

BOITE



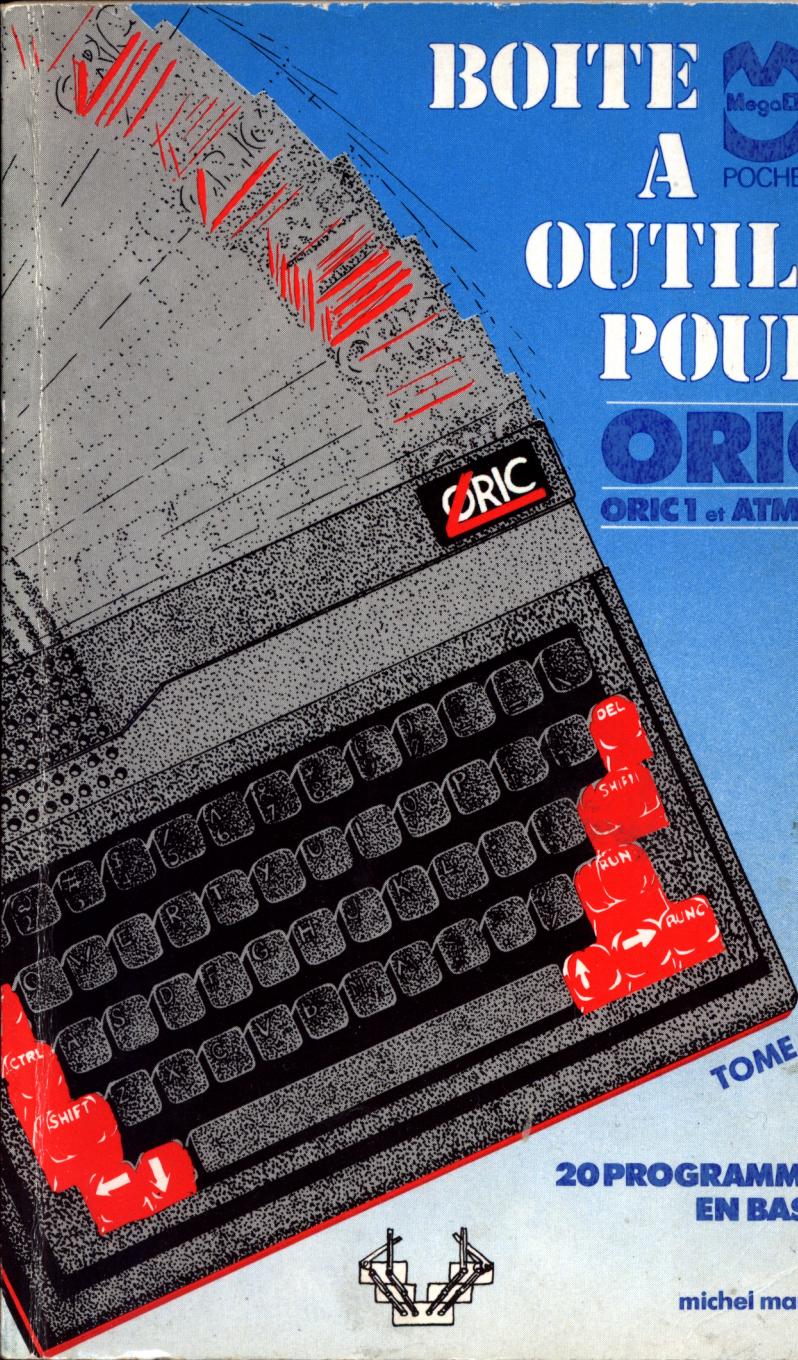
POCHE

A

OUTILS POUR

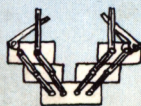
ORIC

ORIC1 et ATMOS



TOME 1

**20 PROGRAMMES
EN BASIC**



michel martin

Boîte à outils
pour
ORIC
(ORIC 1 et ATMOS)

Boîte à outils pour ORIC (ORIC 1 et ATMOS) - ISBN 2.87109.008.4

© s.a. SAPECA, Uccle (Belgique) 1984

Toute reproduction, non strictement réservée à l'usage du copiste, d'un extrait quelconque de ce livre par quelque procédé que ce soit, est interdite sans autorisation écrite de l'éditeur.

Les collections **MégaO** (Poche) sont éditées par SAPECA, 5 avenue de la Ferme Rose, 1180 Bruxelles (Roze Hoevelaan - 1180 Brussel). Le label **MégaO**, les titres des collections et la présentation des volumes sont déposés conformément à la loi. Distributeurs en **France** : PSI DIFFUSION, BP 86, 77402 Lagny-sur-Marne Cedex. Pour le **Canada** et les **USA** : S.C.E. Inc., 65 Avenue Hillside, Westmount - Québec H3 Z1 W1. Pour le **Benelux** : PSI Benelux, 5 Avenue de la Ferme Rose, 1180 Bruxelles.

Boîte à outils **pour** **ORIC** **(ORIC 1 et ATMOS)**

20 programmes

par
Michel Martin



POCHE

1984

Sommaire

Avertissement au lecteur	7
JEU - D'adresse ou de rapidité	
- Attaque aérienne	10
- Casse-pipes	17
- Le piéton courageux	23
- Réflexe	29
JEU - De hasard	
- Bataille navale	34
- Jack pot	40
- Jeu du 21	48
JEU - De réflexion	
- Jeu des allumettes	54
- Les piranhas attaquent	59
- Mémoire à court terme	65
- Mastermind	70
GRAPHIQUES ET MATHÉMATIQUES	
- Figures de Moivre	76
- Biorythmes	81
- Jeu de la vie	87
- Les unités anglo-saxonnes : GB	F 93
- Les unités anglo-saxonnes : F	GB 100
L'ORIC JOUE ET DESSINE	
- ORIC piano	106
- ORIC joue : Pop Corn, Charlot, Classique	112
- Le chien	117
UTILITAIRE	
- ORIC et l'alphabet	120
- Comptabilité ORIC 1 et ATMOS	127

Avertissement

La plupart des programmes peuvent être mis sous une forme structurée et hiérarchisée.

Mais, qu'est-ce qu'un programme structuré ?

C'est un programme présentant une base (ou structure) bien définie et logique.

Pourquoi structurer un programme ?

Simplement pour en faciliter la compréhension et la maintenance.

Nous voyons donc que, dans la mesure du possible, un programme aura intérêt à être structuré.

Voyons maintenant ce qu'est un programme hiérarchisé :

C'est un programme possédant des niveaux de complexité (ou hiérarchie) différents.

Pourquoi hiérarchiser un programme ?

Cela le rend plus compréhensible et facilite la maintenance.

Les objectifs de la hiérarchisation et de la structuration se recoupent. Tous les programmes présentés ici sont, dans la mesure du possible, structurés et hiérarchisés.

Ils possèdent un module de niveau hiérarchiquement élevé :

• C'est le **MODULE DE COMMANDE**

Il se charge d'activer les modules principaux du

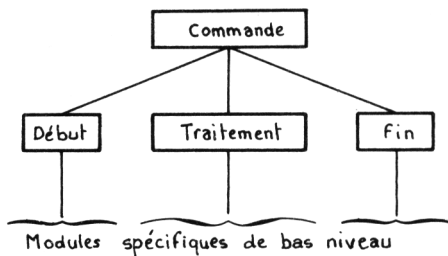
programme.

Au niveau immédiatement inférieur se trouvent :

- Les **MODULES PRINCIPAUX**
Généralement au nombre de trois, ils réalisent :
Le **DEBUT**, le **TRAITEMENT** et la **FIN** du programme.

Les niveaux inférieurs comprennent des **PROCEDURES SPECIFIQUES** ou des **ACTIONS ELEMENTAIRES**, souvent sollicitées.

En conclusion, les programmes qui suivent possèdent la structure suivante :



(structure A)

La logique concernant les modules **DEBUT** et **FIN** sera développée dans le premier programme. Elle ne sera pas reprise par la suite car elle varie très peu. Alors, seule la logique concernant le traitement sera développée.

Remarque : Dans la suite, quelques programmes, parmi les plus simples, possèdent une structure entièrement linéaire référencée " structure B ".

**JEUX D'ADRESSE
OU DE RAPIDITÉ**

Attaque aérienne

PRESENTATION

Un tank disposé au sol doit détruire des missiles ennemis en vol. Le tank peut se déplacer et tirer autant de roquettes anti-missiles qu'il le désire. Le jeu s'arrête quand le tank est détruit.

Ce jeu demande une rapidité certaine et ... une certaine maîtrise de soi.

Combien de temps tiendrez-vous face à l'attaque aérienne ?

ORGANIGRAMME DU JEU :

Le programme a la structure A.

Le module "DEBUT" réalise :

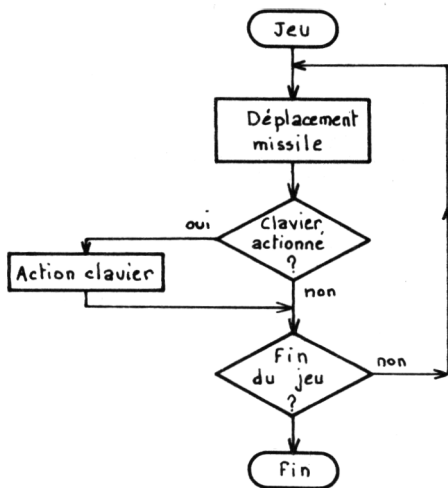
- . L'affichage de la règle du jeu,
- . L'initialisation des variables du jeu.

Le jeu commence dès qu'une touche du clavier est actionnée.

Le module "FIN" commente la partie.

Dans la suite, tous les programmes de structure A possèdent des modules "DEBUT" et "FIN" identiques.

Le jeu à la structure suivante :



Déplacement missile :

Cette procédure comprend les fonctions :

- . Tirage aléatoire de la position de départ du missile,
- . Tirage aléatoire de la direction du missile,
- . Déplacement du missile sur l'écran.

Action clavier :

Tout missile se trouvant sur une ligne verticale issue de la position du tank est détruit et le score augmente de 1.

LE PROGRAMME

Il se compose en trois pavés :

Initialisation	Présentation du jeu	Lignes 1100 à 1540
	Initialisation du jeu	Lignes 1700 à 1920
Jeu	Lignes 2000 à 2950	
	Le jeu se décompose en :	
	- Déplacement missile	Lignes 2100 à 2240
	- Action joueur	Lignes 2600 à 2650

- Diverses procédures
de bas niveau :

● Joueur touché	Lignes 2260 à 2300
● Abri touché	Lignes 2320 à 2350
● Tir joueur	Lignes 2700 à 2770
● Joueur à droite	Lignes 2800 à 2850
● Joueur à gauche	Lignes 2900 à 2950

Fin de jeu

Lignes 3000 à 3060

Restitution du curseur et commentaires.

LES VARIABLES UTILISEES

A	Position du missile au départ
A\$	Code couleur de la ligne
B	Déplacement horizontal du missile
BIS	Indicateur du nouveau missile
B\$	Caractère déplacement vers le haut
C	Position écran
CC	Position du tank
CO	Position colonne missile
FIN	Indicateur de fin de partie
LI	Position ligne missile
NB	Nombre de missiles touchés
TC	Indicateur de missile touché

```

10 : REM Attaque aerienne
20 :
30 : GOSUB 1000 'Initialisation
40 : GOSUB 2000 'Deroulement du jeu
50 : IF RE=1 THEN GOTO 10
55 : GOSUB 3000 'Fin du jeu
56 : END
60 :
70 : REM #####
1000 : REM Initialisation du jeu
1010 :
1020 : GOSUB 1100 'Affichage du menu
1030 : GOSUB 1700 'Initialisation des variables et grille jeu
1035 :
1040 : RETURN
1050 :
1060 : REM #####
1100 : REM Affichage du menu
1110 :
1120 : PAPER 0 : INK 5 : PRINT CHR$(17)
1130 : REM Titre
1140 :
1150 : PRINT CHR$(12)
1160 : PRINT CHR$(4);CHR$(27);"N";
1170 : PRINT CHR$(27);"A" ATTACHE AERIENNE";CHR$(4)
1180 :
1190 : REM Objet
1200 :
1210 : PRINT:PRINT:PRINT:PRINT
1220 : A$="D" 'Code couleur = Bleu
1230 : PRINT"OBJET : ";
1240 : GOSUB 1600:PRINT"Jeu d'adresse et de rapidite";
1250 :
1260 : REM Job
1270 :
1280 : PRINT:PRINT:PRINT
1290 : A$="F" 'Code couleur = Cyan
1300 : PRINT"JOB : ";
1310 : GOSUB 1600:PRINT"Vous etes attaque par des "
1320 : GOSUB 1600:PRINT" missiles aeriens reperes par !"
1330 : GOSUB 1600:PRINT" Le but du jeu est de les"
1340 : GOSUB 1600:PRINT" detruire . Vous disposez de"
1350 : GOSUB 1600:PRINT" roquettes envoyees par la"
1360 : GOSUB 1600:PRINT" touche SPACE . Les touches"
1370 : GOSUB 1600:PRINT" deplacement vers la droite et"
1380 : GOSUB 1600:PRINT" vers la gauche gerent votre"
1390 : GOSUB 1600:PRINT" position au sol . Vous disposez"
;
1400 : GOSUB 1600:PRINT" egalement de 2 abris au sol ."

```

```

1410 :   GOSUB 1600:PRINT"      Combien d'ennemis tuerez-vous ?"
;
1420 :   REM Debut
1430 :
1440 :   PRINT:PRINT:PRINT
1450 :   A$="C" 'Code couleur = Jaune
1460 :   PRINT"DEBUT : ";
1470 :   GOSUB 1600:PRINT"Pour commencer , appuyez sur";
1480 :   GOSUB 1600:PRINT"      une touche quelconque ."
1490 :   GET A$ 'Attente d'une action clavier
1500 :
1510 : RETURN
1520 :
1530 : REM *****
1540 : RETURN
1600 : REM Couleur d'une ligne
1610 :
1620 :   PRINT CHR$(27);"H";CHR$(27);A$;
1630 :
1640 : RETURN
1700 : REM Initialisation des variables
1710 :
1720 :   BIS=1 'Missile a l'origine
1730 :   CC=11 'Joueur a gauche de l'ecran
1735 :   B$=CHR$(11) 'Remise a hauteur de l'affichage
1740 :   CLS 'Effacement de l'ecran
1750 :   FOR I=1 TO 4:PLOT 12+I,18,"=":PLOT 21+I,18,"=":NEXT I
1760 :   FOR I=1 TO 21
1770 :     PLOT 8+I,7,"-":PLOT 8+I,21,"-"
1780 :   NEXT I
1790 :   FOR I=1 TO 13
1800 :     PLOT 9,7+I,"I":PLOT 29,7+I,"I" *
1810 :   NEXT I
1820 :   PLOT 11,20,"2" 'Position joueur
1830 :
1840 :   FOR I=0 TO 7
1850 :     READ DT:POKE46600+25*I,DT
1860 :   NEXT I
1870 DATA 12,12,12,12,63,63,63,0
1880 :
1890 : PLOT1,2,"Score":PRINT:PRINT:PRINT
1900 :
1910 : RETURN
1920 : REM *****
2000 : REM Jeu
2010 :
2020 :   GOSUB 2100 'Deplacement missile
2030 :   A$=KEY$:IF A$("<)" THEN GOSUB 2600
2040 :   IF FIN=0 THEN GOTO 2020 ELSE RETURN

```



```

2050 :
2060 : REM *****
2100 : REM Deplacement missile
2110 :
2120 : IF BIS<>1 THEN GOTO 2160
2130 : A=INT(RND(1)*3):A=A*2+17 'Missile au depart
2140 : B=INT(RND(1)*3)-1 'Deplacement horizontal
2150 : PLOT A,B,"*":LI=10:CO=A+B:BIS=0:RETURN
2160 : C=SCRN(CO,LI) 'Nouvelle position missile
2170 : PLOT CO-B,LI-2," "
2180 : IF LI=22 THEN BIS=1:RETURN
2190 : IF C= 90 THEN GOTO 2260 'Joueur touche
2200 : IF C=61 THEN GOTO 2320 'Abri touche
2210 : PLOT CO,LI,"*" 'Avance missile
2220 : LI=LI+2:CO=CO+B:RETURN 'Prochaine position missile
2230 :
2240 : RETURN
2250 : REM *****
2260 : REM Joueur touche
2270 :
2280 : SHOOT:FIN=1 'Fin de la balle
2290 :
2300 : RETURN
2310 : REM *****
2320 : REM Abri touche
2330 :
2340 : ZAP:PLOT CO-1,LI," ":PLOT CO,LI," ":BIS=1
2350 : RETURN
2360 : REM *****
2600 : REM Action joueur
2610 :
2620 : IF A$=" " THEN GOTO 2700 'Tir joueur
2630 : IF ASC(A$)=9 THEN GOTO 2800 'Deplacement vers la droite
2640 : IF ASC(A$)=8 THEN GOTO 2900 'Deplacement vers la gauche
2650 : RETURN
2660 : REM *****
2700 : REM Tir joueur
2710 :
2720 : FOR I=8 TO 18 STEP 2
2730 : IF SCRN(CC,I)=42 THEN TC=1:J=I:I=18:PLOTCC,J,"0"
2740 : NEXT I
2750 : IF TC=1 THEN EXPLODE:TC=0:NB=NB+1:BIS=1:PLOTCC,J," ":PRINT
" NB;B$
2760 : IF NB=0 THEN FIN=0 'Jeu gagne
2770 : RETURN

```

```
2780 : REM #####
2800 : REM Deplacement vers la drbite
2810 :
2820 : IF CC=27 THEN RETURN
2830 : PLOT CC,20," ":CC=CC+2:PLOTCC,20,"Z"
2840 : RETURN
2850 : REM #####
2900 : REM Deplacement vers la gauche
2910 :
2920 : IF CC=11 THEN RETURN
2930 : PLOT CC,20," ":CC=CC-2:PLOT CC,20,"Z"
2940 : RETURN
2950 : REM #####
3000 : REM Fin de la partie
3010 :
3020 : PRINT CHR$(17) 'Restitution du curseur
3030 : PLOT 0,23," Tu n'es plus de ce monde !!"
3040 : RETURN
3050 :
3060 : REM #####
```

Casse-pipes

PRESENTATION

Ce jeu est largement inspiré d'un stand de tir de fête foraine où plusieurs pipes se déplacent, et où le joueur doit, grâce à un fusil, abattre le plus de pipes possible.

Pour corser l'affaire, dans ce jeu, les pipes sont fixes et le joueur se déplace à vitesse constante.

Le nombre de balles n'est pas limité mais, à chaque partie gagnée, la vitesse de déplacement du tireur sera augmentée. Le but du jeu est d'abattre toutes les pipes avant de se trouver dans leur proximité immédiates.

A vos fusils...

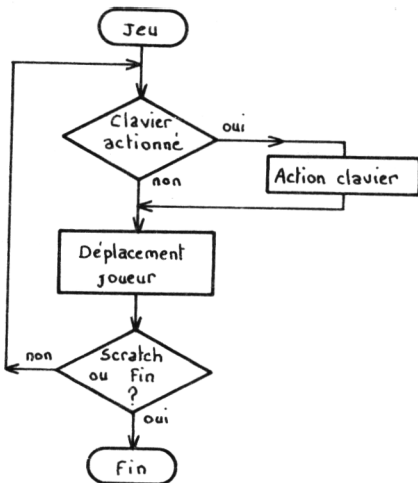
ORGANIGRAMME DU JEU

Le programme a la structure A.

Le déroulement d'une partie consiste en :

- . L'acquisition du tir joueur et tir
- . Le déplacement du joueur
- . La gestion de l'écran

L'organigramme suivant décrit cette structure :



LE PROGRAMME

Il se décompose en trois pavés :

Présentation Jeu

Lignes 1000 à 1540
 Lignes 2000 à 3270
 Module de commande Lignes 2000 à 2060
 Initialisation Lignes 2100 à 2480
 Partie Lignes 3000 à 3270

Fin de jeu

Lignes 4000 à 4140
 Commentaires sur la partie et activation éventuelle d'une autre partie.

LES VARIABLES UTILISEES

A\$	Code couleur de ligne
A	Lecture de données (Pipes/Fusil)
B\$	Caractère affiché (Pipe/Brique)
B	Caractère d'initialisation nombre de pipes
I, J	Index de boucles
NP	Nombre de pipes dans la partie

PAS Pas de progression verticale
 PE Numéro de partie
 SC Score joueur
 T Nombre de tirs
 X, Y Position du joueur sur l'écran.

```

10 : REM Casse-Pipes
20 :
30 : GOSUB 1000 'Presentation du jeu
40 : GOSUB 2000 'Jeu
50 : GOSUB 4000 'Fin du jeu
60 : IF FIN=0 THEN PRINT CHR$(17):GOTO 40
70 :
80 : END
90 : REM #####
1000 : REM Presentation du jeu
1010 :
1020 : PAPER 0 : INK 6 : PRINT CHR$(17)
1030 :
1040 : REM Titre
1050 :
1060 : PRINT CHR$(12)
1070 : PRINT CHR$(4);CHR$(27);"N";
1080 : PRINT CHR$(27);"A" CASSE - PIFES";CHR$(4)
1090 :
1100 : REM Objet
1110 :
1120 : PRINT:PRINT:PRINT:PRINT
1130 : A$="D" 'Code couleur = Bleu
1140 : PRINT"OBJET :";
1150 : GOSUB 1500:PRINT"Jeu de reflexe."
1160 :
1170 : REM Job
1180 :
1190 : PRINT:PRINT:PRINT
1200 : A$="F" 'Code couleur = Cyan
1210 :
1220 : PRINT"JOB :";
1230 : GOSUB 1500:PRINT"Le but du jeu est d'abattre"
1240 : GOSUB 1500:PRINT" un maximum de pipes en un temps."
;
1250 : GOSUB 1500:PRINT" fixe et dependant du talent du"
1260 : GOSUB 1500:PRINT" joueur."
1265 : PRINT
1270 : GOSUB 1500:PRINT" La touche SPACE permet "
1280 : GOSUB 1500:PRINT" d'activer un tir."

```

```

1290 :
1300 : REM Debut
1310 :
1320 : PRINT:PRINT:PRINT
1330 : A$="C" 'Code couleur = Jaune
1340 : PRINT"DEBUT :";
1350 : GOSUB 1500:PRINT"Pour commencer , appuyez sur"
1360 : GOSUB 1500:PRINT" une touche quelconque."
1370 :
1380 : GET A$ 'Attente de l'appui sur une touche
1390 :
1400 : REM Initialisation des caracteres du jeu
1410 :
1420 : PAS=1 'Pas de la progression
1430 : FOR I=0 TO 23
1435 : READ A:POKE 46760+I,A
1440 : NEXT I
1450 DATA 0,0,0,39,63,31,7,0
1460 DATA 0,0,21,42,21,42,0,0
1470 DATA 12,12,12,12,63,63,0,0
1480 :
1485 : RETURN
1490 : REM #####
1500 : REM Couleur d'une ligne
1510 :
1520 : PRINT CHR$(27);"H";CHR$(27);A$;
1530 :
1540 : RETURN
1550 : REM #####
2000 : REM Initialisation et jeu
2010 :
2020 : GOSUB 2100 'Initialisation d'une partie
2030 : GOSUB 3000 'Partie
2040 :
2050 : RETURN
2060 : REM #####
2100 : REM Initialisation d'une partie
2110 :
2120 : REM Titre
2130 :
2140 : CLS
2150 : PRINT CHR$(27);"L";CHR$(27);"C";
2160 : PRINT" CASSE - PIPES"
2170 : PE=PE+1:PRINT
2180 : PRINT" Partie No ";PE
2190 :
2200 : REM Tirage aleatoires des pipes

```

```

2210 :
2220 :   FOR I=1 TO 37 STEP 2
2230 :     B=INT(RND(1)*3)+1
2240 :     IF B=3 THEN B$="V" ELSE B$="U":NP=NP+1
2250 :     PLOT I,4,B$
2260 :   NEXT I
2270 :
2280 :   REM Curseur en position
2290 :
2300 :   FOR I=1 TO 20
2310 :     PRINT
2320 :   NEXT I
2330 :
2340 :   REM Commentaire Debut de jeu
2350 :
2360 :   PRINT CHR$(27);"L";CHR$(27);"A";
2370 :   PRINT"  Pressez une touche pour commencer."
2380 :   GET A$ 'Attente de l'appui sur une touche
2390 :   PRINT CHR$(11);CHR$(14)
2400 :
2410 :   REM Debut du jeu
2420 :
2430 :   PRINT"    Score : ";SC;"    Tir : ";T
2435 :   PRINT CHR$(11);
2440 :   X=1:Y=20:PLOT X,20,"M"
2450 :   REM Joueur au depart
2460 :
2470 :   RETURN
2480 :   REM #####
3000 :   REM Partie
3010 :
3020 :   REPEAT
3030 :     A$=KEY$
3040 :     IF A$(">") THEN GOSUB 3100 'Action joueur
3050 :     GOSUB 3200 'Deplacement joueur
3060 :   UNTIL Y<PAS+4 OR SC=NP
3070 :
3080 :   RETURN
3090 :   REM #####
3100 :   REM Action joueur
3110 :
3120 :   ZAP
3130 :   IF SCRNX(X,4)=85 THEN SC=SC+1:PLOT X,4," ":SHOOT
3140 :   T=T+1
3150 :   PRINT"    Score : ";SC;"    Tir : ";T
3155 :   PRINT CHR$(11);
3160 :
3170 :   RETURN

```

```
3180 : REM #####
3200 : REM Deplacement joueur
3210 :
3220 : PLOT X,Y," "
3230 : IF X=37 THEN Y=Y-PAS:X=X+1 ELSE X=X+2
3240 : PLOT X,Y,"M"
3250 :
3260 : RETURN
3270 : REM #####
4000 : REM Fin du jeu
4010 :
4020 : CLS
4030 : IF SC=NP THEN GOTO 4050
4035 : IF PAS>1 THEN PAS=PAS-1
4040 : PRINT"vous n'etes pas en forme dirait-on!!"
4045 : GOTO 4060
4050 : PRINT"pas mal,mais peut faire mieux."
4055 : PAS=PAS+1
4060 : PRINT:PRINT"Score : ";SC;" Tirs : ";T;
4070 : PRINT" Pipes : ";NP
4080 : PRINT:PRINT:PRINT CHR$(17)
4090 : INPUT"une autre partie ";R$
4100 : IF R$="0" THEN FIN=0 ELSE FIN=1
4110 : NP=0:SC=0:T=0
4120 :
4130 : RETURN
4140 : REM #####
```


Le piéton courageux

PRESENTATION

Vous croyez peut-être qu'il est simple de traverser la route ! Mais, si cette route comporte 4 voies, et si les voitures cherchent à vous écraser, penserez-vous la même chose ?

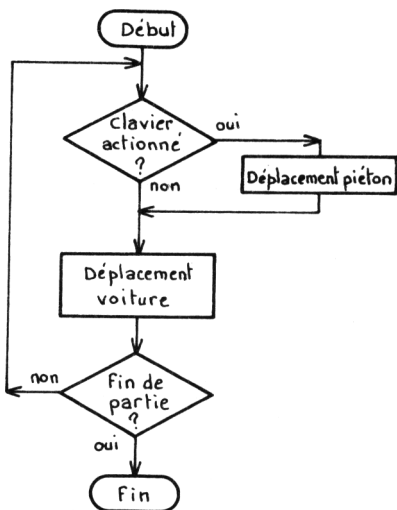
Ce jeu vous propose de prendre le rôle d'un piéton courageux qui n'hésite pas à traverser la route le plus souvent possible. Le temps est limité à 30 secondes. Mais la partie peut aussi s'interrompre avant les 30 secondes imparties... si vous vous faites écraser. Bon courage !

ORGANIGRAMME DU JEU

Le programme a la structure A.

Le jeu consiste simplement à déplacer piéton et voitures sur l'écran.

Il a la structure suivante :



LE PROGRAMME

Il se décompose en trois pavés :

Présentation et Initialisation

Lignes 1000 à 1950	
Module de commande	Lignes 1000 à 1080
Présentation	Lignes 1100 à 1550
Déf. voiture & piéton	Lignes 1600 à 1680
Déf. ct et var.	Lignes 1700 à 1780
Dessin écran	Lignes 1800 à 1950

Jeu

Lignes 2000 à 2610	
Module de commande	Lignes 2000 à 1070
Déplac. piéton cde	Lignes 2100 à 2170
Déplac. haut	Lignes 2200 à 2290
Déplac. bas	Lignes 2300 à 2370
Déplac. voiture	Lignes 2400 à 2610

Fin de jeu

Lignes 3000 à 3060	
Message de fin de partie.	

LES VARIABLES UTILISEES

A	Donnée de redéfinition de caractère, puis attente clavier
---	---

A\$	Code couleur de ligne
FIN	Indicateur de fin de partie
I, J	Index de boucle
PJ	Position du joueur
T	Tableau indiquant la position des voitures
VO	N° de voitures déplacées (1 à 4)
X	Nombre de traversées

```

10 : REM Le pieton courageux
20 :
30 : GOSUB 1000 'Initialisation
40 : GOSUB 2000 'Jeu
50 : GOSUB 3000 'Fin du jeu
60 :
70 : END
80 : REM *****
1000 : REM Initialisation
1010 :
1020 : GOSUB 1100 'Presentation
1030 : GOSUB 1600 'Definition des caracteres
1040 : GOSUB 1700 'Definition des constantes et variables
1050 : GOSUB 1800 'Dessin de la grille du jeu
1060 :
1070 : RETURN
1080 : REM *****
1100 : REM Presentation du jeu
1110 :
1120 : PAPER 0 : INK 5
1130 : REM Titre
1140 : PRINT CHR$(12)
1150 : PRINT CHR$(4);CHR$(27);"N";
1160 : PRINT CHR$(27);"A LE PIETON COURAGEUX";CHR$(4)
1170 : REM Objet
1180 : PRINT:PRINT:PRINT:PRINT
1190 : PRINT"OBJET :";
1200 : A$="D" 'Code couleur = Bleu
1210 : GOSUB 1500:PRINT"Jeu d'adresse."
1230 : REM Job
    
```

```

1240 : PRINT:PRINT:PRINT
1250 : A$="F" 'Code couleur = Cyan
1260 : PRINT"JOB  ";
1270 : GOSUB 1500:PRINT"Le but du jeu est de"
1280 : GOSUB 1500:PRINT"    traverser la route le plus de"
1290 : GOSUB 1500:PRINT"    fois possible en se servant des"
;
1300 : GOSUB 1500:PRINT"    des touches-fleches vers le bas"
;
1310 : GOSUB 1500:PRINT"    et vers le haut."
1320 : GOSUB 1500:PRINT"    Le temps est limite "
1330 : GOSUB 1500:PRINT"    a 30 secondes ."
1340 : REM Debut
1350 : PRINT:PRINT:PRINT
1360 : A$="C" 'Code couleur = Jaune
1370 : PRINT"DEBUT  ";
1380 : GOSUB 1500:PRINT"Pour commencer,appuyez sur"
1390 : GOSUB 1500:PRINT"    une touche quelconque ."
1395 : PRINT CHR$(17) 'Disparition du curseur
1400 : GET A$ 'Attente d'une action clavier
1410 : CLS:PRINT"SCORE":RETURN
1420 :
1430 : REM *****
1500 : REM Couleur d'une ligne
1510 :
1520 : PRINT CHR$(27);"H";CHR$(27);A$;
1530 :
1540 : RETURN
1550 : REM *****
1600 : REM Definition des caracteres
1610 :
1620 : FOR I=0 TO 15
1630 :   READ A:POKE 46600+I,A
1640 : NEXT I
1650 DATA 0,28,28,8,28,8,8,20,0,0,12,63,63,18,0,0
1660 :
1670 : RETURN
1680 : REM *****
1700 : REM Definition des constantes et variables du jeu
1710 :
1720 : V0=1 'No de voie pour voiture
1730 : T(1)=9:T(2)=9 'Voies 1 et 2 voitures a droite
1740 : T(3)=1:T(4)=1 'Voies 3 et 4 voitures a gauche
1750 : PJ=0 'Position joueur en bas
1760 :
1770 : RETURN

```

```

1780 : REM *****
1800 : REM Dessin de la grille du jeu
1810 :
1820 :   FOR I=1 TO 5
1830 :     FOR J=1 TO 19
1840 :       PLOT 10+J,9+2*I,"-"
1850 :     NEXT J
1860 :   NEXT I
1870 :   FOR I=1 TO 19
1880 :     PLOT 10+I,15,"="
1890 :   NEXT I
1900 :   PLOT 28,12,"B":PLOT 28,14,"B" 'Positionnement des voitures
1910 :   PLOT 12,16,"B":PLOT 12,18,"B"
1920 :   PLOT 20,20,"A" 'Positionnement du bonhomme
1930 :
1940 : RETURN
1950 : REM *****
2000 : REM Deroulement du jeu
2010 :
2020 :   A$=KEY$:IF A$(">") THEN GOSUB 2100 'Jeu du joueur
2030 :   GOSUB 2400 'Avancement voitures
2040 :   IF FIN=0 THEN GOTO 2020 ELSE RETURN
2050 :
2060 : RETURN
2070 : REM *****
2100 : REM Deplacement du pieton
2110 :
2120 :
2130 :   IF ASC(A$)=11 THEN GOTO 2200
2140 :   IF ASC(A$)=10 THEN GOTO 2300
2150 :
2160 : RETURN
2170 : REM *****
2200 : REM Deplacement vers le haut
2210 :
2220 :   PJ=PJ+1
2230 :   IF SCRN(20,20-2*PJ)<>32 THEN FIN=1 'Collision
2240 :   IF PJ<>5 THEN GOTO 2270
2250 :   PLOT 20,12," ":PJ=0:SC=SC+1
2260 :   PLOT20,20,"A":ZAP:X=X+1:PRINT " ";X:RETURN
2270 :   PLOT 20,20-2*PJ,"A":PLOT20,22-PJ*2," ":RETURN
2280 :

```

```

2290 : REM *****
2300 : REM Deplacement vers le bas
2310 :
2320 : PJ=PJ-1
2330 : IF SCRN(20,20-PJ*2)<>32 THEN FIN=1 'Collision
2340 : IF PJ=0 THEN PJ=1:RETURN
2345 : IF PJ=-1 THEN PJ=0
2350 : PLOT 20,20-PJ*2,"A":PLOT20,18-PJ*2," ":RETURN
2360 :
2370 : REM *****
2400 : REM Deplacement voiture
2410 :
2415 : TE=TE+1:IF TE=170 THEN FIN=1 'Ecoulement du temps imparté
2420 : IF VO=1 OR VO=2 THEN GOTO 2520 'Deplacement vers la gauche
2430 : REM Deplacement vers la droite
2440 : IF T(VO)<>9 THEN GOTO 2480
2450 : PLOT 28,10+2*VO," ":PLOT 12,10+2*VO,"B":T(VO)=1
2460 : VO=INT(RND(1)*4)+1
2470 : RETURN
2480 : IF T(VO)=4 AND SCRN(20,10+2*VO)=65 THEN FIN=1 'Collision
2490 : PLOT 10+T(VO)*2,10+VO*2," ":T(VO)=T(VO)+1
2495 : PLOT10+T(VO)*2,10+VO*2,"B"
2500 : VO=INT(RND(1)*4)+1
2510 : RETURN
2520 : REM Deplacement vers la gauche
2530 : IF T(VO)<>1 THEN GOTO 2560
2540 : PLOT 12,10+2*VO," ":T(VO)=9:PLOT 28,10+2*VO,"B"
2550 : VO=INT(RND(1)*4)+1:RETURN
2560 : IF T(VO)=6 AND SCRN(20,10+2*VO)=65 THEN FIN=1 'Collision
2570 : PLOT10+T(VO)*2,10+VO*2," ":T(VO)=T(VO)-1
2580 : PLOT10+T(VO)*2,10+VO*2,"B"
2590 : VO=INT(RND(1)*4)+1:RETURN
2600 :
2610 : REM *****
3000 : REM Fin de la partie
3010 :
3020 : EXPLODE
3025 : PLOT 0,22," " 'Code couleur blanche
3030 : PLOT 11,22,"That's all folks !"
3040 : PRINT CHR$(17) 'Restitution du curseur
3050 :
3060 : RETURN

```

Réflexe

PRESENTATION

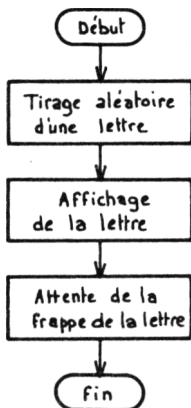
Ce jeu est à conseiller à toute personne qui a du mal à mémoriser la disposition des touches sur le clavier. En effet, ORIC affiche une lettre sur l'écran, et vous devez frapper cette lettre le plus rapidement possible. Avec un peu de pratique, vous arriverez à connaître le clavier ou, à devenir fou...

ORGANIGRAMME DU JEU :

Le programme a la structure A.

Le jeu est très simple. Il suffit de tirer aléatoirement une lettre du clavier, de l'afficher, et d'attendre la réponse du joueur.

L'organigramme du jeu est donc le suivant :



LE PROGRAMME

Il se décompose en trois pavés :

Présentation du jeu Lignes 1000 à 1450

Jeu Lignes 2000 à 2200

Fin du jeu Lignes 3000 à 3100
Commentaires sur le jeu et activation éventuelle d'une autre partie.

```

10 : REM Reflexe
20 :
30 : GOSUB 1000 'Presentation
40 : GOSUB 2000 'Jeu
50 : GOSUB 3000 'Fin du jeu
60 : IF R$="D" THEN GOTO 40 'Nouvelle partie
70 :
80 : PRINT CHR$(17):END
90 : REM #####
1000 : REM Presentation du jeu
1010 :
1020 : INK 5 ; PAPER 0 : PRINT CHR$(17)
1030 : REM Titre
1040 :
1050 : PRINT CHR$(12)
1060 : PRINT CHR$(4);CHR$(27);"N";
1070 : PRINT CHR$(27);"A REFLEXE";CHR$(4)
1080 :
1090 : REM Objet
1100 :
1110 : PRINT:PRINT:PRINT:PRINT
1120 : A$="D" 'Code couleur = Bleu
1130 : PRINT"OBJET :";
1140 : GOSUB 1400:PRINT"Jeu de rapidite."
1150 :
1160 : REM Job
1170 :
1180 : PRINT:PRINT:PRINT:PRINT
1190 : A$="F" 'Code couleur = Cyan
1200 : PRINT"JOB : ";
1210 : GOSUB 1400:PRINT"Une lettre va apparaitre sur";
1220 : GOSUB 1400:PRINT" l'ecran . Appuyez sur la touche"
;
1230 : GOSUB 1400:PRINT" correspondante en un minimum"
1235 : GOSUB 1400:PRINT" de temps."
1240 :
1250 : REM Debut
1260 :

```



```

1270 : PRINT:PRINT:PRINT:PRINT
1280 : A$="C" 'Code couleur = Jaune
1290 : PRINT"DEBUT : ";
1300 : GOSUB 1400:PRINT"Pour commencer , appuyez sur";
1310 : GOSUB 1400:PRINT" une touche quelconque."
1320 : GET A$ 'Attente de l'appui sur une touche
1330 :
1340 : RETURN
1350 : REM *****
1400 : REM Couleur d'une ligne
1410 :
1420 : PRINT CHR$(27);"H";CHR$(27);A$;
1430 :
1440 : RETURN
1450 : REM *****
2000 : REM Jeu
2010 :
2020 : REM Tirage aleatoire d'une lettre du clavier
2030 :
2040 : A=INT(RND(1)*26)+65
2050 :
2060 : REM Affichage de la lettre tiree
2070 :
2075 : CLS
2080 : FOR I=1 TO 11
2090 : PRINT
2100 : NEXT I
2110 : PRINT CHR$(4);CHR$(27);"J";CHR$(27);"A
CHR$(A);
2120 : PRINT CHR$(4)
2130 :
2140 : REM Attente de l'appui sur la lettre affichee
2150 :
2160 : B$=KEY$
2170 : I=I+1:IF B$="" THEN GOTO 2160
2180 :
2190 : RETURN
2200 : REM *****
3000 : REM Fin du jeu , Affichage des resultats
3010 :
3020 : CLS
3030 : IF B$(>)CHR$(A) THEN PRINT"Il faudra acheter des lunettes!!
.
3040 : PRINT:PRINT
3050 : PRINT"Temps de reflexion :";T
3060 : PRINT:PRINT:PRINT
3070 : GOSUB 1400:INPUT"Un autre essai (O/N) ";R$:T=0
3080 :
3090 : RETURN
3100 : REM *****

```


JEUX DE HASARD

Bataille navale

PRESENTATION

Le célèbre jeu de bataille navale est repris par ce programme. Votre adversaire est ORIC. La grille du jeu fait 9 cases de côté.

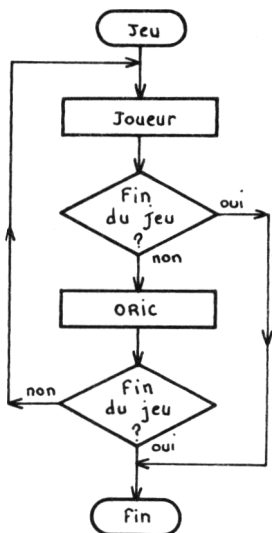
Vous disposez de 6 navires occupant chacun une case sur la grille. Pour découvrir les navires ennemis, vous disposez d'un nombre illimité de torpilles qui sont envoyées à tour de rôle.

ORGANIGRAMME DU JEU

Le programme a la structure A.

Le jeu est initialisé, puis la partie est engagée.

A tour de rôle, vous, puis ORIC tentez de détruire les navires ennemis. La partie se termine quand tous les navires d'un joueur sont coulés.



LE PROGRAMME

Il se décompose en trois pavés :

Présentation

Lignes 1000 à 1550

Jeu

Lignes 2000 à 3160

Module de commande Lignes 2000 à 2060

Initialisation Lignes 2100 à 2530

Partie Lignes 2100 à 3160

Fin de partie

Lignes 4000 à 4160

LES VARIABLES UTILISEES

A, B	Codes entrés par le joueur, puis jeu du joueur
A\$	Code couleur de ligne
CJ, CO	Navires joueur coulés et ORIC coulés
I	Index de boucles
FIN	Indicateur de fin de partie
O, ESSAI	Tableaux de jeu
X, Y	Coordonnées sur tableau

```

10 : REM Bataille Navale
20 :
30 : GOSUB 1000 'Presentation du jeu
40 : GOSUB 2000 'Jeu
50 : GOSUB 4000 'Fin du jeu
60 : IF BIS=1 THEN GOTO 40
70 :
80 : END
90 : REM #####
1000 : REM Presentation du jeu
1010 :
1020 : PAPER 0 : INK 5 : PRINT CHR$(17)
1030 :
1040 : REM Titre
1050 :
1060 : PRINT CHR$(12)
1070 : PRINT CHR$(4);CHR$(27);"N";
1080 : PRINT CHR$(27);"A          BATAILLE NAVALE";CHR$(4)
1090 :
1100 : REM Objet
1110 :
1120 : PRINT:PRINT:PRINT
1130 : A$="D" 'Code couleur = Bleu
1140 : PRINT"OBJET : ";
1150 : GOSUB 1500:PRINT"Jeu de hasard."
1160 :
1170 : REM Job
1180 :
1190 : PRINT:PRINT
1200 : A$="F" 'Code couleur = Cyan
1210 : PRINT"JOB : ";
1220 : GOSUB 1500:PRINT"Le but du jeu est de"
1230 : GOSUB 1500:PRINT" decouvrir des navires ennemis"
1240 : GOSUB 1500:PRINT" se trouvant sur une grille "

1250 : GOSUB 1500:PRINT" 9*9 avant que l'ennemi ne"
1260 : GOSUB 1500:PRINT" decouvre les votres."
1270 : GOSUB 1500:PRINT" Vous disposez tous deux de 6"
1280 : GOSUB 1500:PRINT" navires occupant chacun une"
1290 : GOSUB 1500:PRINT" place sur la grille."
1300 : GOSUB 1500:PRINT" Pour decouvrir les navires "
1310 : GOSUB 1500:PRINT" ennemis,vous disposez de "
1320 : GOSUB 1500:PRINT" torpilles que vous pouvez"
1330 : GOSUB 1500:PRINT" lancer sur une case quelconque"
1340 : GOSUB 1500:PRINT" de la grille."
1350 :

```

```

1360 : REM Debut
1370 :
1380 : PRINT:PRINT
1390 : A$="C" 'Code couleur = Jaune
1400 : PRINT"DEBUT : ";
1410 : GOSUB 1500:PRINT"Pour commencer , appuyez sur";
1420 : GOSUB 1500:PRINT" une touche quelconque."
1430 :
1440 : GET A$ 'Attente de l'appui sur une touche
1445 : DIM O(9,9),ESSAI(9,9) 'Tableaux de jeu
1450 :
1460 : RETURN
1470 : REM *****
1500 : REM Couleur d'une ligne
1510 :
1520 : PRINT CHR$(27);"H";CHR$(27);A$;
1530 :
1540 : RETURN
1550 : REM *****
2000 : REM Jeu
2010 :
2020 : GOSUB 2100 'Initialisation de la partie
2030 : GOSUB 2600 'Partie
2040 :
2050 : RETURN
2060 : REM *****
2100 : REM Initialisation de la partie
2110 :
2120 : REM Avertissement au joueur
2130 :
2140 : CLS : INK 6
2150 : PRINT CHR$(27);"L";CHR$(27);"C";
2160 : PRINT" INITIALISATION DE LA PARTIE"
2170 : PRINT:PRINT:PRINT:PRINT:PRINT
2180 : PRINT" Dessinez une grille 9*9 puis"
2190 : PRINT"placez sur cette grille 6 bateaux de "
2200 : PRINT"dimension 1 case."
2210 : PRINT:PRINT:PRINT
2220 : PRINT" Tapez ensuite sur une touche"
2230 : PRINT"quelconque pour commencer a jouer."
2240 :
2250 : REM Initialisation ORIC
2260 :
2270 : FOR I=1 TO 6
2280 : X=INT(RND(1)*9)+1
2290 : Y=INT(RND(1)*9)+1
2300 : IF O(X,Y)=0 THEN O(X,Y)=1 ELSE I=I-1
2310 : NEXT I
2320 :

```

```

2330 : REM Attente de l'appui sur une touche
2340 :
2350 : GET A$
2360 :
2370 : CLS
2380 : PRINT CHR$(27);"L";CHR$(27);"B";
2390 : PRINT"          BATAILLE NAVALE"
2400 : PRINT:PRINT:PRINT:PRINT:PRINT:PRINT:PRINT
2410 : PLOT 10,6,"Votre grille de tir"
2420 : PRINT"          A B C D E F G H I J":PRINT
2430 : FOR I=1 TO 9
2440 :   PRINT "          ";CHR$(I+48)
2450 : NEXT I
2490 :
2500 : PRINT:PRINT:PRINT CHR$(17)
2510 :
2520 : RETURN
2530 : REM #####
2600 : REM La partie est engagee
2610 :
2620 : GOSUB 2700 'Jeu du joueur'
2630 : IF FIN THEN RETURN
2640 : GOSUB 3000 'Jeu d'ORIC
2650 : IF FIN THEN RETURN ELSE GOTO 2600
2660 :
2670 : REM #####
2700 : REM Jeu du joueur
2710 :
2720 : INPUT"Entrez votre jeu";J$
2730 : A=ASC(LEFT$(J$,1))-64
2740 : B=ASC(RIGHT$(J$,1))-48
2750 : IF O(B,A)=0 THEN GOSUB 2800 'A l'eau
2760 : IF O(B,A)=1 THEN GOSUB 2900 'Coule
2764 : WAIT 260
2765 : PRINT CHR$(11);CHR$(14);CHR$(14);CHR$(11);CHR$(11);
2770 :
2780 : RETURN
2790 : REM #####
2800 : REM Joueur a l'eau
2810 :
2820 : PRINT"          A l'eau !!"
2830 : PLOT 2%A+8,B+9,"O"
2840 :
2850 : RETURN

```



```

2860 : REM #####
2900 : REM Joueur a coule ORIC
2910 :
2920 : PRINT"      Coule.!      Bien joue!!"
2930 : PLOT 2#A+B,B+9,"X"
2935 : O(B,A)=2 'Interdiction de rejouer le meme coup
2940 : CJ=CJ+1:IF CJ=6 THEN FIN=1
2950 :
2960 : RETURN
2970 : REM #####
3000 : REM Jeu d'ORIC
3010 :
3020 : X=INT(RND(1)*9)+1
3030 : Y=INT(RND(1)*9)+1
3040 : IF ESSAI(X,Y)<>0 THEN GOTO 3000
3050 : ESSAI(X,Y)=1
3060 : A%=CHR$(Y+64)+CHR$(X+48)
3070 : PRINT"Je joue en ";A%
3080 : PRINT"Entrez le resultat du tir : "
3090 : INPUT"1=A l'eau 2=Coule ";A
3100 : IF A=2 THEN CO=CO+1
3105 : IF CO=6 THEN FIN=1
3110 : PRINT CHR$(11);CHR$(11);CHR$(11);CHR$(14)
3120 : PRINT CHR$(14)
3130 : PRINT CHR$(14);CHR$(11);
3140 :
3150 : RETURN
3160 : REM #####
4000 : REM Fin de la partie
4010 :
4020 : CLS
4030 : IF CO=6 THEN PRINT"Pas assez de tactique."
4040 : IF CJ=6 THEN PRINT"Pas mal.Il fallait le faire."
4050 : IF CJ=6 THEN PRINT"J'espere que vous n'avez pas triche."
4060 : PRINT:PRINT
4070 : INPUT"Une autre partie (O/N) ";R%
4080 : IF R%="O" THEN BIS=1:PRINT CHR$(17) ELSE BIS=0
4090 : CO=0:CJ=0:FIN=0
4100 : FOR I=1 TO 9
4110 :   FOR J=1 TO 9
4120 :     O(I,J)=0
4130 :   NEXT J
4140 : NEXT I
4150 :
4160 : RETURN

```

Jack-pot

PRESENTATION

Le célèbre jeu de café est repris, de façon simplifiée : trois roulettes représentant des fruits (poire, banane, pomme, cerise) sont animées d'un mouvement de rotation. Lorsqu'elles s'arrêtent :

- les trois roulettes représentent des fruits différents :
la mise est perdue
- deux des roulettes représentent le même fruit :
DOUBLE : vous gagnez 10 \$
- les trois roulettes indiquent le même fruit :
JACK POT : vous gagnez 30 \$.

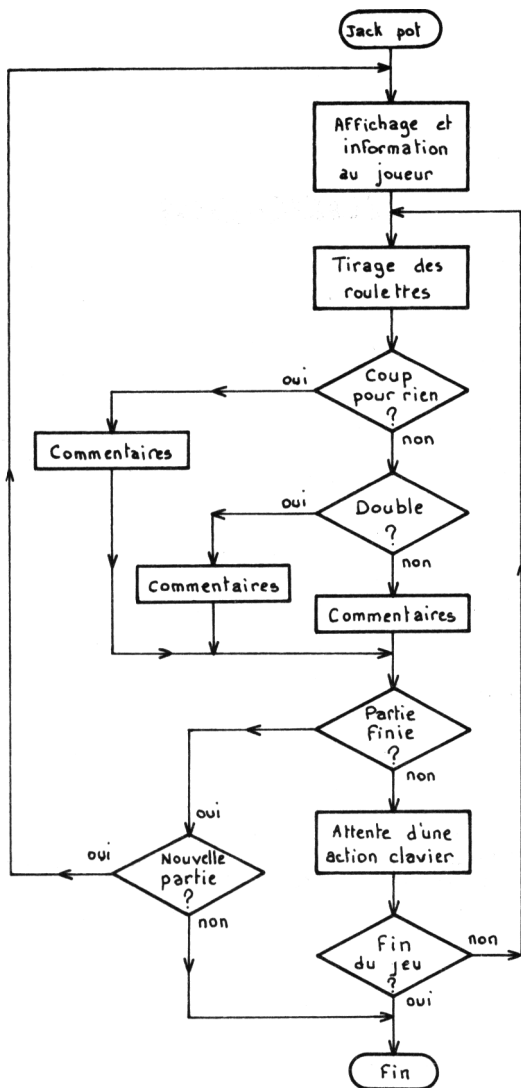
Ce jeu, très simple, peut devenir passionnant, surtout si vous êtes dans un jour de chance.

ORGANIGRAMME DU JEU

Le programme a la structure A.

Une grande partie du jeu est consacrée au tirage aléatoire des fruits et à leur affichage. Ensuite, il suffit de tester les trois configurations possibles : coup pour rien, DOUBLE, JACK POT.

L'organigramme page suivante représente les différentes parties du jeu



LE PROGRAMME

Il se décompose en trois pavés :

**Présentation et
initialisation**

Lignes 1000 à 1960	
Module de commande	Lignes 1000 à 1060
Présentation	Lignes 1100 à 1650
Initialisation	Lignes 1700 à 1960

Jeu

Lignes 2000 à 3160	
Commentaires	Lignes 2000 à 2360
Tirage aléatoire	Lignes 2370 à 2700
Résultats	Lignes 2700 à 3160

Fin de partie

Lignes 4000 à 4090	
Commentaires sur la partie	
Activation éventuelle d'une autre partie.	

LES VARIABLES UTILISEES

A	Donnée de définition des caractères graphiques
A\$	Code couleur de ligne
C1, C2, C3	Fruit sur 1ère, 2ème et 3ème roulette
E	Fruit à afficher
FIN	Code de fin de partie
I	Index de boucle
INFO\$	Commentaires sur un coup
S	Somme disponible
T\$	Tableau des fruits

```

10 : REM Jack Pot
20 :
30 : GOSUB 1000 'Presentation du jeu et Initialisation
40 : GOSUB 2000 'Jack Pot
50 : GOSUB 4000 'Fin de la partie
60 : IF R$="0" THEN GOTO 40
65 : PRINT CHR$(17)
70 :
80 : END
90 : REM *****
1000 : REM Presentation du jeu et Initialisation
1010 :
1020 : GOSUB 1100 'Presentation du jeu
1030 : GOSUB 1700 'Initialisation
1040 :
1050 : RETURN
1060 : REM *****
1100 : REM Presentation du jeu
1110 :
1120 : PAPER 0 : INK 5 : PRINT CHR$(17)
1130 :
1140 : REM Titre
1150 :
1160 : PRINT CHR$(12)
1170 : PRINT CHR$(4);CHR$(27);"N";
1180 : PRINT CHR$(27);"A JACK POT";CHR$(4)
1190 :
1200 : REM Objet
1210 :
1220 : PRINT:PRINT:PRINT:PRINT
1230 : A$="D" 'Code couleur = Bleu
1240 : PRINT"OBJET : ";
1250 : GOSUB 1600:PRINT"Jeu de hasard."
1260 :
1270 : REM Job
1280 :
1290 : PRINT:PRINT:PRINT
1300 : A$="F" 'Code couleur = Cyan
1310 : PRINT"Job : ";
1320 : GOSUB 1600:PRINT"Le jeu consiste a lancer 3"
1330 : GOSUB 1600:PRINT" roulettees comportant chacune:"
1335 : PRINT
1340 : GOSUB 1600:PRINT" - Une poire"
1350 : GOSUB 1600:PRINT" - Une banane"
1360 : GOSUB 1600:PRINT" - Une pomme"
1370 : GOSUB 1600:PRINT" - Une cerise"
1375 : PRINT
1380 : GOSUB 1600:PRINT" et a obtenir les memes fruits"
1390 : GOSUB 1600:PRINT" sur chaque roulette a l'arret."

```

```
1400 :
1410 : REM Debut
1420 :
1430 : PRINT:PRINT:PRINT
1440 : A$="C" 'Code couleur = Jaune
1450 : PRINT"DEBUT :";
1460 : GOSUB 1600:PRINT"Pour commencer,appuyez sur"
1470 : GOSUB 1600:PRINT"      une touche quelconque."
1480 :
1490 : GET A$ 'Attente de l'appui sur une touche
1500 :
1510 : RETURN
1520 : REM *****
1600 : REM Couleur d'une ligne
1610 :
1620 : PRINT CHR$(27);"H";CHR$(27);A$;
1630 :
1640 : RETURN
1650 : REM *****
1700 : REM Initialisation des caracteres du jeu
1710 :
1720 : FOR I=0 TO 31
1730 :   READ A:POKE 46360+I,A
1740 : NEXT I
1750 :
1760 : RETURN
1770 : REM *****
1800 : REM Banane
1810 :
1820 DATA 0,0,0,0,51,63,30,0
1830 :
1840 : REM Paire
1850 :
1860 DATA 4,4,12,30,30,63,63,30
1870 :
1880 : REM Pomme
1890 :
1900 DATA 4,4,30,63,63,63,30,0
1910 :
1920 : REM Cerise
1930 :
1940 DATA 0,1,2,4,24,60,60,24
1950 :
```

```

1960 : REM *****
2000 : REM Jack Pot
2010 :
2020 :   INK 6 : CLS : S=100
2030 :
2040 :   REM Titre
2050 :
2060 :   PRINT CHR$(27);"H";CHR$(27);"D";
2070 :   PRINT"      J A C K P O T"
2080 :
2090 :   REM Configuration Initiale
2100 :
2110 :   PLOT 10,12,"-----"
2120 :   . PLOT 10,13,";           ;"
2130 :   PLOT 10,14,";           ;"
2140 :   PLOT 10,15,";           ;"
2150 :   PLOT 10,16,"-----"
2160 :   T$(1)=CHR$(27)+"H"+CHR$(27)+"C#"
2170 :   T$(2)=CHR$(27)+"H"+CHR$(27)+"B#"
2180 :   T$(3)=CHR$(27)+"H"+CHR$(27)+"AZ"
2190 :   * T$(4)=CHR$(27)+"H"+CHR$(27)+"A#"
2195 :   E$=CHR$(27)+"H"+CHR$(27)+"F"
2200 :
2210 :   FOR I=1 TO 13
2220 :     PRINT
2230 :   NEXT I
2240 :   PRINT"      ;";T$(1);" " ;T$(2);" " ;T$(3);E$
2250 :
2260 :   REM Information au joueur
2270 :
2280 :   PRINT:PRINT:PRINT:PRINT:PRINT
2290 :   A$="C":GOSUB 1600;
2300 :   PRINT" Pour jouer une nouvelle fois,"
2310 :   GOSUB 1600:PRINT" appuyez sur une touche quelconque."
2313 :   PRINT
2314 :   GOSUB 1600:PRINT" Pour arreter,appuyez sur RETURN."
2320 :
2330 :   FOR I=1 TO 9
2340 :     PRINT CHR$(11);
2350 :   NEXT I
2360 :
2370 :   REM Jeu
2380 :
2390 :   GET A$
2395 :   IF ASC(A$)=13 THEN FIN=1
2400 :
2410 :   REM 1ere case
2420 :

```

```
2430 :     FOR I=1 TO 20
2440 :         PRINT CHR$(11);
2450 :         E=INT(RND(1)*4)+1
2460 :         PRINT"          ;"+T$(E)
2470 :         WAIT 10
2480 :     NEXT I
2490 :     F=T$(E):C1=E
2500 :
2510 :     REM 2eme case
2520 :
2530 :     FOR I=1 TO 20
2540 :         PRINT CHR$(11);
2550 :         E=INT(RND(1)*4)+1
2560 :         PRINT"          ;"+F$+" "+T$(E)
2570 :         WAIT 10
2580 :     NEXT I
2590 :     F=F$+" "+T$(E):C2=E
2600 :
2610 :     REM 3eme case
2620 :
2630 :     FOR I=1 TO 20
2640 :         PRINT CHR$(11);
2650 :         E=INT(RND(1)*4)+1
2660 :         PRINT"          ;"+F$+" "+T$(E)
2670 :         WAIT 10
2680 :     NEXT I
2690 :     C3=E
2700 :
2710 :     REM Fin du tirage aleatoire
2720 :     REM Comptabilisation des gains
2730 :
2740 :     IF C1<>C2 AND C1<>C3 THEN D=1
2750 :     IF C1=C2 OR C1=C3 OR C2=C3 THEN D=2
2760 :     IF C1=C2 AND C1=C3 THEN D=3
2770 :
2780 :     ON D GOSUB 2900,3000,3100
2790 :     FOR I=1 TO 10
2795 :         PRINT CHR$(11);
2800 :     NEXT I
2804 :     PRINT CHR$(14);CHR$(11)
2805 :     PRINT SPC(7);INFO$;
2810 :     PRINT "   Score : ";S
2811 :     FOR I=1 TO 9
2812 :         PRINT
2813 :     NEXT I
2820 :
2830 :     IF S<0 OR FIN=1 THEN RETURN ELSE GOTO 2370
2840 :
```



```
2860 : REM #####
2900 : REM Coup pour rien
2910 :
2920 : INFO$="Coup pour rien"
2930 : S=S-10
2940 :
2950 : RETURN
2960 : REM #####
3000 : REM Double
3010 :
3020 : INFO$="Double !"
3030 : S=S+10
3040 :
3050 : RETURN
3060 : REM #####
3100 : REM Jack Pot
3110 :
3120 : INFO$="Jack Pot !!!"
3130 : S=S+30
3140 :
3150 : RETURN
3160 : REM #####
4000 : REM Fin de partie
4010 :
4020 : CLS
4030 : PRINT:PRINT
4040 : PRINT"La partie est finie."
4050 : PRINT:PRINT
4060 : INPUT"En ferez-vous une autre (O/N) ";R$
4070 : FIN=O:S=100
4080 :
4090 : RETURN
```

Jeu du 21

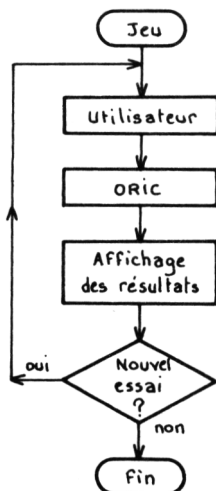
PRESENTATION

Le but du jeu est de se rapprocher au maximum du score 21 par tir de dés répétés sans jamais le dépasser.

ORIC joue contre vous et a les mêmes objectifs.

Saurez-vous vous arrêter à temps ?

ORGANIGRAMME DU JEU



LE PROGRAMME

Il se décompose en trois pavés :

Initialisation	Lignes 1000 à 1550 C'est la présentation du programme à l'utilisateur.
Jeu	Lignes 2000 à 2450 Le jeu se décompose en : <ul style="list-style-type: none"> - Jeu utilisateur Lignes 2100 à 2240 - Jeu ORIC Lignes 2300 à 2360 La procédure de tir aléatoire est commune aux jeux utilisateur et ORIC. Lignes 2500 à 2630 Affichage des résultats du joueur et d'ORIC. Initialisation éventuelle d'une nouvelle partie.
Fin du jeu	Lignes 3000 à 3040 Restitution du curseur et arrêt du programme.

LES VARIABLES UTILISEES

A	Représente le tir de dés
AR	Argent disponible pour le joueur
A\$	Caractère action joueur
MISE	Argent misé par le joueur
OC	Total ORIC
PERDU	Variable booléenne de test de fin de partie
USER	Total utilisateur

```

10 : REM Jeu du 21
20 :
30 : GOSUB 1000 'Initialisation
40 : GOSUB 2000 'Jeu
50 : GOSUB 3000 'Fin du jeu
60 :
70 : END
80 : REM #####
1000 : REM Initialisation
1010 :
1020 : PAPER 0 : INK 5 : PRINT CHR$(17)
1030 : REM Titre
1040 : PRINT CHR$(12)
1050 : PRINT CHR$(4);CHR$(27);"N";
1060 : PRINT CHR$(27);"A JEU DU 21";CHR$(4)
1070 :
1080 : REM objet
1090 : PRINT:PRINT:PRINT:PRINT
1100 : PRINT"OBJET :";
1110 : A$="D" 'Code couleur = Bleu
1120 : GOSUB 1500:PRINT"Jeu de hasard."
1130 :
1140 : REM Job
1150 : PRINT:PRINT:PRINT
1160 : A$="F" 'Code couleur = Cyan
1170 : PRINT"JOB :";
1180 : GOSUB 1500:PRINT"Vous possédez 100 francs."
1190 : GOSUB 1500:PRINT" Misez une somme inferieure a "
1200 : GOSUB 1500:PRINT" 100 francs . Lancez le de "
1210 : GOSUB 1500:PRINT" autant de fois que vous le "
1220 : GOSUB 1500:PRINT" voulez pour vous rapprocher"
1230 : GOSUB 1500:PRINT" au maximum du score 21 ."
1240 : GOSUB 1500:PRINT" ORIC se mesurera alors a vous."
1250 :
1260 : REM Debut
1270 :
1280 : PRINT:PRINT:PRINT
1290 : A$="C" 'Code couleur = Jaune
1300 : PRINT"DEBUT :";
1310 : GOSUB 1500:PRINT"Pour commencer, appuyez"
1320 : GOSUB 1500:PRINT" sur une touche quelconque."
1330 : GET A$ 'Attente de l'appui sur une touche
1340 :
1350 : REM Initialisation des variables
1360 :
1370 : AR=100 'Somme disponible
1380 :
1390 :
1400 : RETURN

```

```

1410 : REM #####
1500 : REM Couleur d'une ligne
1510 :
1520 : PRINT CHR$(27);"H";CHR$(27);A$;
1530 :
1540 : RETURN
1550 : REM #####
2000 : REM Jeu
2010 :
2015 : OC=0;USER=0
2020 : GOSUB 2100 'Jeu utilisateur
2030 : GOSUB 2300 'Jeu ORIC
2035 : GOSUB 2500 'Affichage des resultats
2040 : INPUT "Nouvel essai (O/N)";R$
2050 : IF R$="O" THEN GOTO 2000 ELSE RETURN
2060 :
2070 : REM #####
2100 : REM Jeu de l'utilisateur
2110 :
2120 : CLS
2130 : INPUT "Mise ";MISE
2140 : PRINT:PRINT:PRINT "Pour obtenir un nouveau tir, appuyez"
2150 : PRINT "sur T et pour faire jouer ORIC"
2160 : PRINT "appuyez sur F."
2170 : GET A$ : IF A$(">")="T" THEN GOTO 2210
2180 : GOSUB 2400
2190 : USER=USER+A
2200 : IF USER>21 THEN PERDU=1:RETURN
2205 : PRINT "Tir : ";A$; " Total :";USER
2210 : IF A$="F" THEN RETURN
2220 : GOTO 2170
2230 :
2240 : REM #####
2300 : REM Jeu ORIC
2310 :
2320 : GOSUB 2400
2330 : OC=OC+A
2340 : IF OC>19 THEN RETURN ELSE GOTO 2300
2350 :
2360 : REM #####
2400 : REM Tir aleatoire d'une variable A
2410 : .
2420 : A=INT(RND(1)*6)+1
2430 :
2440 : RETURN

```

```
2450 : REM *****
2500 : REM Affichage des resultats
2510 :
2520 : PRINT"Votre score :";USER
2530 : PRINT"Votre mise :";MISE
2540 : IF PERDU <> 1 THEN GOTO 2590
2550 : PRINT"Perdu par depassement!!"
2560 : AR=AR-MISE
2570 : PRINT"Nouveau score : ";AR
2580 : PERDU=0:RETURN
2590 : PRINT"Score ORIC : ";OC
2600 : IF OC<USER OR OC>21 THEN PRINT"Bien joue":AR=AR+MISE
2610 : IF OC>USER AND OC<=21 THEN AR=AR-MISE:PRINT"Perdu.Il reste
";AR
2620 : IF OC=USER THEN PRINT"Egalite !!"
2630 : RETURN
3000 : REM Fin du jeu
3010 :
3020 : PRINT CHR$(17)
3030 :
3040 : RETURN
```

JEUX DE RÉFLEXION

Jeu des allumettes

PRESENTATION

Ce jeu très simple (à deux joueurs ou plus) n'est pourtant pas dénué d'intérêt. Un nombre N d'allumettes est déposé sur une table. Chaque joueur retire, à tour de rôle, 1, 2 ou 3 allumettes. Celui qui retire la ou les dernières allumettes a gagné. Ici, il y a deux joueurs : vous et ORIC.

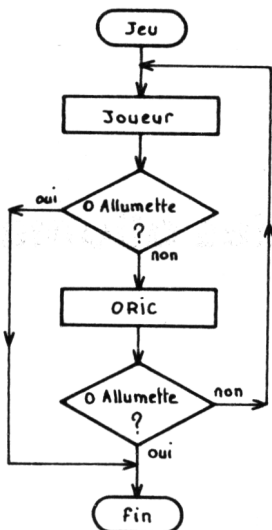
Saurez-vous le battre à coup sûr ?

ORGANIGRAMME DU JEU

Le programme a la structure A.

Les deux joueurs (en l'occurrence vous et ORIC) jouent jusqu'à ce que la dernière allumette ait été retirée. C'est vous qui choisissez le nombre d'allumettes au départ.

Le jeu correspond à l'organigramme suivant :



La tactique d'ORIC est fort simple :

S'il reste plus de 7 allumettes, il en retire un nombre quelconque compris entre 1 et 3.

S'il en reste 7 ou moins :

- Il en reste 7 : ORIC en prend 3 et gagne en 2 coups.
- Il en reste 6 : ORIC en prend 2 et gagne en 2 coups.
- Il en reste 5 : ORIC en prend 1 et gagne en 2 coups.
- Il en reste 4 : ORIC en prend 1 et vous gagnez.
- Il en reste 3 ou moins : ORIC gagne en 1 coup.

LE PROGRAMME

Il se décompose en trois pavés :

Initialisation

Lignes 1000 à 1700	
Module de commande	Lignes 1000 à 1060
Présentation	Lignes 1100 à 1540
Init. du jeu	Lignes 1600 à 1690

Jeu

Lignes 2000 à 2160
La validité du jeu est testée
Lignes 2040 et 2050

Fin du jeu

Lignes 3000 à 3030
La partie est rapidement commentée.

LES VARIABLES UTILISEES

A\$	Code couleur de ligne
GAGN	Indique le vainqueur
N	Nombre d'allumettes restantes
TIR	Nombre d'allumettes retirées par ORIC
USER	Nombre d'allumettes retirées par le joueur.

Remarque :

Avec un peu d'entraînement, vous arriverez sans doute à battre ORIC à coup sûr.

Un bon exercice consisterait à chercher une meilleure tactique de jeu et à la programmer. Si vous n'arrivez plus à le battre, c'est certainement que cette tactique est bonne.

```

10 : REM Jeu des allumettes
20 :
30 : GOSUB 1000 'Initialisation
40 : GOSUB 2000 Jeu
50 : GOSUB 3000 'Fin
60 :
70 : END
80 : REM *****
1000 : REM Initialisation du jeu
1010 :
1020 : GOSUB 1100 'Regle du jeu
1030 : GOSUB 1600 'Initialisation
1040 :
1050 : RETURN

```

```

1060 : REM #####
1100 : REM Regle du jeu
1110 :
1120 : PAPER 0:INK 5
1130 : REM titre
1140 : PRINT CHR$(12);CHR$(4)
1150 : PRINT CHR$(27);"N";CHR$(27);"A"      JEU DES ALLUMETTE
S";
1160 : PRINT CHR$(4)
1170 : REM Objet
1180 : PRINT:PRINT:PRINT:PRINT
1190 : A$="D":PRINT"OBJET :";
1200 : GOSUB 1500:PRINT"Jeu a astuces."
1210 : REM Job
1220 : PRINT:PRINT:PRINT:PRINT
1230 : A$="F":PRINT"JOB  :";
1240 : GOSUB 1500:PRINT"N allumettes sont disposees"
1250 : GOSUB 1500:PRINT"      sur l'ecran devant 2 joueurs:"
1260 : GOSUB 1500:PRINT"      vous et moi . Chacun des deux"
1270 : GOSUB 1500:PRINT"      joueurs va retirer a tour de "
1280 : GOSUB 1500:PRINT"      role 1,2 ou 3 allumettes ."
1290 : GOSUB 1500:PRINT"      Celui qui preleve la ou les"
1300 : GOSUB 1500:PRINT"      derniere(s) allumette(s)"
1400 : GOSUB 1500:PRINT"      a gagne ."
1410 : REM Debut
1420 : PRINT:PRINT:PRINT
1430 : A$="C":PRINT"DEBUT :";
1440 : GOSUB 1500:PRINT"Pour commencer le jeu,appuyez";
1450 : GOSUB 1500:PRINT"      sur une touche quelconque ."
1460 : GET B$
1470 : RETURN
1480 : REM #####
1500 : REM Couleur d'une ligne
1510 :
1520 :PRINT CHR$(27)"H";CHR$(27);A$;
1530 : RETURN
1540 : REM #####
1600 : REM Initialisation du jeu
1610 :
1620 : CLS 'Effacement ecran
1630 : PRINT:PRINT:PRINT:PRINT
1640 : INPUT"Nombre d'allumettes";N
1650 : FOR I=1 TO 7
1660 : READ T(I)
1670 : NEXT I
1680 :DATA 1,2,3,1,1,2,3
1690 : RETURN
1700 : REM #####

```

```
2000 : REM Jeu
2010 :
2020 : PRINT
2030 : INPUT "Combien en prenez-vous";USER
2040 : IF USER<1 THEN GOTO 2150
2050 : IF USER>3 THEN GOTO 2150
2060 : N=N-USER
2070 : IF N=0 THEN GAGN=1:RETURN
2080 : PRINT "Il en reste";N
2090 : IF N>7 THEN TIR=INT(RND(1)*3)+1 ELSE TIR=T(N)
2100 : PRINT "J'en prends";TIR
2110 : N=N-TIR
2120 : PRINT "Il en reste";N
2130 : IF N=0 THEN GAGN=0:RETURN
2140 : GOTO 2000 'Le jeu continue
2150 : PRINT "Prenez 1,2 ou 3 allumettes !!":GOTO 2000
2160 : REM *****
3000 : REM Fin de la partie
3010 :
3020 : IF GAGN=0 THEN PRINT "J'ai gagne." ELSE PRINT "Vous avez gagn
e."
3030 : RETURN
```

Les piranhas attaquent

PRESENTATION

Un baigneur (en l'occurrence vous) doit échapper à une attaque massive de piranhas. Il peut se déplacer d'une case dans n'importe qu'elle direction. Après quoi, chacun des piranhas (eh oui ! la vie est dure) se déplace également d'une case dans **votre** direction. Le seul moyen pour échapper à ces prédateurs est de se cacher derrière un rocher. En effet, un piranha ne sait pas distinguer un nageur d'un rocher, et, s'il a le malheur de croquer un rocher, sa vie s'arrête là, c'est l'indigestion.

ORGANIGRAMME DU JEU

Le jeu, d'apparence complexe se décompose en fait de manière fort simple en deux opérations :

- . Déplacement joueur,
- . Déplacement piranhas

Déplacement joueur :

Les touches E R T } contrôlent le déplacement
 D F G } dans tous les sens de
 C V B } la manière suivante

- Si le joueur appuie sur une autre touche que E, R, T, D, F, G, C V ou B, le déplacement n'est pas accepté.
- Si le joueur tente de percuter la côte (c'est-à-dire les extrémités de l'écran), il ne bouge pas, mais les piranhas avancent tous d'une case.
- Enfin, si le déplacement fait atterir le nageur sur un rocher, ou... sur un piranha, la partie s'arrête.

BOITE A OUTILS POUR ORIC 1 ET ATNOS / 61

FIN	Indicateur de fin de partie
I	Index de boucle.
NB	Nombre de piranhas.
X, Y	Tableaux de déplacements des piranhas
XN, YN	Position du nageur
XP, YP	Position des piranhas
XR, YR	Position des rochers

```
10 : REM Les Pirhanas attaquent
20 :
30 : GOSUB 1000 'Module d'initialisation
40 : GOSUB 2000 'Jeu
50 : GOSUB 3000 'Fin du jeu
60 : END
70 :
80 : REM *****
1000 : REM Module d'affichage du menu
1010 :
1020 : GOSUB 1100 'Definition du jeu
1030 : GOSUB 1800 'Debut du jeu
1040 :
1050 : RETURN
```

```

1060 : REM *****
1100 : REM Regles du jeu
1110 :
1120 : PAPER 0:INK 5:PRINT CHR$(17)
1130 : REM Titre
1140 : PRINT CHR$(12)
1150 : PRINT CHR$(4);CHR$(27);"N";
1160 : PRINT CHR$(27);"A LES PIRHANAS ATTAQUENT";CHR$(4)
1180 : REM Objet
1190 : PRINT:PRINT:PRINT
1200 : A$="D" 'Code couleur=Bleu
1210 : PRINT"OBJET :";
1220 : GOSUB 1500:PRINT"Jeu de reflexion ."
1230 : REM Job
1240 : PRINT:PRINT:PRINT
1245 : GOSUB 1700 'Confection des caracteres particuliers
1250 : A$="F" 'Code couleur=Cyan
1260 : PRINT"JOB :";
1270 : GOSUB 1500:PRINT"Vous etes dans une mer pleine";
1280 : GOSUB 1500:PRINT" de pirhanas.Les $ representent"
1290 : GOSUB 1500:PRINT" des rochers,les + des pirhanas"
1300 : GOSUB 1500:PRINT" et 0 le joueur."
1310 : GOSUB 1500:PRINT" Les touches ERT DFG et CVB"
1320 : GOSUB 1500:PRINT" permettent de se deplacer dans"
1330 : GOSUB 1500:PRINT" toutes les directions selon le"
1340 : GOSUB 1500:PRINT" schema ci-dessous : " : PRINT
1350 : GOSUB 1500:PRINT" Touches ERT Deplacement XYZ"
1360 : GOSUB 1500:PRINT" '' DFG '' [\]"
1370 : GOSUB 1500:PRINT" '' CVB '' ^_";CHR$(
(96)
1380 : REM Debut
1390 : PRINT:PRINT
1400 : A$="C" 'Code couleur = Jaune
1410 : PRINT"DEBUT :";
1420 : GOSUB 1500:PRINT"Pour commencer,appuyez sur"
1430 : GOSUB 1500:PRINT" une touche quelconque ."
1440 : GET A$ 'Attente d'une action clavier
1450 : RETURN
1460 : REM *****
1500 : REM Module de couleur de ligne
1510 :
1520 : PRINT CHR$(27);"H";CHR$(27);A$;
1530 : RETURN
1540 : REM *****
1700 : REM Redefinition des lettres A a J
1710 :
1720 : FOR I=0 TO 8*9-1
1730 : READ A : POKE 46080+8*88+I,A
1740 : NEXT I

```



```

1745 : RESTORE 'Position pointeur
1750 :DATA 0,0,60,48,40,36,2,0 'A
1751 :DATA 4,14,21,4,4,4,4,0 'B
1752 :DATA 0,0,31,3,5,9,16,0 'C
1753 :DATA 0,8,16,63,16,8,0,0 'D
1754 :DATA 0,0,0,4,0,0,0,0 'E
1755 :DATA 0,4,2,63,2,4,0,0 'F
1756 :DATA 1,2,36,40,48,62,0,0 'G
1757 :DATA 0,4,4,4,4,21,14,4 'H
1758 :DATA 32,16,9,5,3,15,0,0 'I
1759 : RETURN
1760 : REM #####
1800 : REM Initialisation du jeu
1810 :
1820 : CLS 'Effacement d'ecran
1830 : DIM X(10),Y(10) 'Dimension des tableaux de travail
1840 : FOR I=1 TO 08 'Nombre de pirhannas au debut
1850 :   XP=INT(RND(1)*38)+1 : YP=INT(RND(1)*24)+1
1860 :   A=SCRN(XP,YP)
1870 :   IF A<>32 THEN I=I-1 ELSE PLOT XP,YP,"+":X(I)=XP:Y(I)=YP
1880 : NEXT I
1890 : FOR I=1 TO 30 'Nombre de rochers
1900 :   XR=INT(RND(1)*38)+1 : YR=INT(RND(1)*24)+1
1910 :   A=SCRN(XR,YR)
1920 :   IF A<>32 THEN I=I-1 ELSE PLOT XR,YR,"*"
1930 : NEXT I
1940 : XN=INT(RND(1)*38)+1 : YN=INT(RND(1)*24)+1
1950 : IF SCRN(XN,YN)<>32 THEN GOTO 1940
1960 : PLOT XN,YN,"0" : RETURN
1970 : REM #####
2000 : REM Deroulement du jeu
2010 :
2020 : GOSUB 2200 'Joueur
2030 : GOSUB 2500 'Pirhannas
2040 : IF FIN<>1 THEN GOTO 2000
2050 : RETURN
2060 : REM #####
2200 : REM Deroulement du jeu
2210 :
2220 : GET A$
2230 : A=ASC(A$)
2240 : IF A=69ORA=82ORA=84ORA=68ORA=70ORA=71ORA=67ORA=86ORA=66 THEN
2260
2250 : GOTO 2200 'Deplacement non accepte
2260 : PLOT XN,YN," ":SX=XN:SY=YN

```

```

2270 : IF A$="E" THEN XN=XN-1:YN=YN-1:GOTO 2360
2280 : IF A$="R" THEN YN=YN-1:GOTO 2360
2290 : IF A$="T" THEN XN=XN+1:YN=YN-1:GOTO 2360
2300 : IF A$="D" THEN XN=XN-1:GOTO 2360
2310 : IF A$="F" THEN GOTO 2360
2320 : IF A$="G" THEN XN=XN+1:GOTO 2360
2330 : IF A$="C" THEN XN=XN-1:YN=YN+1:GOTO 2360
2340 : IF A$="V" THEN YN=YN+1:GOTO 2360
2350 : IF A$="B" THEN XN=XN+1:YN=YN+1
2360 : IF XN=39ORXN<1ORYN=27ORYN<0THENXN=SX:YN=SY: PLOTXN,YN,"O":R
RETURN
2365 : A=SCRN(XN,YN)
2370 : IF A<>32 THEN GOSUB 2700
2380 : PLOT XN,YN,"O"
2390 : RETURN
2400 : REM #####
2500 : REM Deplacement Pirhana
2510 :
2520 : FOR I=1 TO 08
2525 : IF X(I)=99 THEN NEXT I:IF I=9 THEN RETURN
2530 : PLOT X(I),Y(I)," "
2540 : IF X(I)+Y(I)=0 THEN NEXT I
2550 : DX=XN-X(I)
2560 : IF DX>=1 THEN X(I)=X(I)+1
2570 : IF DX<=-1 THEN X(I)=X(I)-1
2580 : DY=YN-Y(I)
2590 : IF DY>=1 THEN Y(I)=Y(I)+1
2600 : IF DY<=-1 THEN Y(I)=Y(I)-1
2610 : IF SCRN(X(I),Y(I))=42 THEN GOSUB 2660:NEXT I:IF I=9THENRE
TURN
2620 : IF SCRN(X(I),Y(I))=48 THEN GOSUB 2700:NEXT I:IF I=9THENRE
TURN
2630 : PLOT X(I),Y(I),"+"
2640 : NEXT I
2650 : RETURN
2660 : SHOOT
2670 : PLOT X(I),Y(I)," "
2680 : X(I)=99
2690 : RETURN
2700 : EXPLODE 'Nageur foutu
2710 : FIN=1 'Fin de la partie
2720 : RETURN
2800 RETURN
3000 : REM Fin de partie
3010 : FOR I=1 TO 5
3020 : PAPER1 : WAIT 10 : PAPER0 : WAIT 10
3030 : NEXT I
3040 : CLS:PRINT CHR$(17)
3050 : RETURN

```

Mémoires à court terme

PRESENTATION

Tout le monde possède une mémoire à court terme. En effet, tout le monde peut emmagasiner des concepts de manière immédiate.

Ce programme se propose de déterminer la limite supérieure de votre mémoire à court terme.

Pour ce faire, ORIC envoie N caractères à l'écran pendant une seconde. Vous devez restituer les N caractères affichés, dans le bon ordre. Une bonne réponse, et le nombre de caractères augmente de un, une mauvaise réponse, et le nombre de caractères diminue de un.

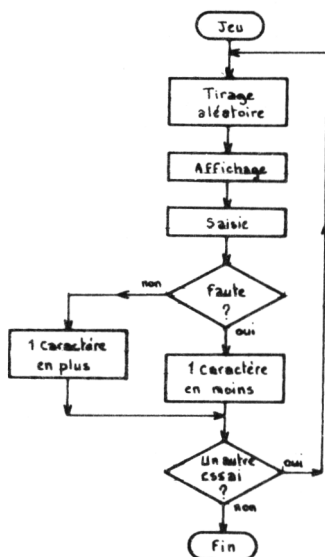
Ce jeu passionnant, devient vite difficile, voire infaisable.

ORGANIGRAMME DU JEU

Le programme a la structure A.

Le jeu a la structure suivante :

(voir page suivante)



LE PROGRAMME

Il se décompose en trois pavés :

Initialisation

Lignes 1000 à 1550
C'est la présentation du programme à l'utilisateur

Jeu

Lignes 2000 à 2260
Le jeu se décompose en :

- Tirage aléatoire Lignes 2200 à 2250
- Affichage Lignes 2050 à 2060
- Saisie réponse Ligne 2100
- Test validité Lignes 2110 à 2170

Fin du jeu

Lignes 3000 à 3070
Restitution du curseur et arrêt du programme

LES VARIABLES UTILISEES

A

Code ASCII d'un tirage aléatoire

A\$	Code couleur de la ligne
B	Un des caractères de la réponse
DI	Nombre de caractères à afficher
R\$	Variable de fin de partie
T	Tableau du jeu

```

10 : REM Memoire a court terme
20 :
30 :   GOSUB 1000 'Initialisation
40 :   GOSUB 2000 'Jeu
50 :   GOSUB 3000 'Fin du jeu
60 :   IF BIS=1 THEN GOTO 40 'Nouvelle partie
70 :
80 :   PRINT CHR$(17):END
90 :   REM #####
1000 : REM Initialisation
1010 :
1020 :   PAPER 0 : INK 5 : PRINT CHR$(17) : DIM T(100):GA=1
1030 :   REM Titre
1040 :
1050 :   PRINT CHR$(12)
1060 :   PRINT CHR$(4);CHR$(27);"N";
1070 :   PRINT CHR$(27);"A      MEMOIRE A COURT TERME";CHR$(4)
1080 :
1090 :   REM Objet
1100 :
1110 :   PRINT:PRINT:PRINT:PRINT:PRINT
1120 :   A$="D" 'Code couleur = Bleu
1130 :   PRINT"OBJET : ";
1140 :   GOSUB 1400:PRINT"Jeu de memoire."
1150 :
1160 :   REM Job
1170 :
1180 :   PRINT:PRINT:PRINT:PRINT
1190 :   A$="F" 'Code couleur = Cyan
1200 :   PRINT"JOB   : ";
1210 :   GOSUB 1400:PRINT"ORIC va afficher une suite"
1220 :   GOSUB 1400:PRINT"      de lettres pendant 3 secondes."
1230 :   GOSUB 1400:PRINT"      Vous devrez la restituer juste."
;
1240 :   GOSUB 1400:PRINT"      Une bonne reponse augmente de 1"
;
1250 :   GOSUB 1400:PRINT"      le nombre de lettres et une"
1260 :   GOSUB 1400:PRINT"      mauvaise reponse le diminue d'1"
;
1270 :

```

```

1280 : REM Debut
1290 :
1300 : PRINT:PRINT:PRINT:PRINT
1310 : A$="C" 'Code couleur = Jaune
1320 : PRINT"DEBUT : ";
1330 : GOSUB 1400:PRINT"Pour commencer , appuyez sur";
1340 : GOSUB 1400:PRINT"      une touche quelconque."
1350 :
1360 : GET A$ 'Attente de l'appui sur une touche
1370 :
1380 : RETURN
1390 : REM *****
1400 : REM Couleur d'une ligne
1410 :
1420 : PRINT CHR$(27);"H";CHR$(27);A$;
1430 :
1440 : RETURN
1450 : REM *****
2000 : REM Deroulement du jeu
2010 :
2020 : IF GA=1 THEN DI=DI+1 ELSE DI=DI-1
2025 : CLS:PRINT:PRINT:PRINT:PRINT
2030 : FOR I=1 TO DI
2040 : GOSUB 2300 'Tir aleatoire d'une lettre
2050 : T(I)=A
2060 : PRINT CHR$(A);" ";
2070 : NEXT I
2080 :
2090 : WAIT 300 '3secondes de pause
2100 :
2110 : CLS
2120 : FOR I=1 TO 8
2130 : PRINT
2140 : NEXT I
2150 : A$="B":GOSUB 1400:PRINT"Appuyez la suite de ";DI;" lettre(s)
)"
2155 : GOSUB 1400:PRINT"appuyez sur l'ecran."
2160 : PRINT:INPUT A$
2170 :
2175 : IF DI<>LEN(A$) THEN EN=1:GA=1:RETURN
2180 : FOR I=1 TO DI
2190 : B$=MID$(A$,I,1)
2200 : IF ASC(B$)<>T(I) THEN ERR=ERR+1 'Erreur
2210 : NEXT I
2220 :
2230 : IF ERR=0 THEN GA=1 ELSE GA=0
2240 :
2250 : RETURN

```

```
2260 : REM *****
2300 : REM Tir aleatoire d'une lettre
2310 :
2320 : A=INT(RND(1)*26)+65
2330 :
2340 : RETURN
2350 : REM *****
3000 : REM Fin de la partie
3010 :
3020 : CLS
3030 : IF GA=1 AND EN=0 THEN PRINT"Bien joue !"
3040 : IF GA=0 THEN PRINT"Pas bien brillant : ";ERR;" erreur(s)!!
":ERR=0
3045 : IF EN=1 THEN EN=0:PRINT"J'avais dit ";DI;" lettres !!!":DI
=DI-1
3050 :
3060 : PRINT:PRINT:PRINT
3070 : INPUT"Un autre essai (O/N) ";R$
3080 : IF R$="O" THEN BIS=1 ELSE BIS=0
3090 : PRINT:PRINT"Fin du jeu au score de ";DI;" lettres."
3100 :
3110 : RETURN
3120 : REM *****
```

Mastermind

PRESENTATION

La réputation de ce jeu n'est plus à faire. Pour les jeunes poussins, rappelons tout de même la règle du jeu :

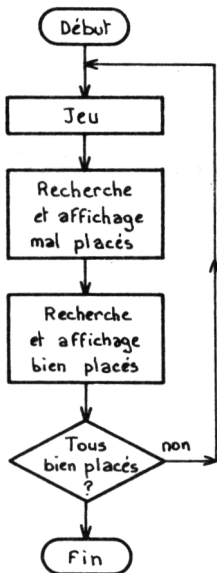
Un nombre à N chiffres (ici 4 chiffres) doit être découvert en un minimum de coups. ORIC choisit ce nombre et indique le nombre de chiffres bien placés et le nombre de chiffres mal placés.

Intellectuel non ?

ORGANIGRAMME DU JEU

Le programme a la structure A.

Le jeu, très simple, consiste à afficher le nombre de chiffres bien (et mal) placés et à arrêter la partie quand les quatre chiffres sont bien placés. Ce qui se représente de la manière suivante :



LE PROGRAMME

Il se décompose en trois pavés :

Présentation et initialisation

Lignes 1000 à 1710
 Module de commande Lignes 1000 à 1060
 Présentation Lignes 1100 à 1550
 Init. variables Lignes 1600 à 1710

Jeu

Lignes 2000 à 2210
 Recherche et affichage des chiffres bien et mal placés.

Fin du jeu

Lignes 3000 à 3060
 Affichage du nombre de coups nécessaires pour trouver les quatre chiffres et leur position.

LES VARIABLES UTILISEES

A\$	Code couleur de ligne
A	Indicateur de double dans la recherche du nombre à trouver
BP	Indicateur de chiffre bien placé
I, J	Index de boucle
MP	Indicateur de chiffre mal placé
NB	Nombre de coups joués
R	Tableau joueur : contient le nombre proposé
T	Tableau ORIC : contient le nombre à trouver.

```

10 : REM MASTERMIND
20 :
30 : GOSUB 1000 'Initialisation de la partie
40 : GOSUB 2000 'Jeu
50 : GOSUB 3000 'Fin de la partie
60 :
70 : END
80 : REM *****
1000 : REM Initialisation
1010 :
1020 : GOSUB 1100 'Affichage menu
1030 : GOSUB 1600 'Initialisation des variables
1040 :
1050 : RETURN
1060 : REM *****
1100 : REM Affichage du menu
1110 :
1120 : PAPER 0 : INK 6 : PRINT CHR$(17) 'Suppression curseur
1130 : REM Titre
1140 :
1150 : PRINT CHR$(12)
1160 : PRINT CHR$(4);CHR$(27);"N";
1170 : PRINT CHR$(27);"A" MASTERMIND";CHR$(4)
1180 : REM Objet
1190 :
1200 : PRINT:PRINT:PRINT:PRINT
1210 : A$="D" 'Code couleur = Bleu
1220 : PRINT"OBJET : ";
1230 : GOSUB 1500:PRINT"Jeu de reflexion."
1240 :
1250 : REM Job

```

```

1260 :
1270 : PRINT:PRINT:PRINT
1280 : A$="F" 'Code couleur = Cyan
1290 : PRINT"JOB  :";
1300 : GOSUB 1500:PRINT"ORIC choisit un nombre"
1310 : GOSUB 1500:PRINT"      a 4 chiffres tous differents."
1320 : GOSUB 1500:PRINT"      Le but du jeu est de trouver ce"
;
1330 : GOSUB 1500:PRINT"      nombre en un minimum de coups."
1340 :
1350 : REM Debut
1360 :
1370 : PRINT:PRINT:PRINT
1380 : A$="C" 'Code couleur = Jaune
1390 : PRINT"DEBUT  :";
1400 : GOSUB 1500:PRINT"Pour commencer,appuyez sur"
1410 : GOSUB 1500:PRINT"      une touche quelconque."
1420 : GET A$ 'Attente d'une action clavier
1430 :
1440 : RETURN
1450 : REM #####
1500 : REM Definition de la couleur d'une ligne
1510 :
1520 : PRINT CHR$(27);"H";CHR$(27);A$;
1530 :
1540 : RETURN
1550 : REM #####
1600 : REM Initialisation des variables
1610 :
1620 : FOR I=1 TO 4
1630 :   T(I)=INT(RND(1)*10)
1635 :   IF I=1 THEN NEXT I
1640 :   FOR J=1 TO I-1
1650 :     IF T(J)=T(I) THEN A=1
1660 :   NEXT J
1670 :   IF A=1 THEN I=I-1:A=0
1680 : NEXT I
1690 :
1700 : RETURN

```

```
1710 : REM *****
2000 : REM Jeu
2010 :
2020 : CLS
2030 : PLOT 20,1,"BP":PLOT 30,1,"MP":PLOT 3,1,"Jeu":PRINT:PRINT:P
RINT
2040 : BP=0:MP=0:NB=NB+1 'Nombre de coups joues
2050 : INPUT A$
2060 : FOR I=1 TO 4
2070 :   R(I)=ASC(MID$(A$,I,1))-48
2080 : NEXT I
2090 : FOR I=1 TO 4
2100 :   IF T(I)=R(I) THEN BP=BP+1 ELSE GOSUB 2170
2140 : NEXT I
2150 : PLOT 20,NB+2,BP+48:PLOT 30,NB+2,MP+48
2160 : IF BP=4 THEN RETURN ELSE GOTO 2040
2170 :   FOR J=1 TO 4
2180 :     IF R(I)=T(J) THEN MP=MP+1
2190 :   NEXT J
2200 : RETURN
3000 : REM Fin du jeu
3010 :
3020 : CLS
3030 : PRINT "Le nombre a ete trouve en ";NB;"coups."
3040 : PRINT CHR$(17) 'Restitution du curseur
3050 :
3060 : RETURN
```

GRAPHIQUES ET MATHÉMATIQUES

Figures de Moivre

PRESENTATION

On attribue souvent le nom de "**figures de Moivre**" aux dessins obtenus par le glissement d'une droite sur deux courbes planes. Les adeptes de "fils et pointes" ne me contrediront pas...

Les ordinateurs graphiques en général, et ORIC en particulier, permettent de tracer de telles figures à moindre effort.

Dans la suite, sept figures sont proposées :

ORGANIGRAMME DU TRACE

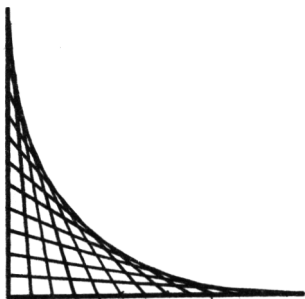
Chaque programme a la structure linéaire B.

Description de chaque programme :

- **MOIVRE 1** : Les deux courbes planes sont deux droites à 90°
- **MOIVRE 2** : Les deux courbes planes sont deux droites d'angle $< 90^\circ$
- **MOIVRE 3** : Combinaison de 4 figures de type 1.
- **MOIVRE 4** : Les deux courbes planes sont deux droites à 90°
- **MOIVRE 5** : Combinaison de 4 figures de type 4
- **MOIVRE 6** : Les deux courbes planes sont deux côtés d'un triangle isocèle.
- **MOIVRE 7** : Combinaison de 2 figures de type 6 pour former un triangle complet.

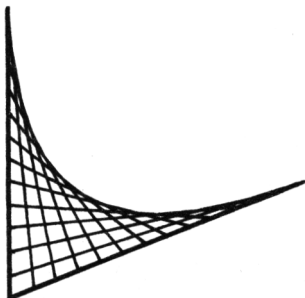
```

10 : REM Moivre 1
20 :
30 : HIRES
40 : FOR I=0 TO 19
50 :   CURSET 15,10*I,1
  0 :   DRAW 10*(I+1),199-(10*I),1
70 : NEXT I
80 : GET A$
90 : LORES 0
100 :
110 : END
    
```



```

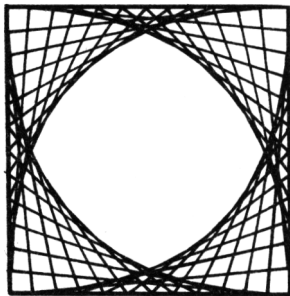
10 : REM Moivre 2
20 :
30 : HIRES
40 : FOR I=0 TO 19
50 :   CURSET 15,10*I,1
  60 :   DRAW 10*(I+1),199-(19*I),1
70 : NEXT I
80 : GET A$
90 : LORES 0
100 :
110 : END
    
```



```

10 : REM Moivre 3
20 :
30 : HIRES
40 : FOR I=1 TO 12
50 :   CURSET 20,16*I,1
60 :   DRAW 16*I,199-16*I,1
70 : NEXT I
80 : FOR I=0 TO 11
90 :   CURSET 20+I*16,0,1
100 :  DRAW 200-16*I,16*I,1
110 : NEXT I
120 : FOR I=0 TO 11
130 :   CURSET 20,199-16*I,1
140 :   DRAW 16*I,-199+16*I,1
150 : NEXT I
160 : FOR I=0 TO 11
170 :   CURSET 20+I*16,199,1
180 :   DRAW 200-16*I,-16*I,1
190 : NEXT I
200 : CURSET 220,0,1:DRAW 0,199,1
210 :
220 : GET A$ : LORES 0
230 :
240 : END

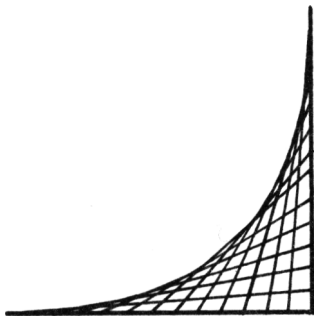
```



```

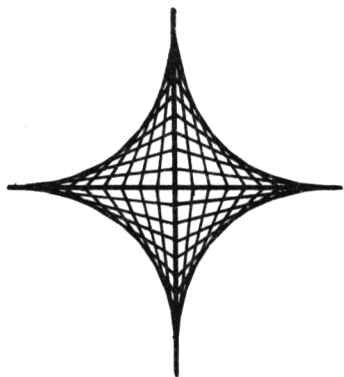
10 : REM Moivre 4
20 :
30 : HIRES
40 : FOR I=1 TO 10
50 :   CURSET 20+I*9,99,1
60 :   DRAW 100-I*9,-I*9,1
70 : NEXT I
80 :
90 : GET A$ : LORES 0
100 :
110 : END

```



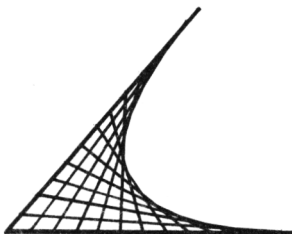

```

10 : REM Moivre 5
20 :
30 : HIRES
40 : FOR I=1 TO 11
50 :   CURSET 20+I*9,99,1
60 :   DRAW 100-9*I,9*I,1
70 :   DRAW 100-9*I,-9*I,1
80 :   CURSET 20+I*9,99,1
90 :   DRAW 100-9*I,-9*I,1
100 :  DRAW 100-9*I,9*I,1
110 : NEXT
120 :
130 : CURSET 120,0,0:DRAW 0,199,1
140 : CURSET 20,99,0:DRAW 200,0,1
150 :
160 : ET A$ : LORES 0
170 :
180 END
    
```

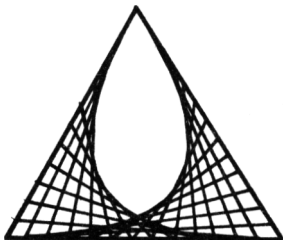


```

5 HIRES
10 : REM Moivre 6
20 :
30 : HIRES
40 : FOR I=1 TO 10
50 :   CURSET 70+I*10,150,1
60 :   DRAW 55-I*15,-120+10*I,1
70 : NEXT I
80 :
90 : GET A$ : LORES 0
100 :
110 : END
    
```



```
5 HIRES
10 : REM Moivre 7
20 :
30 : HIRES
40 : FOR I=1 TO 10
50 :   CURSET 70+I#10,150,1
60 :   DRAW 55-I#15,-120+10#I,1
70 :   CURSET 170-I#10,150,1
80 :   DRAW -55+I#15,-120+I#10,1
90 : NEXT I
100 :
110 : CURSET 65,150,1
120 : DRAW 110,0,1
130 : DRAW -55,-110,1
140 : DRAW -55,110,1
150 :
160 : GET A# : LORES 0
170 :
180 : END
```



Biorythmes

PRESENTATION

Des études de statistiques ont permis de modéliser la "forme" relative d'un individu tout au long de sa vie. Cette modélisation met en jeu trois cycles temporels sinusoidaux :

- . Un cycle physique de 23 jours,
- . Un cycle émotionnel de 28 jours,
- . Un cycle intellectuel de 31 jours.

Le nom de biorythme désigne la représentation mensuelle de ces 3 cycles. Les sinusoides ont pour origine votre date de naissance. (Elles sont donc particulières à chaque individu).

L'ordonnée Y de chaque sinusoides indique :

- . Une période active si $Y > 0$,
- . Une période passive si $Y < 0$, (de récupération)
- . Une période critique si $Y = 0$ (restez bien tranquillement chez vous ces jours là).

ORGANIGRAMME DU TRACE

Le programme a la structure A.

Le tracé est initialisé par l'entrée de la date de naissance de l'intéressé, et par les mois et année du tracé.

Puis, le nombre de jours écoulés depuis la naissance jusqu'au premier jour du mois du tracé est calculé.

Le tracé consiste simplement en :

- calcul des ordonnées de chaque courbe pour tous les jours du mois du tracé.
- affichage de ces ordonnées.

LE PROGRAMME

Il se décompose en trois pavés :

Présentation et initialisation

Lignes 1000 à 1710	
Module de commande	Lignes 1000 à 1060
Présentation	Lignes 1100 à 1550
Initialisation	Lignes 1600 à 1710

Acquisition des données

Lignes 2000 à 2180
Date de naissance,
Mois et année du
tracé

Tracé

Lignes 3000 à 3710	
Calcul du nombre de jours écoulés depuis la naissance	Lignes 3000 à 3260
Affichage	Lignes 3300 à 3710

LES VARIABLES UTILISEES

A\$	Code couleur de ligne
A, B, C	Ordonnées de chaque cycle
AB	Année du biorythme
BI	Indicateur d'année bissextile
I	Index de boucle
IN	Variable de calcul
JN	Jour de naissance
MN	Mois de naissance
T	Tableau du nombre de jours de chaque mois
TT	Nombre de jours écoulés depuis la naissance.

```

10 : REM Biorythmes
20 :
30 :   GOSUB 1000 'Initialisation
40 :   GOSUB 2000 'Acquisition des donnees
50 :   GOSUB 3000 'Trace des biorythmes
60 :
70 : END
80 : REM *****
1000 : REM Affichage du menu et initialisation des variables
1010 :
1020 :   GOSUB 1100 'Presentation du menu
1030 :   GOSUB 1600 'Initialisation des variables
1040 : RETURN
1050 :
1060 : REM *****
1100 : REM Presentation du menu
1110 :
1120 : PAPER 0:INK 5:PRINT CHR$(17)
1130 : REM Titre
1140 :   PRINT CHR$(12)
1150 :   PRINT CHR$(4);CHR$(27);"N";
1160 :   PRINT CHR$(27);"A           BIORYTHMES";CHR$(4)
1170 : REM Objet
1180 :   PRINT:PRINT:PRINT:PRINT
1190 :   A$="D" 'Code couleur = Bleu
1200 :   PRINT"OBJET :";
1210 :   GOSUB 1500:PRINT"Trace de biorythmes."
1220 : REM Job
1230 :   PRINT:PRINT:PRINT
1240 :   A$="F" 'Code couleur = Cyan
1250 :   PRINT"JOB  :";
1260 :   GOSUB 1500:PRINT"Apres l'entree de votre date"
1270 :   GOSUB 1500:PRINT"      de naissance , des mois et"
1280 :   GOSUB 1500:PRINT"      annee pour lesquels vous "
1290 :   GOSUB 1500:PRINT"      desirez connaitre les courbes"
1300 :   GOSUB 1500:PRINT"      biorythmes , ORIC se fera un"
1310 :   GOSUB 1500:PRINT"      plaisir de tracer les cycles"
1320 :   GOSUB 1500:PRINT"      intellectuel , physique et"
1330 :   GOSUB 1500:PRINT"      affectif du mois demande ."
1340 : REM Debut
1350 :   PRINT:PRINT:PRINT
1360 :   A$="C" 'Code couleur = Jaune
1370 :   PRINT"DEBUT :";
1380 :   GOSUB 1500:PRINT"Pour commencer , appuyez sur"
1390 :   GOSUB 1500:PRINT"      une touche quelconque."
1400 :   GET A$ 'Attente de la frappe d'une touche du clavier
1410 :
1420 : RETURN
1430 :

```

```
1440 : REM #####
1500 : REM Couleur d'une ligne
1510 :
1520 : PRINT CHR$(27);"H";CHR$(27);A$;
1530 : RETURN
1540 :
1550 : REM #####
1600 : REM Initialisation des variables du programme
1610 :
1620 : REM Duree en jours de chaque mois
1630 :
1640 : DIM T(12) 'Tableau des mois
1650 : FOR I=1 TO 12
1660 : READ T(I)
1670 : NEXT I
1680 DATA 31,28,31,30,31,30,31,31,30,31,30,31
1690 :
1700 : RETURN
1710 : REM #####
2000 : REM Acquisition des donnees
2010 :
2020 : CLS:PRINT CHR$(17)
2030 : PRINT CHR$(27);"L";CHR$(27);"A Saisie des donnees"
2040 : PRINT:PRINT:PRINT:PRINT:PRINT
2050 : A$="D":GOSUB 1500:PRINT"Date de naissance : "
2060 : PRINT:PRINT
2070 : INPUT"Jour de naissance";JN
2080 : INPUT"Mois de naissance";MN
2090 : INPUT"Annee de naissance";AN
2100 :
2110 : PRINT:PRINT:PRINT:PRINT
2120 : GOSUB 1500:PRINT"Biorythmes : "
2130 : PRINT:PRINT
2140 : INPUT"Mois du trace";MB
2150 : INPUT"Annee du biorythme";AB
2160 :
2170 : RETURN
2180 : REM #####
```

```

3000 : REM Trace des biorythmes
3010 :
3020 : REM Calcul du nombre de jours ecoules depuis la naissance
3030 :
3040 : REM Pendant l'annee de naissance
3050 : IF (AN/4)=INT(AN/4) THEN IN=1 ELSE IN=0
3055 : IF IN=1 AND MN=2 THEN FIN=29 ELSE FIN=T(MN)
3060 : FOR I=MN TO FIN
3070 : TT=TT+1
3080 : NEXT I
3090 : FOR I=MN+1 TO 12
3100 : IF IN=1 AND I=2 THEN TT=TT+29 ELSE TT=TT+T(I)
3110 : NEXT I
3120 :
3130 : REM Pendant les annees intermediaires
3140 : FOR I=2 TO AB-AN
3150 : B=AN+I-1 'Annee consideree
3160 : IF B/4=INT(B/4) THEN TT=TT+1
3170 : TT=TT+365
3180 : NEXT I
3190 :
3200 : REM Pendant l'annee du biorythme
3210 : IF AB/4 = INT(AB/4) THEN BI=1 ELSE BI=0
3220 : FOR I=1 TO MB-1
3230 : IF BI=1 AND I=2 THEN TT=TT+29 ELSE TT=TT+T(I)
3240 : NEXT I
3250 :
3260 : REM *****
3300 : REM Affichage
3310 :
3320 : CLS : INK 6 : PRINT CHR$(17)
3330 : PRINT CHR$(27);"L";CHR$(27);"A" BIORYTHMES"
3340 :
3350 : FOR I=1 TO T(MB)
3360 : A=SIN(2*PI*(TT+I)/23)
3370 : PLOT I+3,13-10*A,"#"
3380 : B=SIN(2*PI*(TT+I)/28)
3390 : PLOT I+3,13-10*B,"."
3400 : C=SIN(2*PI*(TT+I)/33)
3410 : PLOT I+3,13-10*C,"+"
3420 : NEXT I
3430 :
3440 : REM Affichage des jours
3450 :
3460 : PLOT 4,23,"1":PLOT 9,23,"5":PLOT 14,23,"10":PLOT 19,23,"
15"
3470 : PLOT 24,23,"20":PLOT 29,23,"25":PLOT 34,23,"30"

```

```
3480 :  
3490 : REM Axes  
3500 :  
3510 : FOR I=1 TO 31  
3520 : PLOT 3+I,13,"-"  
3530 : NEXT I  
3540 :  
3550 : FOR I=1 TO 20  
3560 : PLOT 3,I+1,"!"  
3570 : NEXT I  
3580 :  
3590 : PLOT 35,13,">":PLOT 3,2,"^"  
3600 :  
3610 : REM Nature des courbes  
3620 :  
3630 : PLOT 0,25," # = Physique . = Affectif"  
3640 : PLOT 0,26," + = Intellectuel"  
3650 :  
3660 : REM Attente de l'appui sur une touche pour finir  
3670 :  
3680 : GET A$ 'Attente  
3690 : CLS : PRINT CHR$(17) 'Restitution du curseur  
3700 :  
3710 : RETURN
```


Jeu de la vie

PRESENTATION

L'évolution temporelle d'un être vivant est dictée par des lois bien précises qui tendent :

- . à maintenir l'espèce vivante, ou ...
- . à anéantir l'espèce.

L'espèce est maintenue vivante si elle arrive à respecter ces lois, et prolifère d'autant plus qu'elle n'en éprouve pas de difficultés.

En revanche, l'espèce disparaît si elle est incompatible avec ces lois. Cette disparition est plus ou moins rapide, selon la faculté d'adaptation de l'espèce.

Ce programme s'intéresse à la vie de cellules animales. Son évolution est dictée par trois lois :

- **Phase d'adaptation :**
Toute cellule entourée par 2 ou 3 autres cellules survit.
- **Phase de reproduction :**
Si une cellule est entourée par 3 autres, une nouvelle cellule naît.
- **Phase de sous (ou sur) population :**
Toute cellule entourée par 1, 4 cellules ou plus meurt.

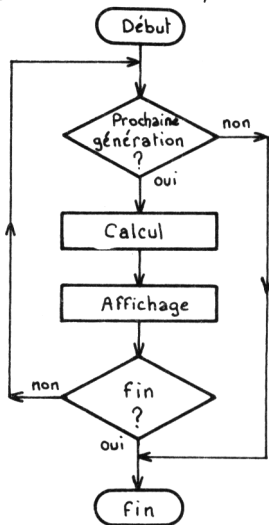
Après avoir entré une population de départ sur une grille 10 x 10, le programme calcule et affiche l'évolution de cette population, génération par génération.

ORGANIGRAMME DU JEU

Le programme possède la structure A.

Après avoir calculé la génération initiale, les générations suivantes sont calculées, puis affichées.

L'organigramme est donc très simple :



LE PROGRAMME

Il se décompose en trois pavés :

Présentation et initialisation

Lignes 1000 à 1650

Présentation

Lignes 1000 à 1360

Initialisation

Lignes 1370 à 1650

Jeu

Lignes 2000 à 2550

Module de commande

Lignes 2000 à 2070

Calcul génération

Lignes 2100 à 2350

Affich. génération

Lignes 2400 à 2550

Fin du jeu

Lignes 3000 à 3130

Commentaires sur l'évolution

LES VARIABLES UTILISEES

A\$	Code couleur de ligne
B	Numéro de la génération
I, J	Index de scrutation de tableaux
R	Nombre de cellules voisines de la cellule courante
T	Tableau du jeu
X, Y	Coordonnées d'une cellule.

```

10 : REM Jeu de la vie
20 :
30 : GOSUB 1000 'Initialisation
40 : GOSUB 2000 'Generations successives
50 : GOSUB 3000 'Fin du jeu
60 :
70 : END
80 : REM *****
1000 : REM Module d'initialisation
1010 :
1020 : PAPER 0 : INK 6
1030 : REM Titre
1040 : PRINT CHR$(12)
1050 : PRINT CHR$(4);CHR$(27);"N";
1060 : PRINT CHR$(27);"A" JEU DE LA VIE";CHR$(4)
1070 : REM Objet
1080 : PRINT:PRINT:PRINT
1090 : A$="D" 'Code couleur = Bleu
1100 : PRINT"OBJET : ";
1110 : GOSUB 1600:PRINT"Evolution d'une population"
1120 : REM Job
1130 : PRINT:PRINT:PRINT
1140 : A$="F" 'Code couleur = Cyan
1150 : PRINT"JOB : ";
1160 : GOSUB 1600:PRINT"Vous allez voir vivre "
1180 : GOSUB 1600:PRINT" et mourir sous vos yeux une"
1190 : GOSUB 1600:PRINT" population de cellules ."
1200 : GOSUB 1600:PRINT" Son evolution est dictee par"
1210 : GOSUB 1600:PRINT" les 3 lois suivantes : "
1220 : GOSUB 1600:PRINT" - Toute cellule entouree par"
1230 : GOSUB 1600:PRINT" 2 ou 3 autres cellules survit."
1240 : GOSUB 1600:PRINT" - Si une cellule est entouree"
1250 : GOSUB 1600:PRINT" par 3 autres cellules ,"
1260 : GOSUB 1600:PRINT" une autre cellule nait ."
1270 : GOSUB 1600:PRINT" - Toute cellule entouree par"
1280 : GOSUB 1600:PRINT" 1,4 cellules ou plus meurt ."

```

```

1290 : REM Debut
1300 : PRINT:PRINT:PRINT
1310 : A$="C" 'Code couleur = Jaune
1320 : PRINT"DEBUT :";
1330 : GOSUB 1600:PRINT"Pour commencer,appuyez sur"
1340 : GOSUB 1600:PRINT" une touche quelconque ."
1350 : GET A$ 'Attente pour passage a l'acquisition des donnees
1360 : REM *****
1370 : REM Entree de la population
1380 :
1390 : CLS
1400 : PRINT CHR$(27);"L";CHR$(27);"A          POPULATION DE DEPA
RT"
1410 : PRINT:PRINT:PRINT:PRINT
1420 : PRINT"Vous allez introduire la generation de";
1430 : PRINT"depart en coordonnees X,Y (<10 et >0)"
1435 : PRINT
1440 : PRINT"La derniere cellule entree aura les"
1450 : PRINT"coordonnees 0,0 ."
1460 : DIM T(10,10) : GENER=0
1470 : PRINT:PRINT
1480 : INPUT"X / Y";X,Y
1490 : IF X+Y=0 THEN PRINT CHR$(17):RETURN
1500 : T(X,Y)=1 'Definition d'une cellule
1510 : GOTO 1480
1520 :
1530 : REM *****
1600 : REM Module de couleur d'une ligne
1610 :
1620 : PRINT CHR$(27);"H";CHR$(27);A$;
1630 : RETURN
1640 :
1650 : REM *****
2000 : REM Evolution de la vie
2010 :
2014 : PRINT:PRINT:PRINT
2015 : PRINT"Appuyez sur une touche quelconque pour";
2016 : PRINT"visualiser la prochaine generation ."
2020 : GET A$:IF A$="F" THEN RETURN
2030 : GOSUB 2100 'Calcul
2040 : GOSUB 2400 'Affichage
2050 : GOTO 2000
2060 :

```

```

2070 : REM *****
2100 : REM Calcul de la generation suivante
2105 : GENER=GENER+1
2110 : FOR I=2 TO 8
2120 :   FOR J=2 TO 8
2130 :     R=0
2140 :     IF T(I-1,J-1)=1 THEN R=R+1
2150 :     IF T(I-1,J)=1 THEN R=R+1
2160 :     IF T(I-1,J+1)=1 THEN R=R+1
2170 :     IF T(I,J-1)=1 THEN R=R+1
2180 :     IF T(I,J+1)=1 THEN R=R+1
2190 :     IF T(I+1,J-1)=1 THEN R=R+1
2200 :     IF T(I+1,J)=1 THEN R=R+1
2210 :     IF T(I+1,J+1)=1 THEN R=R+1
2220 :     IF T(I,J)=1 THEN GOSUB 2250 ELSE GOSUB 2300
2230 :     NEXT J
2240 :   NEXT I
2250 :   REM Mort d'une cellule
2260 :
2270 :   IF R(>)3 AND R(>)2 THEN T(I,J)=0
2280 :   RETURN
2290 :
2300 :   REM Naissance d'une cellule
2310 :
2320 :   IF R=3 THEN T(I,J)=1
2330 :   RETURN
2340 :
2350 : REM *****
:400 : REM Affichage
2410 :
2420 : CLS
2430 : PRINT CHR$(27);"L";CHR$(27);"A           GENERATION ";GENE
R
2440 : PLOT 10,10,"1 2 3 4 5 6 7 8 9"
2450 : FOR I=1 TO 9
2460 :   PLOT 8,10+I,CHR$(I+48)
2470 : NEXT I
:480 : FOR I=1 TO 9
2490 :   FOR J=1 TO 9
2500 :     IF T(I,J)<>0 THEN PLOT 8+2*I,10+J,"0"
2510 :     NEXT J
2520 :   NEXT I
2530 : RETURN
2540 :

```

```
2550 : REM *****
3000 : REM Fin du jeu
3010 :
3020 :   FOR I=1 TO 9
3030 :     FOR J=1 TO 9
3040 :       B=B+F(I,J)
3050 :     NEXT J
3060 :   NEXT I
3070 :   CLS
3080 :   IF B=0 THEN PRINT"Populationn decimée en ";GENER;" generat
ions"
3090 :   IF B(>)0 THEN PRINT"Fin du jeu a la ";GENER;" generation ."
3100 :   PRINT CHR$(17) 'Restitution du curseur
3110 :   RETURN
3120 :
3130 : REM *****
```

Les unités anglo-saxonnes conversion en unités françaises

PRESENTATION

Un utilitaire simple et efficace. Il regroupe les formules de conversion les plus couramment utilisées (longueur, surface, volume, poids, densité et pression).

Ce programme convertit les unités anglaises en unités françaises.

Son petit frère fait la conversion inverse. Seul le premier programme est décrit.

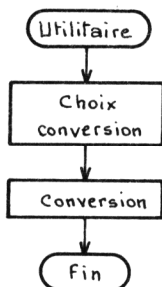
ORGANIGRAMME DE L'UTILITAIRE :

Le programme a la structure A.

Il se déroule en deux temps :

- . Choix de la conversion,
- . Conversion et affichage.

L'organigramme est simple :



LE PROGRAMME

Il se décompose en deux pavés :

Présentation et initialisation

Lignes 1000 à 2180

Module de commande Lignes 1000 à 1060

Présentation Lignes 1100 à 1650

Initialisation des variables Lignes 1700 à 2180

Stockage des unités anglaises, françaises, correspondances et coefficient de conversion.

Conversion

Lignes 3000 à 4340

Module de conversion Lignes 3000 à 3070

Choix conversion Lignes 3100 à 4120

Conversion Lignes 4200 à 4340

LES VARIABLES UTILISEES

A\$	Code couleur de la ligne
I	Index de boucle
GB	Nombre à convertir
F	Nombre converti
R	Code de choix de la conversion
S	Code de sous-choix de la conversion
T	Tableau des coefficients de la conversion
T1\$	Tableau des unités GB
T2\$	Tableau des unités F
T3\$	Tableau des correspondances.


```

10 : REM Les unites anglo-saxonnes Conversion GB -> F
20 :
30 : GOSUB 1000 ' Presentation et Initialisation
40 : GOSUB 3000 ' Conversion GB -> F
50 :
60 : END
70 : REM *****
1000 : REM Presentation et Initialisation des variables
1010 :
1020 : GOSUB 1100 'Presentation
1030 : GOSUB 1700 'Initialisation des variables
1040 :
1050 : RETURN
1060 : REM *****
1100 : REM Presentation
1110 :
1120 : INK 6 : PAPER 0
1130 :
1140 : REM Titre
1150 :
1160 : PRINT CHR$(12)
1170 : PRINT CHR$(4);CHR$(27);"N";
1180 : PRINT CHR$(27);"A LES UNITES ANGLO-SAXONNES";CHR$(4)

1190 :
1200 : REM Objet
1210 :
1220 : PRINT:PRINT:PRINT:PRINT:PRINT
1230 : A$="D" 'Code couleur = Bleu
1240 : PRINT"OBJET :";
1250 : GOSUB 1600:PRINT"Conversion des unites "
1260 : GOSUB 1600:PRINT" Anglaises en unites Francaises."
1270 :
1280 : REM Job
1290 :
1300 : PRINT:PRINT:PRINT
1310 : A$="F" 'Code couleur = Cyan
1320 : PRINT"JOB :";
1330 : GOSUB 1600:PRINT"Ce programme permet d'obtenir";
1340 : GOSUB 1600:PRINT" la conversion d'unites :";
1350 : GOSUB 1600:PRINT" - de Longueur ,";
1360 : GOSUB 1600:PRINT" - de Surface ,";
1370 : GOSUB 1600:PRINT" - de Volume ,";
1380 : GOSUB 1600:PRINT" - de Poids ,";
1390 : GOSUB 1600:PRINT" - de Densite ,";
1400 : GOSUB 1600:PRINT" - de Pression ,";
1410 :
1420 : REM Debut

```

```

1430 :
1440 : PRINT:PRINT:PRINT
1450 : A$="C" 'Code couleur = Jaune
1460 : PRINT"DEBUT :";
1470 : GOSUB 1600:PRINT"Pour effectuer une conversion";
1480 : GOSUB 1600:PRINT" appuyez sur une touche-clavier"
1490 : GET A$ 'Attente de l'appui sur une touche
1500 :
1510 : RETURN
1520 : REM #####
1600 : REM Couleur d'une ligne
1610 :
1620 : PRINT CHR$(27);"H";CHR$(27);A$;
1630 :
1640 : RETURN
1650 : REM #####
1700 : REM Initialisation des variables
1710 :
1720 : DIM T1$(17),T2$(17),T3$(17),T(17) 'Tableaux de constantes
1730 :
1740 : REM Unites Anglaises
1750 :
1760 : FOR I=1 TO 17
1770 : READ T1$(I)
1780 : NEXT I
1790 :
1800 : REM Unites Francaises
1810 :
1820 : FOR I=1 TO 17
1830 : READ T2$(I)
1840 : NEXT I
1850 :
1860 : REM Correspondances
1870 :
1880 : FOR I=1 TO 17
1890 : READ T3$(I)
1900 : NEXT I
1910 :
1920 : REM Coefficients de conversion
1930 :
1940 : FOR I=1 TO 17
1950 : READ T(I)
1960 : NEXT I
1970 :
1980 DATA "Inches","Foot","Inches^2","Foot^2","Inches^3","Foot^3","
Gallon"
1990 DATA "Pound","Ounce","Grain","Lb/Ft^3","Lb/Inch^2","BTU","BTU"
,"BTU"
2000 DATA "BTU","F"

```

```

2010 :
2020 DATA "cm","cm","cm^2","cm^2","cm^3","cm^3","Litres","gr","gr",
"gr"
2030 DATA "gr/cm^3","kg/cm^2","Calories","Joules","Kwh","Mev","Celcius"
2040 :
2050 DATA "1 Inch = 2.54 cm","1 Foot = 30.48 cm","1 Inch^2 = 6.45 c
m^2"
2060 DATA "1 Foot^2 = 929 cm^2","1 Inch^3 = 16.39 cm^3"
2070 DATA "1 Foot^3 = 28316.8 cm^3","1 Gallion = 4.54 Litres"
2080 DATA "1 Pound = 453.59 gr","1 Ounce = 28.35 gr","1 Grain = 0.0
65 gr"
2090 DATA "1 Lb/Ft^3 = g/cm^3","1 Pound/Inch^2 = 0.07 Kg/cm^2"
2100 DATA "1 BTU = 256 Calories","1 BTU = 1054.8 Joules"
2110 DATA "1 BTU = 293.10^-4 Kwh","1 BTU = 6.59.10^-5 Mev","1 F = 1
.8 C + 32"
2120 :
2130 DATA 2.54,30.48,6.4516,929,16.387,28316.8,4.5435,453.5924,28.3
495
2140 DATA 0.064798,0,0.0703,252,1054.8,0.0293,6.59E-5,1.8
2150 :
2160 :
2170 : RETURN
2180 : REM #####
3000 : REM Conversion GB -> F
3010 :
3020 : CLS
3030 : GOSUB 3100 'Demande de conversion
3040 : GOSUB 4200 'Conversion
3050 :
3060 : RETURN
3070 : REM #####
3100 : REM Demande de conversion
3110 :
3120 : GOSUB 1600:PRINT"Dois-je faire une conversion de :"
3140 : PRINT:PRINT
3150 : PRINT" (1) Longueur"
3160 : PRINT" (2) Surface"
3170 : PRINT" (3) Volume"
3180 : PRINT" (4) Poids"
3190 : PRINT" (5) Densite"
3200 : PRINT" (6) Pression"
3210 : PRINT" (7) Quantite de chaleur "
3215 : PRINT" (8) Temperature"
3220 :
3230 : PRINT:INPUT" Choix ";R
3240 :
3250 : IF R<1 OR R>8 THEN GOTO 3100
3260 : ON R GOTO 3300,3400,3500,3600,3700,3800,3900,4000

```

```
3270 :
3300 : REM Longueur
3310 :
3320 : PRINT:PRINT"Je peux faire les conversions : "
3330 : PRINT:PRINT" (1) ";T1$(1);" -> ";T2$(1);" et"
3340 : PRINT" (2) ";T1$(2);" -> ";T2$(2)
3350 : PRINT:INPUT" Choix";S
3360 : R=S
3370 : GOTO 4100
3380 :
3400 : REM Surface
3410 :
3420 : PRINT:PRINT"Je peux faire les conversions : "
3430 : PRINT:PRINT" (1) ";T1$(3);" -> ";T2$(3);" et"
3440 : PRINT" (2) ";T1$(4);" -> ";T2$(4)
3450 : PRINT:INPUT" Choix";S
3460 : R=S+2 'Numero de la conversion
3470 : GOTO 4100
3480 :
3500 : REM Volume
3510 :
3520 : PRINT:PRINT"Je peux faire les conversions : "
3530 : PRINT:PRINT" (1) ";T1$(5);" -> ";T2$(5)
3540 : PRINT" (2) ";T1$(6);" -> ";T2$(6);" et"
3550 : PRINT" (3) ";T1$(7);" -> ";T2$(7)
3560 : PRINT:INPUT" Choix ";S
3570 : R=S+4
3580 : GOTO 4100
3590 :
3600 : REM Po ds
3610 :
3620 : PRINT:PRINT"Je peux faire les conversions : "
3630 : PRINT:PRINT" (1) ";T1$(8);" -> ";T2$(8)
3640 : PRINT" (2) ";T1$(9);" -> ";T2$(9);" et"
3650 : PRINT" (3) ";T1$(10);" -> ";T2$(10)
3660 : PRINT:INPUT" Choix ";S
3670 : R=S+7
3680 : GOTO 4100
3690 :
3700 : REM Densite
3710 :
3720 : PRINT:PRINT"Je peux faire la conversion : "
3730 : PRINT:PRINT" (1) ";T1$(11);" -> ";T2$(11)
3740 : R=11:GOTO 4100
3750 :
```

```

3800 : REM Pression
3810 :
3820 : PRINT:PRINT"Je peux faire la conversion : "
3830 : PRINT:PRINT" (1) ";T1$(12);" -> ";T2$(12)
3840 : R=12:GOTO 4100
3850 :
3900 : REM Quantite de chaleur
3910 :
3920 : PRINT:PRINT"Je peux faire les conversions : "
3930 : PRINT:PRINT" (1) ";T1$(13);" -> ";T2$(13)
3940 : PRINT" (2) ";T1$(14);" -> ";T2$(14)
3950 : PRINT" (3) ";T1$(15);" -> ";T2$(15)
3960 : PRINT" (4) ";T1$(16);" -> ";T2$(16)
3970 : PRINT:INPUT" Choix ";S
3980 : R=S+12:GOTO 4100
3990 :
4000 : REM Temperature
4010 :
4020 : PRINT:PRINT"Je peux faire la conversion : "
4030 : PRINT:PRINT" (1) ";T1$(17);" -> ";T2$(17)
4040 : R=17
4050 :
4100 : RETURN
4110 :
4120 : REM *****
4200 : REM Conversion
4210 :
4220 : CLS
4230 : PRINT:PRINT:PRINT
4240 : GOSUB 1600:PRINT"Formule de conversion : "
4250 : PRINT:GOSUB 1600:PRINT T3$(R)
4260 : PRINT:PRINT:PRINT
4270 : PRINT"Donnez le nombre de ";T1$(R)
4280 : INPUT GB
4290 : F=GB*T(R)
4300 : IF R=17 THEN F=F+32
4310 :
4320 : PRINT:PRINT:PRINT"Cela represente ";INT(F#1E7)/1E7;T2$(R)
4330 :
4340 : RETURN

```

PROGRAMME DE CONVERSIONS DES UNITES FRANÇAISES ET

UNITES ANGLO-SAXONNES

```

10 : REM Les Unites Anglo-Saxonnes / Conversion F -> GB
20 :
30 : GOSUB 1000 'Presentation et Initialisation
40 : GOSUB 3000 'Conversion
50 :
60 : END
70 : RETURN
80 : REM *****
1000 : REM Presentation et Initialisation
1010 :
1020 : GOSUB 1100 'Presentation
1030 : GOSUB 1700 'Initialisation
1040 :
1050 : RETURN
1060 : REM *****
1100 : REM Presentation
1110 :
1120 : PAPER 0 : INK 5 : PRINT CHR$(17)
1130 : REM Titre
1140 :
1150 : PRINT CHR$(12)
1160 : PRINT CHR$(4);CHR$(27);"N";
1170 : PRINT CHR$(27);"A LES UNITES ANGLO-SAXONNES";CHR$(4
)
1180 :
1190 : REM Objet
1200 :
1210 : PRINT:PRINT:PRINT:PRINT
1220 : A$="D" 'Code couleur = Bleu
1230 : PRINT"OBJET :";
1240 : GOSUB 1600:PRINT"Conversion des unites"
1250 : GOSUB 1600:PRINT" Francaises en unites Anglaises."
1260 :
1270 : REM Job
1280 :
1290 : PRINT:PRINT:PRINT
1300 : A$="F" 'Code couleur = Cyan
1310 : PRINT"JOB :";
1320 : GOSUB 1600:PRINT"Ce programme permet d'obtenir";
1330 : GOSUB 1600:PRINT" la conversion d'unites :";
1340 : PRINT:GOSUB 1600:PRINT" - de longueur,"

```

```

1350 GOSUB 1600:PRINT"      - de surface,"
1360 :   GOSUB 1600:PRINT"      - de volume,"
1370 :   GOSUB 1600:PRINT"      - de poids,"
1380 :   GOSUB 1600:PRINT"      - de densite,"
1390 :   GOSUB 1600:PRINT"      - de pression,"
1400 :   GOSUB 1600:PRINT"      - de temperature."
1410 :
1420 :   REM Debut
1430 :
1440 :   PRINT:PRINT:PRINT
1450 :   A$="C" 'Code couleur = Jaune
1460 :   PRINT"DEBUT :";
1470 :   GOSUB 1600:PRINT"Pour commencer , appuyez sur"
1480 :   GOSUB 1600:PRINT"      une touche quelconque."
1490 :   GET A$ 'Attente de l'appui sur une touche
1500 :
1510 : RETURN
1520 : REM #####
1600 : REM Couleur d'une ligne
1610 :
1620 :   PRINT CHR$(27);"H";CHR$(27);A$;
1630 :
1640 : RETURN
1650 : REM #####
1700 : REM Initialisation des constantes
1710 :
1720 :   DIM T1$(17),T2$(17),T3$(17),T(17)
1730 :
1740 :   REM Unites Francaises
1750 :
1760 :     FOR I=1 TO 17
1770 :       READ T1$(I)
1780 :     NEXT I
1790 :
1800 :   REM Unites Anglaises
1810 :
1820 :     FOR I=1 TO 17
1830 :       READ T2$(I)
1840 :     NEXT I
1850 :
1860 :   REM Correspondances
1870 :
1880 :     FOR I=1 TO 17
1890 :       READ T3$(I)
1900 :     NEXT I
1910 :

```

```

1920 DATA "cm","ca","cm^2","ca^2","cm^3","ca^3","litres","gr","gr","
gr"
1930 DATA "gr/cm^3","Kg/cm^2","Calories","Joules","Kwh","Mev","Celcius"
1940 :
1950 DATA "inches","foot","inches^2","foot^2","inches^3","foot^3","
gallon"
1960 DATA "pound","ounce","grain","Lb/ft^3","Lb/inch^2","BTU","BTU"
1970 DATA "BTU","BTU","F"
1980 :
1990 DATA "1 cm = 0.394 inch","1 cm = 0.0328 foot","1 cm^2 = 0.155
inch^2"
2000 DATA "1 cm^2 = 1.08 E3 foot^2","1 cm^3 = 6.1 E-2 inch^3"
2010 DATA "1 cm^3 = 3.53 E-5 foot^3","1 litre = 0.220 Gallon"
2020 DATA "1 gr = 2.20462 E-3 pound",,"1 gr/cm^3 = 1.6 E2 lb/ft^3"
2030 DATA "1 Kg/cm^2 = 14.2 lb/inch^2","1 Calorie = 3.97 E-3 BTU"
2040 DATA "1 Joule = 9.49 E-4 BTU","1 Kwh = 3417 BTU","1 eV = 1.52
E-16 BTU"
2050 DATA "1 Celcius = 5/9 (F - 32)"
2060 :
2070 :   REM Coefficients de conversion
2080 :
2090 :   FOR I=1 TO 17
2100 :     READ T(I)
2110 :     NEXT I
2120 :
2130 DATA 0.394,0.0328,0.155,1.08E3,6.1E-2,3.53E-5,0.22,2.20462E-3
2140 DATA 0,0,1.6E2,14.214,3.97E-3,9.49E-4,3417,1.52E-16,.5555555
2150 :
2160 : RETURN
2170 : REM #####
3000 : REM Initialisation de la conversion et Conversion
3010 :
3020 :   GOSUB 3100 'Initialisation de la conversion
3030 :   GOSUB 4100 'Conversion
3040 :
3050 : RETURN
3060 : REM #####
3100 : REM Initialisation de la conversion
3110 :
3120 :   CLS : PRINT CHR$(17)
3130 :   PRINT:PRINT
3140 :   GOSUB 1600:PRINT"Desirez-vous faire une conversion de:"
3150 :   PRINT:PRINT"  (1) Longueur"
3160 :   PRINT"  (2) Surface"
3170 :   PRINT"  (3) Volume"

```



```

3180 : PRINT" (4) Poids"
3190 : PRINT" (5) Densite"
3200 : PRINT" (6) Pression"
3210 : PRINT" (7) Quantite de chaleur"
3215 : PRINT" (8) Temperature"
3220 :
3230 : PRINT:PRINT:INPUT "Choix: ";R:PRINT:PRINT:PRINT:PRINT
3235 : IF R<1 OR R>8 THEN PRINT"1<Choix<8":GOTO 3230
3240 :
3250 : ON R GOSUB 3300,3400,3500,3600,3700,3800,3900,4000
3260 :
3270 : RETURN
3280 : REM *****
3300 : REM Conversion de longueur
3310 :
3320 : GOSUB 1600:PRINT"Je peux faire les conversions:"
3330 : PRINT:PRINT" (1) cm -> inches et"
3340 : PRINT" (2) cm -> foot ."
3350 : PRINT:INPUT"Choix ";R
3360 : S=R:RETURN
3370 :
3380 : REM *****
3400 : REM Conversion de surface
3410 :
3420 : GOSUB 1600:PRINT"Je peux faire les conversions:"
3430 : PRINT:PRINT" (1) cm^2 -> inches^2"
3440 : PRINT" (2) cm^2 -> foot^2"
3450 : PRINT:INPUT"Choix ";R
3460 : S=R+2:RETURN
3470 :
3480 : REM *****
3500 : REM Conversion de volume
3510 :
3520 : GOSUB 1600:PRINT"Je peux faire les conversions:"
3530 : PRINT:PRINT" (1) cm^3 -> inches^3"
3540 : PRINT" (2) cm^3 -> foot^3"
3550 : PRINT:INPUT"Choix ";R
3560 : S=R+4:RETURN
3570 :
3580 : REM *****
3600 : REM Conversion de poids
3610 :
3620 : GOSUB 1600:PRINT"Je peux faire les conversions:"
3630 : PRINT:PRINT" (1) gr -> pound"
3640 : PRINT" (2) gr -> ounce"
3650 : PRINT" (3) gr -> grain"
3660 : PRINT:INPUT"Choix ";R
3670 : S=R+6:RETURN
3680 :

```

```
3690 : REM #####
3700 : REM Conversion de densite
3710 :
3720 : GOSUB 1600:PRINT"Je peux faire la conversion:"
3730 : PRINT:PRINT" gr/cm^3 -> Lb/ft^3"
3740 : GET B$
3750 : S=11:RETURN
3760 :
3770 : REM #####
3800 : REM Conversion de pression
3810 :
3820 : GOSUB 1600:PRINT"Je peux faire la conversion:"
3830 : PRINT:PRINT" Kg/cm^2 -> Lb/inch^2"
3840 : GET B$
3850 : S=12:RETURN
3860 :
3870 : REM #####
3900 : REM Conversion de quantite de chaleur
3910 :
3920 : GOSUB 1600:PRINT"Je peux faire les conversions:"
3930 : PRINT:PRINT:PRINT" (1) Calories -> BTU"
3940 : PRINT" (2) Joules -> BTU"
3950 : PRINT" (3) Kwh -> BTU"
3960 : PRINT" (4) MeV -> BTU"
3970 : PRINT:INPUT"Choix ";R
3980 : S=12+R:RETURN
3990 :
3995 : REM #####
4000 : REM Conversion de temperature
4010 :
4020 : GOSUB 1600:PRINT"Je peux faire la conversion:"
4030 : PRINT:PRINT" Celcius -> Farenheit"
4040 : S=17:GET B$:RETURN
4050 :
4060 : REM #####
4100 : REM Conversion
4110 :
4120 : CLS : INK 6
4130 : GOSUB 1600:PRINT"Formule de conversion :"
4140 : PRINT:PRINT:PRINT T3$(S)
4150 : PRINT:PRINT:PRINT:PRINT"Nombre de ";T1$(S);:INPUT F
4170 : GB=F*T(S):IF S=17 THEN GB=GB-17.77777
4180 : PRINT:PRINT:PRINT"Cela represente ";INT(GB*1E5)/1E5;T2$(S)
4190 :
4200 : RETURN
```

**L'ORIC LE SON
ET LE DESSIN**

Oric piano

PRESENTATION

Vous jouez du piano ? Non ? Alors il est temps de vous y mettre !

Ce programme simule le clavier d'un piano (malheureusement à une seule octave). Les touches du clavier reprennent la disposition des tons et demi-tons d'un clavier de piano. Les notes et leur correspondant-touche sont indiqués en haut de l'écran en haute résolution.

Musiciens, à vos claviers...

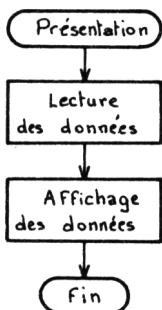
ORGANIGRAMME DU PROGRAMME

Le programme a la structure A.

L'organigramme de présentation est donné, car il ne ressemble pas aux autres organigrammes de présentation.

Présentation

Elle consiste en l'affichage (en haute résolution) d'une portée, de la clé de Sol, et des notes avec leur correspondant-lettre.

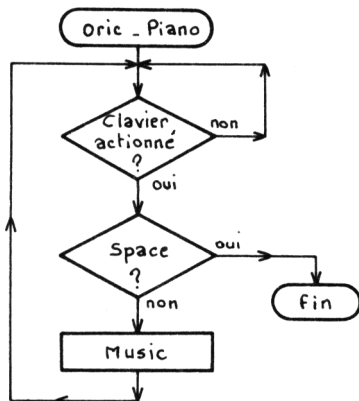


Oric-Piano

Le programme de jeu consiste en une attente de frappe de touche clavier. La touche "SPACE" commande la sortie du programme.

Les touches-notes commandent le canal son n° 1 de l'ORIC grâce à l'ordre BASIC "Music a, b, c, d" avec :

- a : canal
- b : octave
- c : note
- d : volume



LE PROGRAMME

Il se décompose en trois pavés :

Présentation et initialisation

Lignes 1000 à 1980
Acquisition et affichage clé, portée, notes et correspondants-touches.

ORIC joue

Lignes 2000 à 2150
Activation du canal 1 par MUSIC

Fin du jeu

Lignes 3000 à 3080
Restitution du curseur et commentaires.

LES VARIABLES UTILISEES

A	Donnée lue (clé, note, titre)
B	Donnée lue (touche-note) puis attribut sonore
B\$	Touche-note actionnée
I	Index de boucle
T	Tableau des attributs sonores (paramètre de MUSIC pour activer telle ou telle note).

```

10 : REM ORIC-PIANO
20 :
30 : GOSUB 1000 'Presentation et Initialisation
40 : GOSUB 2000 'ORIC joue
50 : GOSUB 3000 'Fin du jeu
60 :
70 : END
80 : REM *****
1000 : REM Affichage Presentation
1010 :
1020 : REM Cle de Sol
1030 :
1040 : FOR I=0 TO 15
1050 : READ A : POKE 46464+I,A
1060 : NEXT I
1070 :
1080 : REM Notes
1090 :
1100 : FOR I=0 TO 23
1110 : READ A : POKE 46480+I,A
1120 : NEXT I
1130 :
1140 : REM Titre
1150 :
1160 : HIRES : INK 6 : PRINT CHR$(17)
1170 : CURSET 50,10,0
1180 : FOR I=1 TO 13
1190 : READ A : CHAR A,0,1
1200 : CURSET 50+I*10,10,0
1210 : NEXT I
1220 :
1300 : REM Placement Cle et Notes
1310 :
1320 : CURSET 33,44,0
1330 : CHAR 48,0,1
1340 : CURSET 33,52,0
1350 : CHAR 49,0,1
1360 :
1370 :

```

```
1380 : CURSET 45,58,0:CHAR 51,0,1:CURSET 45,50,0:CHAR 52,0,1
1390 : CURSET 55,61,0:CHAR 35,0,1
1400 : CURSET 60,58,0:CHAR 51,0,1:CURSET 60,50,0:CHAR 52,0,1
1420 : CURSET 75,56,0:CHAR 50,0,1:CURSET 75,48,0:CHAR 52,0,1
1430 : CURSET 85,59,0:CHAR 35,0,1
1440 : CURSET 90,56,0:CHAR 50,0,1:CURSET 90,48,0:CHAR 52,0,1
1450 : CURSET100,54,0:CHAR 51,0,1:CURSET100,46,0:CHAR 52,0,1
1460 : CURSET 110,52,0:CHAR 50,0,1:CURSET 110,44,0:CHAR 52,0,1
1470 : CURSET 120,55,0:CHAR 35,0,1
1480 : CURSET 125,52,0:CHAR 50,0,1:CURSET 125,44,0:CHAR 52,0,1
1490 : CURSET 135,50,0:CHAR 51,0,1:CURSET 135,42,0:CHAR 52,0,1
1500 : CURSET 145,53,0:CHAR 35,0,1
1510 : CURSET 150,50,0:CHAR 51,0,1:CURSET 150,42,0:CHAR 52,0,1
1520 : CURSET 160,48,0:CHAR 50,0,1:CURSET 160,40,0:CHAR 52,0,1
1530 : CURSET 170,51,0:CHAR 35,0,1
1540 : CURSET 175,48,0:CHAR 50,0,1:CURSET 175,40,0:CHAR 52,0,1
1550 : CURSET 185,46,0:CHAR 51,0,1:CURSET 185,38,0:CHAR 52,0,1
1555 :
1560 : REM Portee
1570 :
1580 : FOR I=1 TO 5
1590 : CURSET 40,40+I*4,1
1600 : DRAW 155,0,1
1610 : NEXT I
1620 :
1630 : REM Designation des Touches-Notes
1640 : FOR I=1 TO 12
1650 : READ A,B
1660 : CURSET A,70,0
1670 : CHAR B,0,1
1680 : NEXT I
1690 :
1700 : FOR I=1 TO 34
1710 : READ A
1720 : CURSET I*6+20,140,0
1730 : CHAR A,0,1
1740 : NEXT I
1750 : FOR I=1 TO 32
1760 : READ A
1770 : CURSET I*6+20,160,0
1780 : CHAR A,0,1
1790 : NEXT I
1800 :
1810 : RETURN
```



```

1820 : REM *****
1830 : REM Donnees de l'initialisation
1840 :
1900 DATA 0,0,12,10,10,10,12,24,28,42,42,28,8,40,24,0
1910 DATA 63,4,4,4,63,28,28,28,4,4,63,4,4,28,63,28,0,0,4,4,4,4,4
1920 DATA 79,82,73,67,32,77,85,83,73,67,73,69,78
1930 DATA 45,65,58,87,75,83,88,69,100,68,110,70,123,84
1940 DATA 135,71,148,89,160,72,173,85,185,74
1950 DATA 80,111,117,114,32,106,111,117,101,114,44,97,112,112,117,1
21,101
1960 DATA 122,32,115,117,114,32,108,101,115,32,116,111,117,99,104,1
01,115
1970 DATA 99,111,114,114,101,115,112,111,110,100,97,110,116,32,97,1
17,120
1980 DATA 32,110,111,116,101,115,32,100,101,115,105,114,101,101,115
2000 : REM ORIC joue
2010 :
2015 : DIM T(25)
2020 : FOR I=1 TO 25
2030 :   READ A:T(I)=A
2040 : NEXT I
2045 :
2050 : PRINT CHR$(6) 'Curseur muet
2060 :
2070 : GET B$
2080 : IF B$=" " THEN RETURN
2090 : B=T(ASC(B$)-64)
2100 : IF B<>0 THEN MUSIC 1,5,B,7
2110 :
2120 : GOTO 2070
2130 :
2140 : RETURN
2150 : REM *****
2900 DATA 1,0,0,5,4,6,8,10,0,12,0,0,0,0,0,0,0,3,7,11,0,2,0,9
3000 : REM Fin du programme
3010 :
3020 : PLAY 0,0,0,0
3030 : PRINT CHR$(6)
3040 : PRINT CHR$(17)
3050 : LORES 0:CLS
3060 : PLOT 08,10,"Au revoir et a bientot"
3070 :
3080 : RETURN

```

Oric joue

PRESENTATION

Outre les quatre commandes sonores préprogrammées, ORIC possède des ordres sonores très sophistiqués.

Les trois programmes qui suivent, mettent en oeuvre deux de ces ordres **MUSIC** et **PLAY**.

Avant d'entrer dans le détail de chaque programme, examinons de plus près ces deux commandes.

MUSIC : Possède 4 arguments : **canal**, **octave**, **note**, **volume**

Par la suite :

- Canal sonore est validé : **canal** = 1
- Le volume est mis à 0 pour pouvoir être commandé par l'enveloppe choisie : **volume** = 0
- L'octave et la note varient au rythme du morceau

PLAY : Possède 4 arguments : **validation canaux musicaux**,
validation canaux de bruit,
type d'enveloppe et **durée d'enveloppe**.

Par la suite :

- Aucun canal de bruit n'est utilisé :
validation canaux de bruit = 0
- Le type d'enveloppe est fixé :
type d'enveloppe = 0 ou 1

- La validation des canaux musicaux est fixée :
validation des canaux musicaux = 1
- La durée de l'enveloppe est fonction du morceau.

LE PROGRAMME

Décrivons la façon de procéder dans chaque programme : (POP CORN, CHARLOT et CLASSIQUE).

Tout d'abord, ces trois programmes possèdent un tronc commun : leur structure. Cette structure comporte deux étapes :

- **La lecture des données** (Note/Octave/Validation canaux musicaux ou Durée de pause entre 2 notes) jusqu'à la fin du morceau.
- **Activation des ordres** MUSIC et PLAY.

Pour les programmes "POP CORN" et "CHARLOT", la donnée C commande le temps de pause entre 2 notes et la durée de l'enveloppe.

Pour le programme "CLASSIQUE", la donnée C commande le temps de pause entre 2 notes. La durée de l'enveloppe est fixe.

```

10 : REM Pop corn
20 :
30 : FOR I=1 TO 74
40 : READ A,B,C
50 : IF C=7 THEN D=4000 ELSE D=600
60 : MUSIC 1,B,A,0
70 : PLAY C,0,1,D
80 : IF C=1 THEN WAIT 20 ELSE WAIT 100
90 : IF I=7 OR I=14 OR I=29 OR I=45 OR I=52 OR I=59 THEN WAIT
   30
100 : NEXT I
110 :
120 : END
130 : REM *****
1000 DATA 5,4,1,3,4,1,5,4,1,12,3,1,8,3,1,12,3,1,5,3,1,5,4,1,3,4,1,5
,4,1
1010 DATA 12,3,1,8,3,1,12,3,1,5,3,1,5,4,1,7,4,1,8,4,1,7,4,1,8,4,1,5
,4,1
1020 DATA 7,4,1,5,4,1,7,4,1,3,4,1,5,4,1,3,4,1,5,4,1,1,4,1,5,4,1
1030 DATA 8,4,7,5,4,7,12,3,7,8,4,1,10,4,1,12,4,7,10,4,7,8,4,7
1040 DATA 8,4,7,5,4,7,12,3,7,8,4,1,10,4,1,12,4,7,10,4,7,8,4,7
1050 DATA 5,4,1,3,4,1,5,4,1,12,3,1,8,3,1,12,3,1,5,3,1,5,4,1,3,4,1,5
,4,1
1060 DATA 12,3,1,8,3,1,12,3,1,5,3,1,5,4,1,7,4,1,8,4,1,7,4,1,8,4,1,5
,4,1
1070 DATA 7,4,1,5,4,1,7,4,1,3,4,1,5,4,1,3,4,1,5,4,1,1,4,1,5,4,1

```

```

10 : REM Charlot
20 :
30 : FOR I=1 TO 76
40 :   READ A,B,C
50 :   IF C=7 THEN D=1800:WAIT 5 ELSE D=600
60 :   MUSIC 1,B,A,0
70 :   PLAY 1,0,1,D
80 :   IF C=7 THEN WAIT 18
90 :   WAIT 15
100 :   IF I=12 OR I=25 OR I=29 OR I=33 OR I=37 THEN WAIT 30
110 :   IF I=44 OR I=56 OR I=69 OR I=74 THEN WAIT 30
120 : NEXT I
130 : REM *****
1000 DATA 5,4,1,1,4,1,8,3,1,5,3,1,1,3,1,5,3,1,8,3,1,5,3,1,8,3,1,10,
3,7
1010 DATA 5,3,1,8,3,1,5,4,1,1,4,1,8,3,1,5,3,1,1,3,1,5,3,1,8,3,1,5,3
,1
1020 DATA 8,3,1,1,4,1,12,3,1,3,4,1,1,4,1,5,4,1,8,4,7,5,4,1,1,4,1,5,
4,1
1030 DATA 1,4,1,8,3,7,10,3,1,5,4,1,8,4,7,5,4,1,1,4,1,5,4,1,1,4,1,8,
3,1
1040 DATA 1,4,1,1,4,1,12,3,1,1,4,1
1050 DATA 5,4,1,1,4,1,8,3,1,5,3,1,1,3,1,5,3,1,8,3,1,5,3,1,8,3,1,10,
3,7
1060 DATA 5,3,1,8,3,1,5,4,1,1,4,1,8,3,1,5,3,1,1,3,1,5,3,1,8,3,1,5,3
,1
1070 DATA 8,3,1,1,4,1,12,3,1,3,4,1,1,4,1,8,3,1,8,3,1,8,3,1,9,3,1,8,
3,1
1080 DATA 8,3,1,1,4,1

```

```

10 : REM Classique
20 :
30 : FOR I=1 TO 73
40 :   READ A,B,C
50 :   MUSIC 1,B,A,0
60 :   PLAY 1,0,0,600
70 :   WAIT 15
80 :   IF C=1 THEN WAIT 14
90 :   NEXT I
100 :
110 : END
120 : REM *****
1000 DATA 1,3,1,6,3,2,5,3,2,6,3,1,1,3,1,8,3,1,1,3,1,9,3,1,6,3,1,11,
3,1
1010 DATA 6,3,1,1,4,1,6,3,1,2,4,1,6,3,1,5,3,2,6,3,2,8,3,2,5,3,2,1,3
,1
1020 DATA 1,4,2,11,3,2,10,3,2,11,3,2,1,4,2,10,3,2,6,3,2,8,3,2,10,3,
2
1030 DATA 6,3,2,4,3,1,2,3,2,1,3,2,2,3,1,11,3,2,9,3,2,8,3,2,9,3,2,11
1040 DATA 8,3,2,4,3,2,6,3,2,8,3,2,4,3,2,2,3,1,1,3,2,11,2,2,1,3,1
1050 DATA 9,3,2,8,3,2,6,3,2,8,3,2,9,3,2,6,3,2,2,3,2,4,3,2,6,3,2,2,3
,2
1060 DATA 1,3,1,11,2,2,10,2,2,11,2,1,2,3,1,1,3,1,6,3,1,5,3,2,6,3,2
1070 DATA 8,3,2,5,3,2,5,3,1,6,3,2,5,3,2,6,3,2

```

Le chien

PRESENTATION

Ce programme réalise le dessin d'un chien sur une imprimante 80 colonnes en basse résolution, grâce aux lettres de l'alphabet. Plusieurs lettres sont utilisées afin d'obtenir différents contrastes.

ORGANIGRAMME DU PROGRAMME :

Le programme possède une structure linéaire. Il consiste en une suite d'ordres d'impression "**LPRINT**".

```

10 LPRINT SPC(18)*"MMMM"
20 LPRINT SPC(17)*"MMMMMM"
30 LPRINT SPC(16)*"MMMMIIIMMMMM"
40 LPRINT SPC(14)*"MMMMIIIMMMMMMM"
50 LPRINT SPC(12)*"MMMMVVMMIIIVMMIIIM"
60 LPRINT SPC(10)*"MMMMVVVVVVVVVVVVMMVIMM"
70 LPRINT SPC(9)*"M,,VVVVVVVVVVVVMMVIVM"
80 LPRINT SPC(9)*"MVMVVVVVVVVVVVV MVMVIVMM"
90 LPRINT SPC(9)*"MVMVV VVVVVVMMMMVVVVMM"
100 LPRINT SPC(8)*"MVMIVMVMVMV M VM MVMVVVVMM"
110 LPRINT SPC(7)*"MVMIVM VM M VV--VVVVVVMM"
120 LPRINT SPC(7)*"MVMIVM--MMMMMVMVVVVVVMMMM"
130 LPRINT " MVMVVIIVMMVM MVMVVVVVVMM"
140 LPRINT " MVMIVMVMVVMM MVMVVVVVVMM"
150 LPRINT " MVMVVIIMIIIVMMVVMMVMVIVMMVM"
160 LPRINT " MIMMVIIMVIIIMMMIIVMIVMMVM"
170 LPRINT " MIMMVMIVVV IM M VMVVMVM"
180 LPRINT " MIIVMVIM MMM VM VM"
190 LPRINT " MIMVMVMV VMVM VM VM"
200 LPRINT " MIMVIVVVMMMMVM VM"
210 LPRINT SPC(9)*"MVMVIMIIIMMVM VM"
220 LPRINT SPC(10)*"MVMIMMMMMMMIM V"
230 LPRINT SPC(11)*"MI MVMVVIIM,MM"
240 LPRINT SPC(11)*"MI ,MM VM,MM"
250 LPRINT SPC(12)*"M IMVMMM,MM"
260 LPRINT SPC(14)*"MIVVVV"IMMM"
270 LPRINT SPC(13)*"MV,IVVII,MMMM"
280 LPRINT SPC(12)*"MVMV,,,VMMMM"
290 LPRINT SPC(12)*"MVMVMK-MVMVMK"
300 LPRINT SPC(11)*"MVMIVIVVIIIVMMMM"
310 LPRINT SPC(11)*"MVMIIIIIVVVMMMM"
320 LPRINT SPC(10)*"MVMIIIIIVIIIVMMKCK"
330 LPRINT SPC(10)*"MVMIIIIIVIIIIVMMMM"
340 LPRINT SPC(10)*"MVMVIIIMIIIIIVMMMM"
350 LPRINT SPC(10)*"MVMIVIVMIIIIIVMMMM"
360 LPRINT SPC(10)*"MVMIVIVMIIIVMMMM"
370 LPRINT SPC(10)*"MVMIIIVMIIIIIVMMMM"
380 LPRINT SPC(9)*"MVMIIVIVMIIIIIVMM"
390 LPRINT SPC(9)*"MVMIIVIVMIIIIIVMMMM"
400 LPRINT SPC(9)*"MVMIVVIMIIIIIVMMMM"
410 LPRINT SPC(8)*"MVMVMVVIIMVIIIIIVMMMM"
420 LPRINT SPC(8)*"MVMVMVMVMVVIIVMMVMVM"
430 LPRINT SPC(8)*"MVMVMVMVMVVIIVMMVMVM"
440 LPRINT SPC(8)*"MVMVMVMVVIIMVVMVMVM"
450 LPRINT SPC(8)*"MVMIVV MVMIIIIIVMMVM"
460 LPRINT SPC(7)*"MVMIVV VVVIIMMVMVMVM"
470 LPRINT SPC(7)*"MVMIVV VVVIIMMVMVMVM"
480 LPRINT " MVMIVV VVVIIMMVMVMVM" L"
490 LPRINT " MVMIVV VVVIIMMVMVMVM" LL"
500 LPRINT " MVMIV VVVIIMMVMVMVM" LL"
510 LPRINT " MVMIV MVMIIIIIVMMVM" MLL"
520 LPRINT " MVMIV MVMIIIIIVMMVM" MLL"
530 LPRINT " MVMIV MVMIIIVVMVMVM" MML"
540 LPRINT " MVMIV MVMIIIVVMVMVM" MMLL"
550 LPRINT " MVMIVV MVMIVVMVM VVVVMVM" MLL"
560 LPRINT "MVMVIV MVMVMVM VVMVMVM"
570 LPRINT "MVMVMVVMVIVV VVMVM VM"
580 LPRINT " VM VMVVVV VM"
590 LPRINT SPC(10)*"MVMVV"
600 LPRINT SPC(12)*"MV"

```


UTILITAIRE

L'oric et l'alphabet

PRESENTATION

Lors de la création d'un programme, vous pouvez être amené à définir des caractères n'existant pas dans l'ensemble des caractères standard de **LORES 0** et **LORES 1**

ORIC permet la redéfinition de son jeu de caractères basse résolution.

Ce programme, un peu limité permet la redéfinition de tous les caractères **LORES 0** appartenant au clavier.

Un caractère est défini dans une grille 8 lignes, 6 colonnes. Chaque case de cette grille constitue un point élémentaire (ou pixel) qui peut être allumé ou éteint. Le caractère étant redéfini, il suffit de l'affecter à une lettre quelconque. Par la suite, cette lettre sera systématiquement remplacée par le caractère redéfini.

Pour revenir au jeu de caractères standard, deux solutions s'offrent à vous :

Débrancher ORIC puis le rebrancher.

Exécuter la routine d'initialisation des caractères **LORES X** en actionnant l'interrupteur **RESET** placé sous l'ORIC.

ORGANIGRAMME DE L'UTILITAIRE

Le programme a la structure A.

Le module principal consiste en une série de tests sur les touches du clavier.

- Les quatre touches de déplacement (droite, gauche, haut, bas) permettent de se déplacer sur la grille caractère.
- Les touches P et V servent (respectivement) à allumer (caractère plein) ou éteindre (caractère vide) le pixel courant.
- La touche "RETURN" fait prendre en compte le caractère redéfini. (voir organigramme page suivante)

LE PROGRAMME

Il se décompose en trois pavés :

Initialisation

Lignes 1000 à 1730	
Module de commande	Lignes 1000 à 1070
Présentation	Lignes 1100 à 1420
Affichage grille	Lignes 1500 à 1595
Init. curseur	Lignes 1600 à 1650

Traitement

Lignes 2000 à 2500	
Module de commande	Lignes 2000 à 2110
Déplacement ←	Lignes 2200 à 2250
Déplacement →	Lignes 2250 à 2290
Déplacement ↓	Lignes 2300 à 2340
Déplacement ↑	Lignes 2350 à 2390
Caract. plein	Lignes 2400 à 2440
Caract. vide	Lignes 2450 à 2500

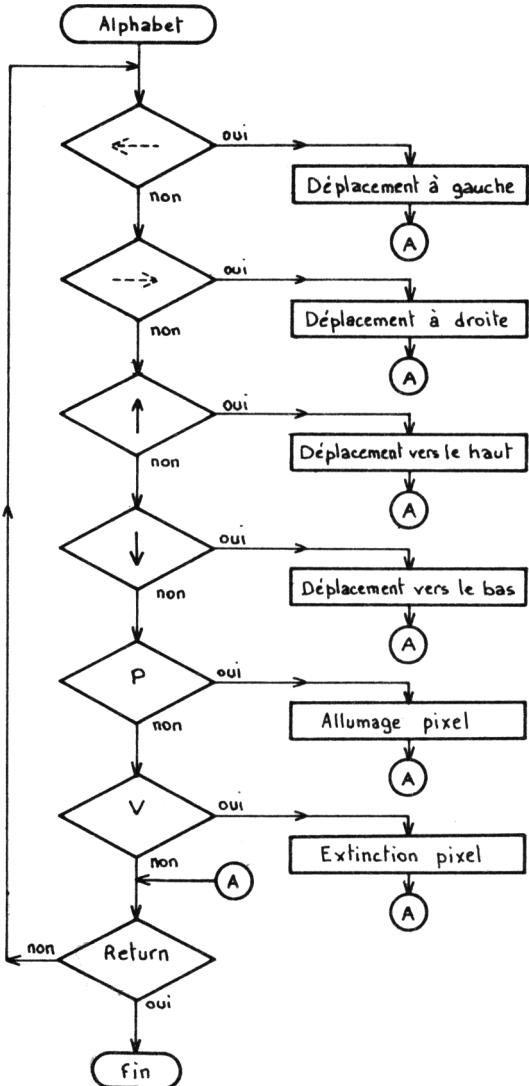
Modification caractère

Lignes 3000 à 3110
 Modification d'un des caractères clavier par le caractère que l'on vient de définir.

LES VARIABLES UTILISEES

A\$	Code couleur de ligne
B	Code ASCII du caractère à modifier
B\$	Caractère à modifier
BAS	Position verticale dans la grille caractère
DROITE	Position horizontale dans la grille caractère
I, J	Compteurs de boucles
T	Tableau image de la grille.

Le module principal a donc la structure suivante :



```

10 : REM ORIC et l'alphabet
20 :
30 : GOSUB 1000 'Initialisation
40 : GOSUB 2000 'Traitement
50 : GOSUB 3000 'Modification caractere
60 : END
70 :
80 : REM *****
1000 : REM Module d'initialisation et d'affichage
1010 :
1020 : GOSUB 1100 'Affichage menu
1030 : GOSUB 1500 'Affichage grille
1040 : GOSUB 1600 'Init. curseur
1050 : RETURN
1060 :
1070 : REM *****
1100 : REM Module d'affichage du menu
1110 :
1120 : PAPER 0:INK 5
1130 : REM Titre
1140 : PRINT CHR$(12)
1150 : PRINT CHR$(4);CHR$(27);"M";
1160 : PRINT CHR$(27);"A          ORIC ET L'ALPHABET";CHR$(4)
1170 : REM Objet
1180 : PRINT:PRINT:PRINT
1190 : A$="D" 'Code couleur = Bleu
1200 : PRINT"OBJET : ";
1210 : GOSUB 1700:PRINT"Modification des caracteres"
1220 : GOSUB 1700:PRINT"          standard de l'ORIC"
1230 : REM Job
1240 : PRINT:PRINT
1250 : A$="F" 'Code couleur = Cyan
1260 : PRINT"JOB  :";
1270 : GOSUB 1700:PRINT"Les caracteres sont definis"
1280 : GOSUB 1700:PRINT"          dans une grille 8 # 6"
1290 : GOSUB 1700:PRINT"          pixels . Chaque pixel peut"
1300 : GOSUB 1700:PRINT"          etre plein (Commande P)"
1310 : GOSUB 1700:PRINT"          ou vide (Commande V) ."
1320 : GOSUB 1700:PRINT"          Les deplacements se font"
1330 : GOSUB 1700:PRINT"          grace aux touches-fleches ."
1340 : REM Debut
1350 : PRINT:PRINT
1360 : A$="C" 'Code couleur = Jaune
1370 : PRINT"DEBUT :";
1380 : GOSUB 1700:PRINT"Pour commencer , appuyez sur"
1390 : GOSUB 1700:PRINT"          la touche 'RETURN'"
1400 : GET A$ 'Attente d'une action clavier
1410 : RETURN

```

```

1420 : REM *****
1500 : REM Module d'affichage grille-caractere et commentaires
1510 :
1520 : CLS 'Nettoyage d'ecran
1530 : FOR I=0 TO 5
1540 :   FOR J=0 TO 7
1550 :     PLOT 14+I,8+J, "."
1560 :   NEXT J
1570 : NEXT I
1580 : PLOT 0,22, "Quand vous aurez dessine le caractere,"
1585 : PLOT 0,23, "tappez RETURN ."
1590 : PRINT CHR$(27);"L";CHR$(27);"A      DEFINITION DE LA LETTRE
":RETURN
1595 : REM *****
1600 : REM Module d'initialisation curseur
1610 :
1620 : PLOT 14,8, "."
1630 : DROITE=0:BAS=0
1640 : RETURN
1650 : REM *****
1700 : REM Module de couleur de ligne
1710 : PRINT CHR$(27);"H";CHR$(27);A$;
1720 : RETURN
1730 : REM *****
2000 : REM Module de traitement
2010 :
2020 : GET A$
2030 : IF ASC(A$)=8 THEN GOSUB 2200
2040 : IF ASC(A$)=9 THEN GOSUB 2250
2050 : IF ASC(A$)=10 THEN GOSUB 2300
2060 : IF ASC(A$)=11 THEN GOSUB 2350
2070 : IF A$="P" THEN GOSUB 2400
2080 : IF A$="V" THEN GOSUB 2450
2090 : IF ASC(A$)=13 THEN RETURN
2100 : GOTO 2000

```

```

2110 : REM #####
2200 : REM Deplacement vers la gauche
2210 :
2220 : DROITE=DROITE-1
2230 : IF DROITE=-1 THEN DROITE=0
2240 : RETURN
2250 : REM Deplacement vers la gauche
2260 :
2270 : DROITE=DROITE+1
2280 : IF DROITE=6 THEN DROITE=5
2290 : RETURN
2300 : REM Deplacement vers le bas
2310 :
2320 : BAS=BAS+1
2330 : IF BAS=8 THEN BAS=7
2340 : RETURN
2350 : REM Deplacement vers le haut
2360 :
2370 : BAS=BAS-1
2380 : IF BAS=-1 THEN BAS=0
2390 : RETURN
2400 : REM Caractere plein
2410 :
2420 : T(BAS,DROITE)=1
2430 : PLOT 14+DROITE,8+BAS,"#"
2440 : RETURN
2450 : REM Caractere vide
2460 :
2470 : T(BAS,DROITE)=0
2480 : PLOT 14+DROITE,8+BAS, "."
2490 : RETURN
2500 : REM #####
3000 : REM Modification du caractere
3010 :
3020 : PRINT:PRINT:PRINT
3030 : INPUT"Caractere a modifier";B$
3040 : B=ASC(B$) : A=0
3050 : FOR I=0 TO 7
3060 :   FOR J=0 TO 5
3070 :     IF T(I,J)=1 THEN A=A+(2^(15-J))
3080 :   NEXT J
3090 :   POKE 46080+(B*8)+I,A : A=0
3100 : NEXT I
3110 : RETURN

```


Comptabilité oric 1 et atmos

Les 20 programmes proposés dans ce livre ont été développés sur ORIC-1. Ils sont compatibles ORIC ATMOS moyennant les modifications suivantes :

Attaque aérienne	Menu*
Casse-pipes	Menu* 2240 IF B<>3 THEN B\$="U":NP=NP+1 ELSE B\$="V" 2370 PRINT "Pressez une touche pour commencer"
Le piéton courageux	Menu*
Réflexe	Menu*
Bataille navale	2830 PLOT 2*A+9,B+9,"0" 2930 PLOT 2*A+9,B+9,"X"
Jack pot	Lignes 2240,2460,2560,2660 : supprimer un blanc dans les premières côtes
Jeu du 21	Rien à modifier
Jeu des allumettes	Rien à modifier
Les piranhas attaquent	Menu*
Mémoire à court terme	Menu*
Mastermind	Menu*
Figures de Moivre	Rien à modifier
Biorythmes	Rien à modifier
La vie	Menu*
Les unités anglo-saxonnes	Menu*
ORIC-piano	Rien à modifier
ORIC joue	Rien à modifier
Le chien	Rien à modifier
Modification des caractères	Rien à modifier

Menu : certaines lignes du menu de présentation devront comporter un nombre de "blancs" moins important ou des mots plus courts ...*

Imprimé en Belgique
sur les presses du Scorpion
D/1984/3548/3
N° d'édition 87109.8.1
ISBN 2.87109.008.4



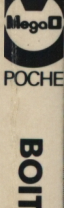
BOITE A OUTILS POUR ORIC ORIC 1 et ATMOS 20 PROGRAMMES EN BASIC

Qui n'a jamais regretté de ne pas avoir sous la main le tournevis ou la pince pour faire tel ou tel bricolage ? Les ordinateurs par leur aspect universel sont comme les boîtes à outils (et les auberges espagnoles...): on y trouve ce que l'on y met. Cette collection MegaO-Poche n'a pas la prétention de livrer le secret des grandes applications informatiques, mais bien de mettre à la disposition des utilisateurs des petits programmes tout prêts qui leur permettront de résoudre de nombreux problèmes de la vie quotidienne.

35,00 FF

ISBN 2.87109.008.4





POCHÉ

BOITES A OUILES POUR ORIC, ORIC 1 et ATMOS.

