IER 3-NI L UN NI L AUT
RE 3 2 0"
430 GOSUB 2000
440 LET Q#="21-VOUS ETES CITADI
N 2-VOUS ETES CAMPAG
NARD 1-1"
450 GOSUB 2000
460 LET Q#="51-VOUS ETES MARIE
OU VIVEZ MARI-TALEMENT
2-VOUS ETES SOLITA
IRE 3-VOUS ETES VEUF
4-FEMME SEPARÉE OU
DIVORCÉE 5-FEMME VEUVE 1-9-
7-4-3"
470 GOSUB 2000
480 REM -----
490 REM RESULTAT
500 CLS
510 PRINT TAB 6;T#
520 LET E=INT (E)
530 PRINT AT 10,0;"VOTRE ESPERA
NCE DE VIE EST DE"
540 PRINT
550 PRINT TAB 11;E;" ANS"
600 IF INKEY#="" THEN GOTO 600
610 CLS
620 RUN
1980 REM -----
1990 REM INTERPRETEUR DE
QUESTIONS

2000 CLS
2010 PRINT TAB 6;T#
2020 LET N=VAL Q#(1 TO 1)
2030 LET LQ=LEN (Q#)-1-(N*2)
2040 FOR I=1 TO N
2050 LET A(I)=VAL (Q#(LQ+2+(2*(I
-1)) TO LQ+3+(2*(I-1))))
2060 NEXT I
2070 PRINT AT 10,0;Q#(2 TO LQ+1)
2080 IF N>1 THEN GOTO 2200
2090 PRINT AT 21,0;"REPONDEZ PAR
O OU N (OUI/NON) ?"
2100 INPUT R#
2110 IF R#<>"O" AND R#<>"N" THEN
GOTO 2100
2120 IF R#="O" THEN LET E=E+A(1)
2130 RETURN
2200 PRINT AT 20,0;"REPONDEZ PAR
UN CHIFFRE COMPRIS"
2210 PRINT "ENTRE 1 ET ";N
2220 INPUT R
2230 IF R<1 OR R>N THEN GOTO 222
0
2240 LET E=E+A(R)
2250 RETURN

```

LE PROGRAMME**Lignes 40 à 110**

L'espérance de vie initiale (variable E) est calculée suivant une courbe statistique pour des âges allant de 15 à 80 ans. Pour suivre le plus possible cette courbe, on effectue un découpage en trois tranches : de 15 à 40 ans, de 41 à 59 ans et de 60 à 80 ans.

Lignes 220 à 470

On place dans la variable Q\$ la question ainsi que tous les éléments nécessaires au calcul de l'espérance de vie. Le premier caractère correspond au nombre de réponses possibles, les derniers aux valeurs à ajouter ou à retrancher à E (suivant la réponse de l'utilisateur). Puis l'on va au sous-programme de la ligne 2000 pour interpréter cette chaîne de caractères.

Lignes 500 à 620

On affiche le résultat contenu dans la variable E avant de passer à un autre cas.

Lignes 2000 à 2250 sous-programme

La variable Q\$ est interprétée suivant le découpage expliqué plus haut. Le nombre de questions est placé dans N, les valeurs étant placées dans le tableau A.

On va à la ligne 2200, si une seule question est posée, car dans ce cas il faut répondre par oui ou non et non par le numéro de la question.

On ajoute à l'espérance de vie (variable E) une des valeurs du tableau A suivant la réponse (toujours A(1) s'il n'y a qu'une seule question).

82,00FF **

PROGRAMMES

**JEAN-FRANÇOIS
SEHAN**

ETUDES TOME 2 POUR ZX81

Ce second tome "d'Études pour ZX81" contient des programmes en langage Assembleur appliqués aux modules d'extension comme l'imprimante ou la carte génératrice de caractères. Les 20 programmes proposés vous permettront d'exercer vos talents de compositeurs grâce à la carte sonore et au générateur de caractères, de dessiner des histogrammes grâce à la carte haute résolution, ou tout simplement de jouer au baccara, aux piranhas et au taquin.

Editions du P.S.I.

Boîte Postale 86

F-77400 Lagny/Marne

ISBN 2.86595.061.1 — Imprimé en France