

```

ier
8010 NO=5:POH=13:POE=13:POV=8:PO
X=8
8030 A#=GR$(1)+GR$(1)+GR$(1):
      B#=GR$(4)+GR$(4):
      C#=GR$(3)+G
R$(2):E#=N#+N#:D#=E#+N#
8040 CI1#=GR$(14)+GR$(15):
      CI2#=GR$(7)+GR$(7)+
GR$(7)+GR$(7):      CI3#=E#:CI4
#=E#+E#
8050 A1=1:A2=11:A3=1:A4=11:B1=11
:B2=21:      B3=11:B4=21:SC=0:SC
0=0-100*NIV
8060 LOCATE12,3:COLOR0,4:PRINT"B
onus":      LOCATE11,4:PRINT(SC
0+100):COLOR,0
8070 NOB=0:PA=0:NBD=0:LOCATE0,0,
0
8080 LOCATE 35,3:COLOR3:PRINT NO
8090 RETURN
8500 REM.....TITRE.....
8501 M1#="DODO#RERE#MIFASOSO#LAL
A#SI"
8502 M2#="SILA#LASO#SOFA#FAMIRE#
REDO#DO"      :M3#="04A1T4L8LALASI
SIMIMIFAF"
8503 M4#="04A2T8L16DOREMIFASOLAS
ILASOFAMIREDO":M7#="04A1T2L4DODO
"
8504 M5#="04A2T4L8MISOLARE"
8505 M6#="04T8L6A2SIRESIRESIDO"
8506 M8#="04A1T2L4SISI"
8510 C1#=GR$(16)+GR$(17):
      C2#=GR$(18)+GR$(19)+
GR$(20)+GR$(21)
8520 C3#=GR$(22)+GR$(23)+GR$(24)
+GR$(25)      :C4#=GR$(26)+GR$(27)+
GR$(28)+GR$(29)
8530 C5#=GR$(30)+GR$(31)+GR$(32)
+GR$(33)      :C6#=GR$(34)+GR$(35)+
GR$(36)+GR$(37):      C7#=GR$(38)+
GR$(39)+GR$(40)+GR$(41)
8540 FOR I=1 TO 3:C=(I-1)*9
8550 FOR J=1 TO 8
8560 L=(J-1)*5:K=K+1:IF K>5 THEN
K=1
8570 COLORK:LOCATEL+1,C:PRINTC1#
:LOCATEL,C+1:PRINTC2#:LOCATEL,C+
2:PRINTC3#:LOCATEL,C+3:PRINTC4#
:LOCATEL,C+4:PRINTC5#:LOCATEL,C+5
:PRINTC6#:LOCATEL,C+6:PRINTC7#;
8580 NEXT J,I
8590 LOCATE0,8:ATTRB1,1:COLOR1:
PRINT"# C H A N T I
E R #"
8600 COLOR7: ATTRB0,1:LOCATE0,17
:PRINT" CES CAMIONS ATTENDENT LE

```

```

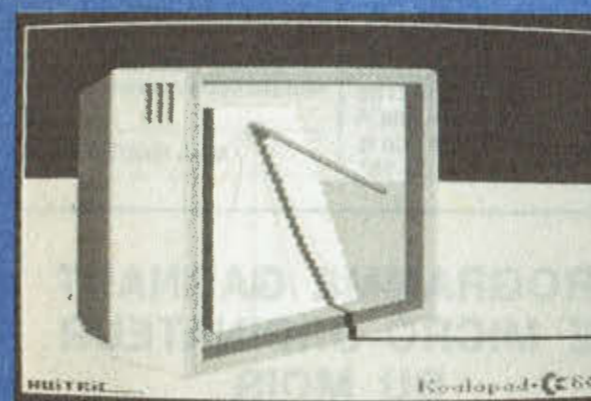
BETON LIQUIDE"
8610 PLAY"T1AOL12":FORI=1TO 5:PL
AY"03"+      M1#+"04"+M1#+"05"+
M1#+"04"+M2#      NEXTI:CLS:
ATTRB0,0:RETURN
9000 REM##### Fin du jeu ##
#####
9010 COLOR3:LOCATE35,3:PRINT NO
9020 PLAY"A305T8L16DODOREREMIMIF
AFASOSO"
9030 CLS:SCREEN0,7,5:LINE(0,0)-(
39,0)"#",1:COLOR0
9040 LOCATE1,2:PRINT"NB POINTS M
ARQUES="      ;SC;" Niveau=";NI
V:LOCATE1,4:      PRINT"NB OU
V.BLESSES =" ;NOB;"DONT";      NBD
;"CHAUFF."
9050 R=NOB-4:ON R GOTO 9200,9200
,9200
9090 IF NIV=9 THEN NIV=8
9100 LOCATE 1,6:PRINT"C'est bien
,il faut passer au niveau";NIV+1
:GOTO 9300
9200 LOCATE 0,6:PRINT " Il ne
faut pas abandonner....."
9300 IF NOB>NBD*2 THEN LOCATE0,8
: PRINT " Les ascenseurs vous po
sent des pro-      blemes.Pour pr
endre l'ascenseur,incli      -ner l
e manche quand l'ascenseur est
a une position avant celle de
la      passerelle ou se trouv
e l'ouvrier."      :GOTO9330
9310 IF NOB<=NBD*2 THEN LOCATE0,
8:PRINT " Il faut surveiller les
cuves et ne pas laisser accumu
ler le ciment.      Pour o
uvrir les trappes l'ouvrier doit
etre pres de la cuve a vider."
9320 PRINT" Il faut appuyer sur
le bouton ACTION."
9330 NFA=PA*10:IFNFA>SC/1.5 THEN
LOCATE1,      15:PRINT"La product
ion est faible:"      ;SC-NFA;"to
nnes":GOTO9400
9340 IF SC>1000THEN LOCATE1,15:F
RINT      "La production est
bonne:";SC-NFA;      "tonnes"
9400 LOCATE10,19:ATTRB0,1:PRINT
" F I N D U J
E U"
9410 LOCATE 9,22:COLOR1: PRINT
"ON REJOU E ==> (O/
N)";:ATTRB0,0
9420 LINE(0,24)-(39,24)"#",1
9430 R#=INPUT$(1)
9440 IF R#="0" THEN RUN
9450 END 'fin des instructions

```

micro7

LE
CAHIER
DU

n°20 LOGICIEL



CADRAN SOLAIRE

HELIUM
ATTAQUE D'UN OVNI
ANIMORIC
ATTACK

INITIATION
ATARIDEES

VENDREDI SHOW
PROMENADE DANS
LA MEMOIRE ECRAN

CHANTIER
CAHIER DES AS

EDITO

Pour nous envoyer vos programmes, enregistrez votre programme sur une cassette après avoir nettoyé les têtes de votre magnétophone. De l'alcool et un coton-tige suffisent. Essayez de relire la cassette au moins une fois (utiliser la fonction VERIFY pour ceux qui l'ont). Ecrivez votre nom, celui de la machine, celui du programme et le mode de chargement (ex : CLOAD « PROG », S) sur la cassette et sur le boîtier. Rédigez une lettre d'accompagnement comprenant : bien visible la machine sur laquelle le programme a été réalisé, le Basic utilisé (étendu, numéro de version ou normal, etc.). Donnez au moins un résumé de ce que fait le programme et ses particularités. Indiquez également les particularités de langage utilisées (caractères de contrôles spécifiques ...). Nous acceptons aussi les disquettes. Et n'oubliez pas d'envoyer le tout. A bientôt.

Avertissement : les imprimantes AZERTY ne possédant pas la fonte ascii américaine, certains signes utilisés pour la programmation n'apparaissent pas. Ils ont cependant un équivalent. C'est le cas pour l'arobas, remplacé par le caractère à et pour le signe dièse remplacé par le symbole £.

Basic : ZX
Difficulté : **
Adaptabilité : ***

sinclair

CADRAN SOLAIRE

Un cadran solaire, c'est plus chic qu'une pendule ! Pour étalonner le vôtre, voici l'indispensable programme. Il tourne sur ZX 81.

Ce programme qui fonctionne sur ZX81, avec une extension de mémoire de 16 Ko permet de réaliser un cadran solaire comme son nom l'indique. Il est assez simple et se passe d'explications. Analyse des différentes phases du programme :
1- La latitude du lieu (alpha) est introduite en degrés et minutes sexagésimaux ; puis convertie en radians dans la variable T ;
2- La déclinaison du soleil Y, qui est variable dans l'année mais qui passe par un maximum : 23 degrés 27 minutes, ou 23,45 degrés, ou encore 23,45(PI/180) radians au moment du solstice d'été, le 21 juin ;
3- L'angle beta du plan du méridien avec le plan passant par le soleil et l'axe des pôles ; à midi beta=0 ; dans le programme, l'angle beta est rangé dans la variable A de la boucle FORNEXT de la ligne 150, un angle de 15°

correspond à une heure et une demi-heure correspond à PI/24 radians ;
4- L'angle beta maximum rangé dans la variable B représente l'angle beta au coucher du soleil le jour du solstice d'été. Il est donné par la relation $\text{Cos}(\text{beta max}) = -\text{Tan}(\text{alpha}) \text{Tan}(23,45(\text{PI}/180))$ et est calculé à la ligne 110.
5- L'angle psi, rangé dans la variable X, permet de graduer correctement le cadran.
- lignes 180 à 220 pour le cadran horizontal : $\text{Tan}(\text{psi}) = \text{Sin}(\text{alpha}) \text{Tan}(\text{beta})$;
- lignes 220 à 250 pour le cadran vertical : $\text{Tan}(\text{psi}) = \text{Cos}(\text{alpha}) \text{Tan}(\text{beta})$.
- la ligne 290 convertit l'angle psi de radians en degrés en ajoutant 180° si l'angle est négatif.
Ne perdez pas une minute réalisez votre cadran solaire.

M.-J. GAUTHERON

PROGRAMME GAGNANT
LE MICRO-ORDINATEUR
DU MOIS

```
6630 DEFGR$(62)=0,0,128,192,192,
224,176,248
6640 DEFGR$(63)=10,15,26,31,58,6
3,106,255
6650 DEFGR$(64)=170,255,170,255,
170,255,170,255
6660 DEFGR$(65)=168,252,172,254,
170,255,171,255
6665 REM.....OUVRIER RECOUVERT..
.....
6670 RD1#=GR$(60)+GR$(61)+GR$(62
)
6680 RD2#=GR$(63)+GR$(64)+GR$(65
)
6705 REM.....CHUTE OUVRIER.....
.....
6710 O4#=GR$(54)+GR$(55)+GR$(56)
6720 O5#=GR$(57)+GR$(58)+GR$(59)
6760 RETURN
7000 REM====Mise en place du cha
ntier====
7005 REM.....CUVES.....
7010 FC#=GR$(5)+GR$(4)+GR$(4)+GR
$(4)+GR$(4)+GR$(6)
7020 FOR V=7 TO 12 STEP 5
7030 FOR H=4 TO 30 STEP 26
7040 LINE(H,V)-(H,V+2)GR$(2),2:L
OCATE H,V+3:COLOR2:PRINTF
C#;:LINE(H+5,V+2)-(H+5,V)GR
$(3),2
7050 NEXT H:NEXT V
7055 REM.....PASSERELLES.....
7060 DATA 10,11,10,16,23,11,23,1
6
7100 FOR I=1TO4:READ H(I),V(I)
7110 LINE(H(I),V(I))-(H(I)+6,V(I
))GR$(0),2:NEXT I
7115 REM.....PYLONES.....
7120 P1#=GR$(8)+GR$(9):P2#=GR$(1
0)+GR$(11)
7130 FOR H=11 TO 25 STEP 14
7140 FOR V=17 TO 23 STEP 2
7150 LOCATE H,V:COLOR2:PRINTF1#;
LOCATEH,V+1:PRINTF2
#;
7160 NEXT V:NEXT H
7195 REM.....DISTRIBUTEURS.....
....
7200 N#=CHR$(127):BOXF(12,0)-(16
,5)N#,4
7210 DATA 9,3,10,3,11,3
7220 BOXF(9,4)-(11,5)N#,4
7230 PSET(9,5)N#,0
7240 FORI=1TO3:READH(I),V(I)
7250 PSET(H(I),V(I))GR$(I+41),4:
NEXTI
7260 BOXF(23,0)-(27,5)N#,4:
BOXF(23,3)-(26,5)N#
```

```
,0
7270 BOXF(27,4)-(30,5)N#,4:
PSET(30,5)N#,0
7280 DATA 28,3,29,3,30,3
7290 FORI=1TO3:READH(I),V(I):
PSET(H(I),V(I))GR$(
I+44),4:NEXTI
7300 LINE(13,21)-(16,21)GR$(0),2
7310 LINE(23,6)-(29,6)GR$(0),2
7320 PSET(14,22)GR$(12),2:PSET(1
4,23)GR$(13),2
7330 PSET(15,23)GR$(12),2:PSET(1
5,24)GR$(13),2
7340 PSET(16,24)GR$(12),2
7345 REM.....OUVRIER SUR PASSER
ELLE...
7350 O1#=GR$(48)+GR$(49):
O2#=GR$(50)+GR$(51)
O3#=GR$(52)
+GR$(53)
7360 POH=13:POV=8:GOSUB230
7400 POH=2:POV=22:GOSUB230
7410 POH=36:POV=22:GOSUB230
7600 REM.....CAMIONS.....
7610 COLOR1:LOCATE6,18:PRINTC1#;
LOCATE5,19:PRINTC2#;LOCATE5,20:P
RINTC3#;LOCATE5,21:PRINTC4#;LOCA
TE5,22:PRINTC5#;LOCATE5,23:PRINT
C6#;LOCATE5,24:PRINTC7#;
7620 COLOR1:LOCATE32,18:PRINTC1#;
LOCATE31,19:PRINTC2#;LOCATE31,2
0:PRINTC3#;LOCATE31,21:PRINTC4#;
LOCATE31,22:PRINTC5#;LOCATE31,23
:PRINTC6#;LOCATE31,24:PRINTC7#;
7630 COLOR 1,3:PSET(5,23)GR$(34)
:PSET(8,23)GR$(37):PSET(31,23)GR
$(34):PSET(34,23)GR$(37):COLOR1,
0 'Phares allumes
7701 LOCATE 1,1:COLOR1:PRINT" Ni
veau": LOCATE 1,2:PRINT"(0
-->9)"
7702 LOCATE1,3:INPUTNI#:NIV=VAL(
NI#):IF NIV<0 OR NIV>9 TH
EN LOCATE 1,3:PRINT" "
:GOTO7702
7703 Y1=0.94-0.02*NIV:Z1=0.90-0.
02*NIV
7704 LOCATE1,5:PRINT" N=";NIV
7710 LOCATE 1,1:COLOR1:PRINT" Po
ints ": LOCATE1,2:PRINT"
"
7720 LOCATE 1,3:COLOR3:PRINT" 0
"
7730 LOCATE29,i:COLOR1:PRINT"Nb.
Ouvriers"
7740 BOX(0,0)-(319,199),1
7900 RETURN
8000 REM Init. Ascenseurs + Duvr
```

```

55,0,0,0
6030 DEFGR$(2)=7,7,7,7,7,7,7,7
6040 DEFGR$(3)=224,224,224,224,224,224,224,224
6050 DEFGR$(4)=255,255,255,0,0,0,0,0
6060 DEFGR$(5)=7,7,7,0,0,0,0,0
6070 DEFGR$(6)=224,224,224,0,0,0,0,0
6080 DEFGR$(7)=0,255,255,255,255,255,255,255
6090 DEFGR$(8)=255,255,224,224,224,224,224,224
6100 DEFGR$(9)=255,255,7,15,31,63,119,231
6110 DEFGR$(10)=225,227,231,238,252,248,240,255
6120 DEFGR$(11)=199,135,7,7,7,7,7,255
6130 DEFGR$(12)=240,240,48,48,63,63,3,3
6140 DEFGR$(13)=3,3,0,0,0,0,0,0
6150 DEFGR$(14)=31,63,127,255,255,255,252,248
6160 DEFGR$(15)=254,255,255,255,255,255,231,198
6170 DEFGR$(16)=0,0,0,0,0,63,31,15
6180 DEFGR$(17)=0,0,0,0,0,252,248,240
6190 DEFGR$(18)=0,0,0,3,7,15,31,31
6200 DEFGR$(19)=15,63,255,255,255,255,255,255
6210 DEFGR$(20)=240,252,255,255,255,255,255,255
6220 DEFGR$(21)=0,0,0,192,224,240,248,248
6230 DEFGR$(22)=63,63,127,127,255,255,255,0
6240 DEFGR$(23)=255,255,255,255,255,255,255,0
6250 DEFGR$(24)=255,255,255,255,255,255,255,0
6260 DEFGR$(25)=252,252,254,254,255,255,255,0
6270 DEFGR$(26)=31,255,224,192,200,200,200,204
6280 DEFGR$(27)=255,255,0,0,0,0,0,0
6290 DEFGR$(28)=255,255,0,0,32,32,32,48
6300 DEFGR$(29)=248,255,7,3,3,3,3,3
6310 DEFGR$(30)=202,201,200,192,224,255,255,128
6320 DEFGR$(31)=0,0,128,64,32,255,255,0

```

```

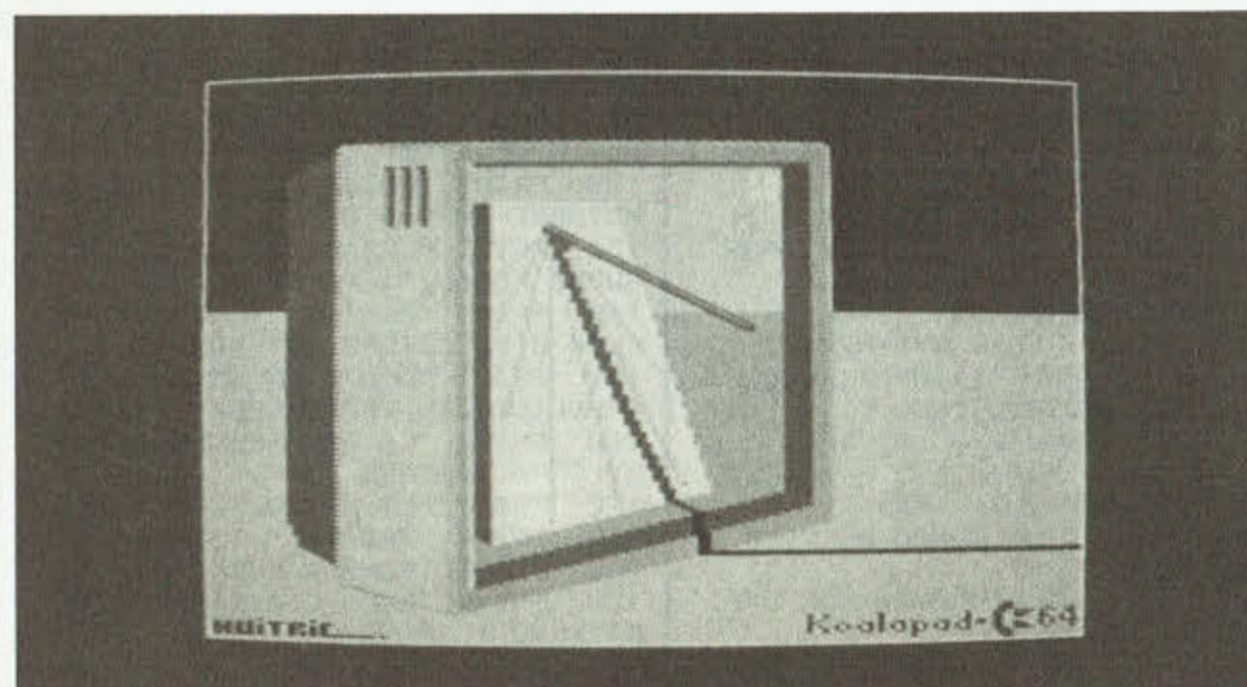
6330 DEFGR$(32)=40,36,34,1,0,255,255,0
6340 DEFGR$(33)=3,3,3,3,135,255,255,1
6350 DEFGR$(34)=255,255,192,192,192,192,255,255
6360 DEFGR$(35)=255,128,255,128,255,128,255,128
6370 DEFGR$(36)=255,1,255,1,255,1,255,1
6380 DEFGR$(37)=255,255,3,3,3,3,255,255
6390 DEFGR$(38)=255,255,78,114,78,114,78,114
6400 DEFGR$(39)=255,255,0,0,0,0,0,0
6410 DEFGR$(40)=255,255,0,0,0,0,0,0
6420 DEFGR$(41)=255,255,78,114,78,114,78,114
6430 DEFGR$(42)=0,0,0,0,0,0,7,63
6440 DEFGR$(43)=0,0,0,3,31,255,255,255
6450 DEFGR$(44)=1,15,127,255,255,255,255,255
6460 DEFGR$(45)=128,240,254,255,255,255,255,255
6470 DEFGR$(46)=0,0,0,192,248,255,255,255
6480 DEFGR$(47)=0,0,0,0,0,0,224,252
6490 DEFGR$(48)=0,7,15,48,31,17,27,31
6500 DEFGR$(49)=0,224,240,12,248,136,216,120
6510 DEFGR$(50)=30,12,15,123,251,251,203,203
6520 DEFGR$(51)=120,48,240,222,223,223,211,211
6530 DEFGR$(52)=207,207,15,14,28,28,120,120
6540 DEFGR$(53)=243,243,240,112,48,48,60,60
6550 DEFGR$(54)=0,0,0,0,16,31,43,105
6560 DEFGR$(55)=0,24,24,24,24,29,159,255
6570 DEFGR$(56)=3,3,7,15,30,60,248,243
6580 DEFGR$(57)=107,111,111,107,105,43,31,16
6590 DEFGR$(58)=191,48,54,182,246,246,150,31
6600 DEFGR$(59)=227,227,231,254,254,252,240,224
6610 DEFGR$(60)=0,0,0,0,1,1,3,7
6620 DEFGR$(61)=12,62,107,255,170,255,170,255

```

```

10 REM *CADRAN SOLAIRE *
20 PRINT "RENTREE DES DONNEES : "
30 PRINT "LATITUDE DU MILIEU EN DEGRES ET MINUTES SEXAGESIMAUX : "
40 INPUT "DEGRES : ";D
50 INPUT "MINUTES : ";M
60 INPUT "VILLE : ";A$
70 PRINT "TRAITEMENT :";
80 REM FAST
90 T=(D+M/60)*PI/180
100 W =ACS (-TAN(T)*TAN(23.45 *PI/180))
105 GOSUB 10000
110 H=12
120 CLS
130 PRINT "GRADUATION D'UN CADRAN SOLAIRE A LA LATITUDE DE ";A$
140 PRINT TAB(7);D;" DEGRES";
150 PRINT M;" MINUTES"
160 PRINT "HEURE HORIZONTAL VERTICAL"
170 FOR A=0 TO B STEP PI/24
180 X=SIN(T)
190 PRINT TAB(0);INT(H);"H";"30" AND H<>INT(H);TAB(7);
200 GOSUB 300
210 X=COS(T)
220 IF A>PI/2 THEN GOTO 250
230 PRINT TAB(20);
240 GOSUB 300
250 H=H+.5
260 NEXT A
270 REM COPY
280 SLOW
290 STOP
300 X=ATN(X*TAN(A))
310 X=(X+(X<0)*PI)*180/PI
320 A$=" "+STR$(INT(X))
330 B$="0"+STR$(INT((X-INT(X))*60))
340 PRINT A$(LEN(A$)-2 TO );"DG";B$(LEN(B$)-1 TO);"MN";
350 RETURN

```



Basic : Applesoft
Difficulté : **
Adaptabilité : **



HELIUM

Canarder des atomes d'Hélium avec un canon électronique c'est mieux que la bataille navale. Et on joue tout seul ! Un jeu de réflexions.

Bien que ce soit un classique du genre, ce jeu représente toujours un bon exercice de réflexion. L'esprit en est assez simple, vous avez sur l'écran un quadrillage dans lequel sont placés des atomes d'hélium. Il s'agit de trouver dans quelles cases ils se trouvent. Pour cela vous disposez d'un canon à électrons avec lequel vous envoyez un flux électronique sur les atomes. Si le flux est dirigé sur un atome, il est absorbé, s'il passe à côté d'un atome, il est dévié, sinon il avance tout droit. La structure du programme est assez claire mais nous don-

nons la liste des différentes variables et leurs utilisations :

- IT : Si IT=2, l'Apple dessine l'itinéraire du tir au fin de correction ou de modification. Si IT=0 en mode jeux, la trajectoire des électrons reste secrète.

- M(X,Y) : L'aire de jeux est assimilée à une matrice 12,12. Si M(X,Y)=0 la case est vide. Si M(X,Y)=1 on a toujours x=0 ou Y=0 ou X=11 ou Y=11, il s'agit d'une des cases du bord, elle est occupée par un numéro ou une cible et si M(X,Y)=2, la case est occupée par un atome.

- CO est le code du chiffre

utilisé

- N(CO,X,Y) est un élément de la matrice définissant un chiffre, si N(CO,X,Y)=1 on met un point en (Xo+X,Yo+Y) sinon. Il n'y a pas de point.

- HX,HY représente des coordonnées utilisables en HPLLOT

- X,Y sont les coordonnées d'une case dans la matrice 12,12

- D indique la direction du canon, ou du tir à un instant : D=1, vers le haut (touche I), D=2, vers la droite (touche K), D=3, vers le bas (touche M) et D=4, vers la gauche (touche J)

- DX,DY sont les coordon-

nées de départ du tir, position du canon.

- AX,AY sont les coordonnées d'arrivée du tir.

- X(I),Y(I) sont les coordonnées des six atomes à trouver.

- NA est le nombre d'atomes découverts.

- TX(I),TY(I) sont les coordonnées des atomes trouvées par le joueur.

Attention, les possesseurs d'Apple IIe doivent utiliser le clavier QWERTY pour plus de commodité. Nous vous souhaitons bonne chance, mais il est inutile de vous atomisez pour si peu. ■

Denis MENARD

```
10 LOMEM: 24576: DIM
N(10,7,11),M(12,12): HOME :
PRINT "PATIENCE !..."
20 RESTORE : FOR J = 1 TO 84:
READ A: NEXT J
30 FOR J = 10 TO 1 STEP - 1:
PRINT J: FOR YY = 1 TO 11: FOR
XX = 1 TO 7
40 READ A:N(J,XX,YY) = A
50 NEXT XX: NEXT YY: NEXT J
60 TEXT : HOME : INPUT "VOULEZ
VOUS DES INSTRUCTIONS !...":A#:
IF LEFT$(A$,1) = "N" THEN
HGR2 : GOTO 560
70 HOME : PRINT TAB(
2)"-->MUNI D'UN CANON A
ELECTRONS VOUS"
80 PRINT "ALLEZ JOUER SUR UNE
GRILLE (10*10) OU"
90 PRINT "SONT CACHES 6 ATOMES
D'HELIUM.": PRINT
```

```
100 PRINT TAB(2)"-->VOUS
DEVEZ DETERMINER LA POSITION"
110 PRINT "DES 6 ATOMES EN LE
MOINS DE TIRS"
120 PRINT "POSSIBLE; MAIS A LA
MOINDRE ERREUR DE"
130 PRINT "LOCALISATION VOUS
AUREZ PERDU.": PRINT
140 PRINT TAB(2)"-->SI LE
FLUX D'ELECTRONS EST DANS"
150 PRINT "L'AXE DU NOYAU
D'HELIUM, IL TRAVERSERA"
160 PRINT "LA ZONE ELECTRONIQUE
ET SERA ABSORBE PAR":
170 PRINT "LE NOYAU.": PRINT
180 PRINT TAB(2)"-->SI LE
FLUX D'ELECTRONS N'EST PAS"
190 PRINT "DANS L'AXE DU NOYAU
MAIS QU'IL PENETRE"
200 PRINT "LA ZONE ELECTRONIQUE
DE L'ATOME, IL"
```

```
1280 COLOR1:LOCATE32,15:PRINTC#:
RETURN
1290 REM*****TRAITEMENT DU CIME
NT*****
1410 IF SCREEN(6,16)=0 THEN LOCA
TE 6,16: COLOR 0:PRINTCI3#:
1415 IF SCREEN(32,16)=0 THEN LOC
ATE32,16 :COLOR 0:PRINTCI3#:
1420 IF SCREEN(5,14)=0 AND TR=2
THEN LOCATE 5,14:COLOR 0
:PRINTCI4#: LOCATE6,16:
COLOR7:PRINTCI1#;: SC=
SC+5:LOCATE1,3:COLOR3:PRINTSC
1425 IF SCREEN(31,14)=0 AND TR=4
THEN LOCATE31,14:COLOR 0
:PRINTCI4#: LOCATE32,16
:COLOR7:PRINTCI1#;: SC=
SC+5:LOCATE1,3:COLOR3:PRINTSC
1430 IFSCREEN(5,13)=0AND SCREEN(
6,14)<>0 THENLOCATE5,13:COLO
RO:PRINTCI4#: LOCATE 5,14
:COLOR7:PRINTCI2#:
1435 IFSCREEN(31,13)=0AND SCREEN
(31,14) <>0THENLOCATE31,13:
COLOR0:PRINTCI4# :LOCATE31,1
4:COLOR7:PRINTCI2#:
1440 IFSCREEN(5,12)=0AND SCREEN(
6,13)<>0 THENLOCATE5,12:COLO
RO:PRINTCI4#: LOCATE5,13:
COLOR7:PRINTCI2#:
1445 IFSCREEN(31,12)=0AND SCREEN
(31,13) <>0THENLOCATE31,12:
COLOR0:PRINTCI4# :LOCATE 31,
13:COLOR7:PRINTCI2#:
1450 IFSCREEN(6,11)=0AND SCREEN(
6,12)<>0 THENLOCATE6,11:COLO
RO:PRINTCI3#: LOCATE 5,12
:COLOR7:PRINTCI2#:
1452 IFSCREEN(6,11)=0AND SCREEN(
6,12)=0 THENCC=2:CL=12:GOTO
900
1455 IFSCREEN(32,11)=0AND SCREEN
(31,12) <>0THENLOCATE32,11:
COLOR0:PRINTCI3# :LOCATE31,1
2:COLOR7:PRINTCI2#:
1457 IFSCREEN(32,11)=0AND SCREEN
(31,12)= 0 THEN CC=36:CL=12:
GOTO900
1460 IF SCREEN(5,9)=0ANDTR=1 THE
N LOCATE 5,9:COLOR0:PRINTCI4
#:LOCATE 6,11: COLOR7:PRIN
TCI1#;:SC=SC+5:LOCATE1,3 :CO
LOR3:PRINT SC
1465 IF SCREEN(31,9)=0ANDTR=3THE
N LOCATE 31,9:COLOR0:PRINTCI
4#:LOCATE32,11: COLOR7:PRIN
TCI1#;:SC=SC+5:LOCATE1,3 :CO
LOR3:PRINT SC
```

```
1470 IFSCREEN(5,8)=0AND SCREEN(5
,9)<>0 THENLOCATE5,8:COLOR
0:PRINTCI4#: LOCATE5,9:C
OLOR7:PRINTCI2#:
1475 IFSCREEN(31,8)=0AND SCREEN(
31,9) <>0THENLOCATE31,8:C
OLOR0:PRINTCI4#: LOCATE31,9:
COLOR7:PRINTCI2#:
1480 IFSCREEN(5,7)=0AND SCREEN(5
,8)<>0 THEN LOCATE5,7:COLO
RO:PRINTCI4#: LOCATE5,8:C
OLOR7:PRINTCI2#:
1485 IFSCREEN(31,7)=0AND SCREEN(
31,8) <>0THENLOCATE31,7:C
OLOR0:PRINTCI4#: LOCATE31,8:
COLOR7:PRINTCI2#:
1490 IFSCREEN(6,5)=0AND SCREEN(5
,7)<>0 THENLOCATE6,5:COLOR
0:PRINTCI3#: LOCATE5,7:C
OLOR7:PRINTCI2#:
1492 IF SCREEN(6,5)=0ANDSCREEN(5
,7)=0 THEN CC=2:CL=8:GOTO
900
1495 IFSCREEN(32,5)=0AND SCREEN(
31,7) <>0THENLOCATE32,5:C
OLOR0:PRINTCI3# :LOCATE31,7:
COLOR7:PRINTCI2#:
1497 IF SCREEN(32,5)=0ANDSCREEN(
31,7)=0 THEN CC=36:CL=8:G
OTO900
1500 IF SCREEN(7,3)=0 THEN LOCAT
E 7,3: COLOR0:PRINTCI3#:LO
CATE 6,5:COLOR7 :PRINTCI1#:
1505 IF SCREEN(31,3)=0 THEN LOCA
TE 31,3: COLOR0:PRINTCI3#:LO
CATE 32,5:COLOR7 :PRINTCI1#:
1510 IF SCREEN(10,2)=0 THEN LOCA
TE 10,2: COLOR0:PRINTCI3#:LO
CATE 7, 3:COLOR7 :PRINTCI1#:
1515 IF SCREEN(28,2)=0 THEN LOCA
TE 28,2: COLOR0:PRINTCI3#:LO
CATE 31,3:COLOR7 :PRINTCI1#:
1600 Y=RND:IF Y>Y1 THEN LOCATE 1
0,2: COLOR7:PRINTCI1#;:P
LAY M7#
1605 Z=RND:IF Z>Z1 AND Z<.96 THE
N LOCATE 28,2:COLOR7:PRINTCI1#;:
PLAY M8#
1610 COLOR2:LOCATE6,10:PRINTB#:L
OCATE6, 15:PRINTB#:LOCATE32
,10:PRINTB#: LOCATE 32,1
5:PRINTB#
1700 RETURN
6000 REM=====CODES CARACTERES==
=====
6010 DEFGR$(0)=255,255,6,12,24,4
8,255, 255
6020 DEFGR$(1)=255,231,129,219,2
```

```

560 POH=POH-(POH=28)+(POH=10):PO
X=POV
570 RETURN
600 REM Deplac. Ouvrier sur Asc
610 D1=-4*(D=3)+4*(D=7):POE=POH
620 POH=POH+D1
630 POH=POH+3*(POV=18)*(POH=10)-
4*(POV= 18)*(POH=25)+4*(POV=
3)*(POH=13)-3*(POV=3)*(POH
=28)
640 POX=POV:GOSUB200:GOSUB230
650 POE=POH:RETURN
700 REM Accident sur Ascenseur
710 GOSUB200:EX=0
720 FOR I=POV TO 23 STEP 5
725 EX=EX+1: SCREEN,,EX
730 COLOR 3:IF I=23 THEN COLOR 1
740 LOCATE POH,I:PRINT 04$:LOCAT
E POH, I+1:PRINT 05$;:PLAY
M3$
750 COLOR 0:LOCATE POH,I:PRINT 0
4$: LOCATE POH,I+1:PRINT
05$;:NEXT I
760 SCREEN,,0:PLAY M4$:GOTO960
900 REM Debordement et chute du
ciment
910 EFC=-6*(CC=2)-32*(CC=36):
EFL=-5*(CL=8)-11*(CL
=12):COLOR0: LOCATE EFC,E
FL:PRINT CI3$
915 IF EFL=5THENLOCATE5,7:PRINTC
I4$: LOCATE5,8:PRINTCI4$:LOC
ATE5,9:PRINT CI4$:LOCATE31,7
:PRINTCI4$:LOCATE31,8 :PRINTC
I4$:LOCATE31,9:PRINTCI4$
920 EX=0:FORI=CL TO 20 STEP 4
925 EX=EX+1: SCREEN,,EX:LOCATE C
C,I
930 COLOR 7:PRINT CI1$:PLAY M5$:
LOCATE CC,I:COLOR 0:PRINT C
I3$:NEXT I
935 COLOR0:LOCATE CC,22:PRINTE$:
LOCATE CC,23:PRINTE$:LOCATE
CC,24:PRINTE$;
940 DD=CC+(CC=2):COLOR7:LOCATE D
D,23: PRINT RO1$:LOCATE DD
,24:PRINTRO2$;
945 PLAY M4$:NBD=NBD+1:COLOR0:LO
CATE DD, 23:PRINTD$:LOCATE DD
,24:PRINT D$;
950 COLOR3:LOCATE CC,22:PRINTD1$
:LOCATE CC,23:PRINTD2$:LOCATE
CC,24:PRINTD3$;
960 GOSUB200:COLOR0:GOSUB220:GOS
UB225
970 BOX(0,0)-(319,199),1:SCREEN,
,0:PLAY "L4AOT105"+M2$+"04"+
M2$+"03"+M2$

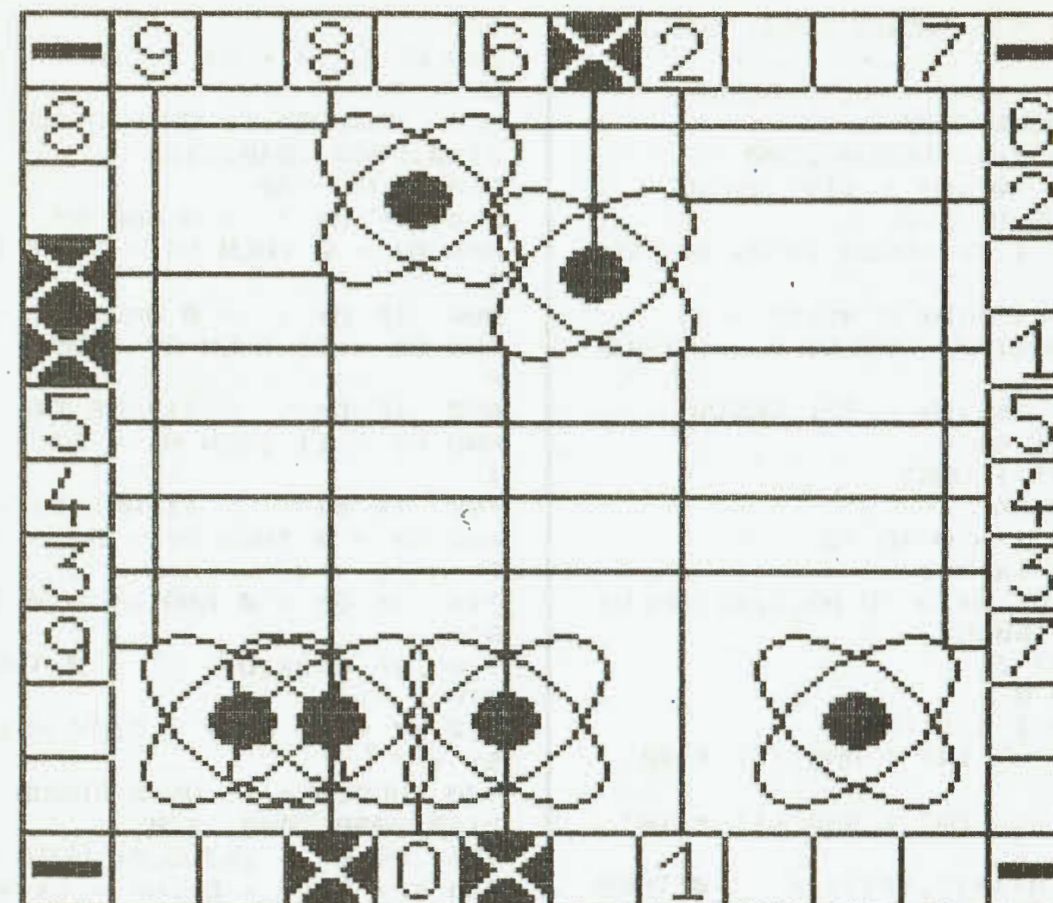
```

```

980 NOB=NOB+1:NO=NO-1:IF NO=0 TH
EN9000
985 LOCATE 35,3:COLOR 3:PRINT NO
990 A1=1:A2=11:A3=1:A4=11:B1=11:
B2=21
995 B3=11:B4=21:POH=13:POE=13:PO
V=8:POX= 8:ACC=1:GOSUB230
1000 REM .....Deplacements....
.....
1005 IF SC-SC0>=1000 THEN NO=NO+
1:SC0=SC -100*NIV-100*(NO>=8
):PLAY M6$:PLAY "L4AOT103"+
M1$+"04"+M1$+"05"+M1$: LOC
ATE35,3:COLOR3:PRINT NO
1010 COLOR0,4:LOCATE11,4:PRINTSC
0+1000: COLOR,0
1020 IF POH=17 OR POH=21 THEN GO
SUB 300 ELSE GOSUB 400
1030 D=STICK(0):IF D<>0 THEN GOS
UB 500
1040 GOSUB 1200
1050 D=STICK(0):IF D<> 0 AND (PO
H=17 OR POH=21)THEN GOSUB 6
00
1060 ACC=1-(POH=17)*(POV+3<>A1)-
(POH=17) *(POV+3<>A2)-(POH=2
1)*(POV+3<>B1)- (POH=21)*(P
OV+3<>B2)
1065 IF ACC<0 THEN GOTO 700
1070 A3=A1:A1=A1+5:IF A1>21 THEN
A1=1
1075 A4=A2:A2=A2+5:IF A2>21 THEN
A2=1
1080 B3=B1:B1=B1-5:IF B1<1 THEN
B1=21
1085 B4=B2:B2=B2-5:IF B2<1 THEN
B2=21
1090 IF A1=21 THEN B1=21
1095 IF B2=6 THEN A2=6
1096 IF B1=6 THEN B2=16
1100 GOTO 1005
1200 REM====Ouverture des porte
s====
1210 P=STRIG(0)
1220 TR=P*(POH= 9)*(POV= 8)*-1+
P*(POH= 9)*(POV=
13)*-2+ P*(POH=2
9)*(POV= 8)*-3+
P*(POH=29)*(POV=13)*-4
1240 ON TR GOSUB 1250,1260,1270,
1280
1245 GOTO 1290
1250 COLOR1:LOCATE 6,10:PRINTC$:
RETURN
1260 COLOR1:LOCATE 6,15:PRINTC$:
RETURN
1270 COLOR1:LOCATE32,10:PRINTC$:
RETURN

```

```
SPRINT CHR$(9);"G2E"
```



```

210 PRINT "SERA DEVIE DE 90
DEGRES.": PRINT
220 PRINT TAB( 2)"-->SI LE
FLUX N'EST PAS ABSORBE, LA"
230 PRINT "CASE PAR OU IL SORT
EST INDIQUEE.": PRINT
240 PRINT "$ -->": HTAB 2: VTAB
23: GET A$: HOME
250 PRINT TAB( 2)"-->QUAND LA
CASE EN HAUT A GAUCHE EST"
260 PRINT "PLEINE.": PRINT
270 PRINT TAB( 3)"1) VOUS
POUVEZ APPUYER SUR "; INVERSE :
PRINT "-->";: NORMAL : PRINT " OU
";: INVERSE : PRINT "<--": NORMAL
280 PRINT "POUR PLACER LE CANON
PUIS SUR ";: INVERSE : PRINT "
": NORMAL
290 PRINT "POUR TIRER.": PRINT
300 PRINT TAB( 3)"2) VOUS
POUVEZ APPUYER SUR ";: INVERSE :
PRINT "IJKM": NORMAL
310 PRINT "PUIS SUR ";: INVERSE
: PRINT " ";: NORMAL : PRINT "
POUR PLACER UN ATOME.": PRINT
320 PRINT TAB( 2)"SI VOTRE
LOCALISATION EST BONNE"
330 PRINT "LE JEU REPRENDR
JUSQU'A CE QUE VOUS"

```

```

340 PRINT "AYEZ LOCALISE LES 6
ATOMES SI NON LA"
350 PRINT "BONNE SOLUTION VOUS
SERA DONNEE.": PRINT
360 PRINT " (ETUDIEZ LE CAS
SUIVANT)": PRINT
370 PRINT "$ -->": HTAB 2: VTAB
18: GET A$
380 HGR2 : HCOLOR= 7: FOR I = 0
TO 12:HX = 14 * I: HPLT HX,0 TO
HX,168: HPLT 0,HX TO 168,HX:
NEXT I
390 HY = 1:HX = 31:CO = 1: GOSUB
1890:HX = 45:CO = 2: GOSUB
1890:HX = 59:CO = 3: GOSUB 1890
400 HX = 101:CO = 4: GOSUB
1890:HX = 115:CO = 5: GOSUB
1890:HX = 129:CO = 6: GOSUB
1890
410 HX = 3:HY = 29:CO = 6: GOSUB
1890:HX = 157:CO = 7: GOSUB
1890
420 HX = 147:HY = 49: GOSUB
2450: GOSUB 2530
430 HX = 3:HY = 71:CO = 3: GOSUB
1890:HX = 157:CO = 4: GOSUB
1890
440 HX = 77:HY = 91: GOSUB 2450:
GOSUB 2530

```

```

450 HX = 91: GOSUB 2450: GOSUB
2530
460 HX = 21:HY = 119: GOSUB
2450: GOSUB 2530
470 HX = 161: GOSUB 2200
480 HX = 105:HY = 119: GOSUB
2450: GOSUB 2530
490 HX = 133: GOSUB 2450: GOSUB
2530
500 HY = 155:HX = 45:CO = 2:
GOSUB 1890:HX = 59:CO = 1: GOSUB
1890
510 HY = 161:HX = 77: GOSUB
2200: GET A#
520 TEXT : HGR2
560 HCOLOR= 7
570 FOR I = 0 TO 12
580 HX = 14 * I
590 HPlot HX,0 TO HX,168: HPlot
0,HX TO 168,HX
600 NEXT I
610 CO = 0
620 FOR I = 1 TO 6
630 X(I) = INT ( RND (1) * 10)
+ 1
640 Y(I) = INT ( RND (1) * 10)
+ 1
650 IF M(X(I),Y(I)) < > 0 THEN
630
660 M(X(I),Y(I)) = 2
670 NEXT I
680 DX = 0:DY = 0:HX = 7:HY = 7
690 GOSUB 2450
700 GET A#
710 IF A# = "I" OR A# = "J" OR
A# = "K" OR A# = "M" THEN 1480
720 GOTO 750
730 GET A#
740 IF A# = " " THEN 960
750 HX = 14 * DX + 7:HY = 14 *
DY + 7
760 GOSUB 1840
770 IF ASC (A#) = 21 THEN 870
780 IF DY > = 0 AND DY < 11
AND DX = 0 THEN DY = DY + 1:D =
2
790 IF DX > = 0 AND DX < 11
AND DY = 11 THEN DX = DX + 1:D =
1
800 IF DY < = 11 AND DY > 0
AND DX = 11 THEN DY = DY - 1:D =
4
810 IF DX < = 11 AND DX > 0
AND DY = 0 THEN DX = DX - 1:D =
3
820 IF DX = 0 AND DY = 0 THEN
780
830 IF M(DX,DY) < > 0 THEN

```

```

780
840 HX = 14 * DX + 7:HY = 14 *
DY + 7
850 HCOLOR= 7: ON D GOSUB
2160,2080,2040,2120
860 GOTO 730
870 IF DX > = 0 AND DX < 11
AND DY = 0 THEN DX = DX + 1:D =
3
880 IF DY > = 0 AND DY < 11
AND DX = 11 THEN DY = DY + 1:D =
4
890 IF DX < = 11 AND DX > 0
AND DY = 11 THEN DX = DX - 1:D =
1
900 IF DY < = 11 AND DY > 0
AND DX = 0 THEN DY = DY - 1:D =
2
910 IF DX = 0 AND DY = 0 THEN
870
920 IF M(DX,DY) < > 0 THEN
870
930 HX = 14 * DX + 7:HY = 14 *
DY + 7
940 HCOLOR= 7: ON D GOSUB
2160,2080,2040,2120
950 POKE - 16368,0: GOTO 730
960 X = DX:Y = DY:HX = 14 * X +
7:HY = 14 * Y + 7: GOSUB 1840
970 M(DX,DY) = 1
980 AX = X:AY = Y
990 IF D = 1 THEN 1200
1000 IF D = 4 THEN 1140
1010 IF D = 2 THEN 1080
1020 IF M(X,Y + 1) = 2 THEN Y =
Y + 1: GOTO 1300
1030 X1 = X - 1:Y1 = Y + 1:X2 =
X + 1:Y2 = Y + 1
1040 IF M(X1,Y1) = 2 THEN D =
2
1050 IF M(X2,Y2) = 2 THEN D =
4
1060 IF M(X1,Y1) = 2 AND
M(X2,Y2) = 2 THEN D = 1
1070 GOTO 1250
1080 IF M(X + 1,Y) = 2 THEN X =
X + 1: GOTO 1300
1090 X1 = X + 1:Y1 = Y - 1:X2 =
X + 1:Y2 = Y + 1
1100 IF M(X1,Y1) = 2 THEN D =
3
1110 IF M(X2,Y2) = 2 THEN D =
1
1120 IF M(X1,Y1) = 2 AND
M(X2,Y2) = 2 THEN D = 4
1130 GOTO 1250
1140 IF M(X - 1,Y) = 2 THEN X =
X - 1: GOTO 1300

```



```

10 '=====  

===== 11 '=  

=
12 '= U S I N E   D E   C I M E  

N T =  

13 '= -----  

---- =  

14 '=  

=  

15 '= Sur T07+ 16ko + manettes d  

e jeu =  

16 '=====  

=====
17 '=  

=  

18 '= Auteurs===> E. et P. BERT  

RAND =  

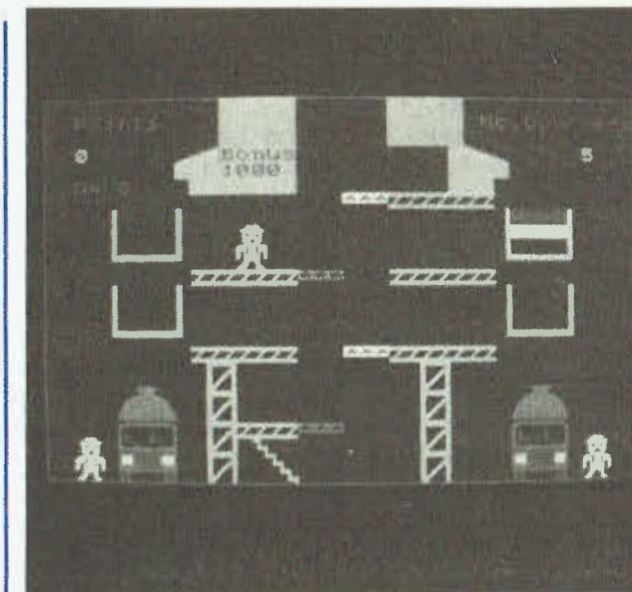
19 '=                               Mai  

1984 =  

20 '=====  

=====
50 SCREEN7,0,0:LOCATE0,0,0:CLS
60 CLEAR,,80:DIM H(30),V(30):DE
FINTA-X
70 GOSUB6000 'Caracteres specia
ux
80 GOSUB8500 'Titre
90 GOSUB7000 'Mise en place cha
ntier
100 GOSUB8000 'Initialisation As
censeur
110 GOTO 1000 'Debut du jeu
200 COLOR 0:LOCATE POE,POX:PRINT
01$: LOCATE POE,POX+1:PR
INTO2$:LOCATE POE,POX+2:
PRINTO3$:RETURN
210 COLOR 0:LOCATE 17,A3:PRINTA#
:LOCATE 17,A4:PRINTA#:LOCATE
20,B3:PRINTA#: LOCATE20,B4:

```



```

PRINTA#::RETURN
220 LOCATE 17,A1:PRINTA#:LOCATE
17,A2: PRINTA#::RETURN
225 LOCATE 20,B1:PRINTA#:LOCATE
20,B2: PRINTA#::RETURN
230 COLOR 3:LOCATE POH,POV:PRINT
01$: LOCATE POH,POV+1:PRI
NTO2$:LOCATE POH,POV+2:PR
INTO3$:RETURN
300 REM Deplac. Asc+Ouv sur Asc.
310 POE=POH:POX=POV
320 POV=POV-5*(POH=17)+5*(POH=21
)
325 POV=POV+5*(POV=23)-5*(POV=-2
)
330 GOSUB200:GOSUB210:COLOR1:GOS
UB220
340 COLOR5:GOSUB225:GOSUB230:POX
=POV: POE=POH
350 SC=SC+10:LOCATE1,3:COLOR 3:P
RINT SC
360 PA=PA+1:RETURN
400 REM Deplac. Ascenseurs vides
410 GOSUB210:COLOR1:GOSUB220
430 COLOR5:GOSUB225:RETURN
500 REM Deplac. Ouvrier sur Pass
erelle
510 D1=-4*(D=3)+4*(D=7):POE=POH
520 POE=POE+(POE>28)-(POE<10):PO
H=POH+D1
530 POH=POH+5*(POH=33)+(POH=29)
-5*(POH=5)-(P
OH=9)
540 POH=POH+3*(POV=18)*(POH=10)-
4*(POV= 18)*(POH=25)+4*(POV=
3)*(POH=13)-3*(POV=3)*(POH
=28)
550 GOSUB200:GOSUB230:POE=POH

```

```

,255,255:DEFGR$(36)=0,255,255,25
5,255,255,255,255:DEFGR$(37)=0,2
54,249,240,224,192,128,0:DEFGR$(
38)=128,128,128,129,128,128,128,
128:DEFGR$(39)=0,0,0,240,224,224
,224,1
1410 DEFGR$(40)=192,192,15,19,33
,64,128,0:DEFGR$(41)=0,0,254,255
,255,255,127,63:DEFGR$(42)=16,48
,120,124,124,252,252,252:DEFGR$(
43)=50,115,247,255,255,127,57,51
1450 DEFGR$(44)=24,56,248,252,25
2,254,190,24:DEFGR$(45)=255,254,
126,126,124,124,60,56:DEFGR$(46)
=140,140,128,128,140,140,140,128
:DEFGR$(47)=25,25,1,1,25,25,25,1
:DEFGR$(48)=36,61,126,127,254,25
0,112,64
1500 DEFGR$(49)=128,128,131,143,
143,159,159,159:DEFGR$(50)=2,28,
200,248,248,248,248,248:DEFGR$(5
1)=0,0,0,62,28,28,28,28
1530 DEFGR$(52)=31,14,8,8,8,8,8,
8:DEFGR$(53)=254,254,254,254,126
,124,124,124:DEFGR$(54)=255,0,0,
0,0,0,0,0:DEFGR$(55)=255,0,0,0,1
,7,31,127:DEFGR$(56)=241,7,31,12
7,255,255,255,255
1550 DEFGR$(54)=255,0,0,0,0,0,0,
0
1580 DEFGR$(57)=129,192,225,247,
255,255,255,255:DEFGR$(58)=0,0,1
28,192,224,240,248,252:DEFGR$(59
)=159,159,158,156,152,144,144,24
0:DEFGR$(60)=200,8,8,8,8,8,11,9
1620 DEFGR$(61)=0,0,0,0,0,0,240,
224:DEFGR$(62)=8,8,8,8,8,8,8,8:D
EFGR$(63)=120,56,56,48,6,63,127,
127:DEFGR$(64)=3,12,48,192,64,64
,76,79
1660 DEFGR$(65)=255,127,63,31,15
,7,3,1:DEFGR$(66)=255,255,255,25
5,255,255,254,204:DEFGR$(67)=254
,255,255,254,252,228,132,4:DEFGR
$(68)=9,9,9,9,57,97,64,192
1700 DEFGR$(69)=224,224,224,224,
224,224,0,0:DEFGR$(70)=50,119,12
7,255,255,127,126,62:DEFGR$(71)=
255,127,127,127,127,63,31,15:DEF
GR$(72)=79,79,79,79,67,64,64,64
1740 DEFGR$(73)=255,252,176,128,
128,134,158,158:DEFGR$(74)=34,34
,34,42,42,42,34,34:DEFGR$(75)=20
,52,52,52,36,4,4,4:DEFGR$(76)=1,
1,1,1,2,2,2,2
1780 DEFGR$(77)=64,64,64,64,64,6
4,64,64:DEFGR$(78)=48,60,60,60,6
0,60,12,0:DEFGR$(79)=158,158,158

```

```

,152,128,128,128,224:DEFGR$(80)=
34,34,34,34,34,34,34,50:DEFGR$(8
1)=4,20,52,52,52,36,12,48
1830 DEFGR$(82)=2,1,1,1,1,1,1,1:
DEFGR$(83)=0,0,0,128,64,96,24,6:
DEFGR$(84)=24,121,251,255,207,20
5,228,241:DEFGR$(85)=94,11,239,2
06,254,251,251,251
1870 DEFGR$(86)=60,7,0,0,7,15,31
,31:DEFGR$(87)=10,135,240,28,7,1
28,192,224:DEFGR$(88)=192,0,0,0,
192,64,127,1:DEFGR$(89)=1,1,1,1,
0,0,0,128:DEFGR$(90)=1,0,0,0,128
,128,64,56
1920 DEFGR$(91)=72,252,11,207,10
2,60,112,192:DEFGR$(92)=0,0,128,
112,14,1,0,0:DEFGR$(93)=0,0,0,0,
0,192,48,12:DEFGR$(94)=95,123,25
5,255,127,127,15,12
1960 DEFGR$(95)=251,238,238,254,
255,222,144,16:DEFGR$(96)=31,31,
31,31,31,31,31:DEFGR$(97)=224
,224,224,224,224,224,224:DEF
GR$(98)=90,255,52,111,228,206,12
4,120:DEFGR$(99)=192,64,96,48,24
,12,6,3
2010 DEFGR$(100)=6,1,0,0,0,0,0,1
28:DEFGR$(101)=0,192,56,6,1,0,0,
0:DEFGR$(102)=0,0,0,0,128,120,7,
0:DEFGR$(103)=3,0,0,0,0,0,128,12
0
3010 DATA1,2,4,2,7,7,7,0,0,0,0,1
2,15,14
3020 DATA15,17,2,0,1,2,3,4,13,0,
25,15,28,29,31,2,4,0,1,5,
6,26,0,39,15
3050 FORI=1TO3:READC1,C2,C3,L,N,
R,V,J,B,CH,LH,CB,LB:BOXF(CH,LH)-
(CB,LB)"",N,B:GOSUB120:NEXTI
3090 DATA 14,6,11,12,9,10,7,10,6
,10,5,9,4,9,3,9,3,8,2,9,2,8,2,7,
1,8,1,7,1
3091 DATA 7,1,7,1,6,1,6,1,6,1,6,
2,5,2,5,3,4,3,4,4,3,4,3,5,3,6,2,
7,1
3100 FORI=1TO29:READL1,L2:LINE(L
1,I)-(L1+L2,I),7:PLAY"AOT1L604DO
MI":NEXTI
3140 DATA 4,7,9,10,11,12,13,14,1
4,15,15,16,16,16:DIMS(14):L=22*8
3160 FORI=1TO14:READS(I):FORJ=1T
OI:LINE(L-S(I-J+1),J)-(L+S(I-J+1
),J),3:NEXTJ:PLAY"FAFA#":NEXT I
3210 LOCATE8,18:ATTRBO,1:COLOR0,
5:PRINT"* PREPAREZ LES OUTILS *"
:LOCATE2,21:PRINT"LE CHANTIER ES
T DERRIERE LE VILLAGE";
3220 ATTRBO,0:RUN"CHANTIER": END

```

```

1150 X1 = X - 1:Y1 = Y - 1:X2 =
X - 1:Y2 = Y + 1
1160 IF M(X1,Y1) = 2 THEN D =
3
1170 IF M(X2,Y2) = 2 THEN D =
1
1180 IF M(X1,Y1) = 2 AND
M(X2,Y2) = 2 THEN D = 2
1190 GOTO 1250
1200 IF M(X,Y - 1) = 2 THEN Y =
Y - 1:GOTO 1300
1210 X1 = X + 1:Y1 = Y - 1:X2 =
X - 1:Y2 = Y - 1
1220 IF M(X1,Y1) = 2 THEN D =
4
1230 IF M(X2,Y2) = 2 THEN D =
2
1240 IF M(X1,Y1) = 2 AND
M(X2,Y2) = 2 THEN D = 3
1250 IF M(X1,Y1) = 2 OR
M(X2,Y2) = 2 THEN 2020
1260 IF D = 1 THEN Y = Y - 1:
GOTO 1300
1270 IF D = 4 THEN X = X - 1:
GOTO 1300
1280 IF D = 2 THEN X = X + 1:
GOTO 1300
1290 Y = Y + 1
1300 BX = X:BY = Y
1310 GOSUB 2620
1320 IF IT = 2 THEN HCOLOR= 7:
HPLLOT 14 * AX + 7,14 * AY + 7 TO
14 * BX + 7,14 * BY + 7
1330 IF M(X,Y) = 2 THEN 1360
1340 IF X = 0 OR X = 11 OR Y =
0 OR Y = 11 THEN M(X,Y) = 1:
GOTO 1400
1350 GOTO 980
1360 IF IT = 2 THEN 1680
1370 HX = 14 * DX + 7:HY = 14 *
DY + 7
1380 GOSUB 2200
1390 GOTO 680
1400 IF IT = 2 THEN M(X,Y) = 0:
GOTO 1680
1410 IF CO = 10 THEN M(DX,DY) =
0:M(X,Y) = 0:GOTO 1600
1420 CO = CO + 1
1430 HX = 14 * DX + 3:HY = 14 *
DY + 1
1440 GOSUB 1890
1450 HX = 14 * X + 3:HY = 14 * Y
+ 1
1460 GOSUB 1890
1470 GOTO 680
1480 HX = 7:HY = 7:GOSUB 1840:X
= 1:Y = 1:HX = 21:HY = 21:GOSUB
2450

```

```

1490 GET A#
1500 HX = 14 * X + 7:HY = 14 * Y
+ 7:GOSUB 1840
1510 IF A# = "I" THEN Y = Y -
1:IF Y = 0 THEN Y = 10
1520 IF A# = "J" THEN X = X -
1:IF X = 0 THEN X = 10
1530 IF A# = "K" THEN X = X +
1:IF X = 11 THEN X = 1
1540 IF A# = "M" THEN Y = Y +
1:IF Y = 11 THEN Y = 1
1550 IF A# = " " THEN 1580
1560 HX = 14 * X + 7:HY = 14 * Y
+ 7:GOSUB 2450
1570 GOTO 1490
1580 IF M(X,Y) = 2 THEN NA = NA
+ 1:GOTO 1750
1600 GOSUB 2270
1610 GOSUB 1950
1620 FOR I = 1 TO 6
1630 HX = 14 * X(I) + 7:HY = 14
* Y(I) + 7
1640 GOSUB 2450
1650 GOSUB 2530
1660 NEXT I
1670 DX = 0:DY = 0:IT = 2
1680 IF DX >= 0 AND DX < 11
AND DY = 0 THEN DX = DX + 1:D =
3:GOTO 1730
1690 IF DY >= 0 AND DY < 11
AND DX = 11 THEN DY = DY + 1:D =
4:GOTO 1730
1700 IF DX <= 11 AND DX > 0
AND DY = 11 THEN DX = DX - 1:D =
1:GOTO 1730
1710 DY = DY - 1:D = 2
1720 IF DX = 0 AND DY = 0 THEN
2870
1730 IF M(DX,DY) = 0 THEN 1680
1740 X = DX:Y = DY:GOTO 980
1750 TX(NA) = X:TY(NA) = Y
1760 IF NA = 6 THEN GOSUB
2360:GOSUB 2270:GOSUB 1950
1770 FOR I = 1 TO NA
1780 HX = 14 * TX(I) + 7:HY = 14
* TY(I) + 7
1790 GOSUB 2450
1800 GOSUB 2530
1810 NEXT I
1820 IF NA = 6 THEN 1670
1830 GOTO 680
1840 HCOLOR= 0
1850 FOR J = - 6 TO 6
1860 HPLLOT HX + J,HY - 6 TO HX
+ J,HY + 6
1870 NEXT J
1880 RETURN
1890 HCOLOR= 7

```

```

1900 FOR YY = 1 TO 11: FOR XX =
1 TO 7
1910 IF N(CO,XX,YY) = 0 THEN
1930
1920 H PLOT HX + XX, HY + YY
1930 NEXT XX: NEXT YY
1940 RETURN
1950 HCOLOR= 0
1960 FOR I = 2 TO 10
1970 HX = 14 * I
1980 H PLOT HX, 14 TO HX, 154
1990 H PLOT 14, HX TO 154, HX
2000 NEXT I
2010 RETURN
2020 IF X = 0 OR Y = 0 OR X =
11 OR Y = 11 THEN 1400
2030 GOTO 1260
2040 HCOLOR= 7
2050 FOR J = - 6 TO 6
2060 H PLOT HX + J, HY - 6 TO
HX, HY + 6
2070 NEXT J: RETURN
2080 HCOLOR= 7
2090 FOR J = - 6 TO 6
2100 H PLOT HX - 6, HY + J TO HX
+ 6, HY
2110 NEXT J: RETURN
2120 HCOLOR= 7
2130 FOR J = - 6 TO 6
2140 H PLOT HX - 6, HY TO HX +
6, HY + J
2150 NEXT J: RETURN
2160 HCOLOR= 7
2170 FOR J = - 6 TO 6
2180 H PLOT HX, HY - 6 TO HX +
J, HY + 6
2190 NEXT J: RETURN
2200 HCOLOR= 7
2210 FOR J = - 4 TO 4
2220 H PLOT HX - 2, HY TO HX -
6, HY + J
2230 H PLOT HX + 2, HY TO HX +
6, HY + J
2240 H PLOT HX, HY - 2 TO HX +
J, HY - 6
2250 H PLOT HX, HY + 2 TO HX +
J, HY + 6
2260 NEXT J: RETURN
2270 HCOLOR= 7: HX = 7: HY = 7:
GOSUB 2320
2280 HX = 161: HY = 7: GOSUB
2320
2290 HX = 161: HY = 161: GOSUB
2320
2300 HX = 7: HY = 161: GOSUB
2320
2310 RETURN
2320 H PLOT HX - 5, HY - 1 TO HX

```

```

+ 5, HY - 1
2330 H PLOT HX - 5, HY TO HX +
5, HY
2340 H PLOT HX - 5, HY + 1 TO HX
+ 5, HY + 1
2350 RETURN
2360 HCOLOR= 7: HX = 7: HY = 7:
GOSUB 2410
2370 HX = 161: HY = 7: GOSUB
2410
2380 HX = 161: HY = 161: GOSUB
2410
2390 HX = 7: HY = 161: GOSUB
2410
2400 RETURN
2410 H PLOT HX - 1, HY - 5 TO HX
- 1, HY + 5
2420 H PLOT HX, HY - 5 TO HX, HY +
5
2430 H PLOT HX + 1, HY - 5 TO HX
+ 1, HY + 5
2440 RETURN
2450 HCOLOR= 7
2460 Z(1) = 0: Z(2) = 1: Z(3) =
2: Z(4) = 3: Z(5) = 4: Z(6) =
4: Z(7) = 5: Z(8) = 5
2470 FOR J = 1 TO 8
2480 H PLOT HX, HY TO HX +
Z(J), HY + Z(9 - J)
2490 H PLOT HX, HY TO HX +
Z(J), HY - Z(9 - J)
2500 H PLOT HX, HY TO HX -
Z(J), HY + Z(9 - J)
2510 H PLOT HX, HY TO HX -
Z(J), HY - Z(9 - J)
2520 NEXT J: RETURN
2530 RESTORE : HCOLOR= 7
2540 FOR J = 1 TO 42
2550 READ A: READ B
2555 IF HX + A > 154 OR HY - B
> 154 THEN 2565
2560 H PLOT HX + A, HY - B
2565 IF HX - A < 14 OR HY + B <
14 THEN 2575
2570 H PLOT HX - A, HY + B
2575 IF HX - A < 14 OR HY - B >
154 THEN 2585
2580 H PLOT HX - A, HY - B
2585 IF HX + A > 154 OR HY + B
< 14 THEN 2600
2590 H PLOT HX + A, HY + B
2600 NEXT J
2610 RETURN
2620 IF AX = 0 THEN AX = 0.5
2630 IF AY = 0 THEN AY = 0.5
2640 IF BX = 0 THEN BX = 0.5
2650 IF BY = 0 THEN BY = 0.5
2660 IF AX = 11 THEN AX = 10.5

```

```

(73)+GR$(74)+GR$(75)+GR$(76)::LO
CATEC2, L+11: COLORN, B: PRINTGR$(77
)+GR$(78)+GR$(79)+GR$(80)+GR$(81
)+GR$(82)+GR$(83)::LOCATEC2, L+12
: COLORV, B: PRINTGR$(84)+GR$(85):
242 COLORN, B: PRINTGR$(86)+GR$(87
)+GR$(88)+GR$(89)+GR$(90)::COLOR
V, B: PRINTGR$(91)::COLORN, B: PRINT
GR$(92)+GR$(93)::LOCATEC2, L+13: C
OLORV, B: PRINTGR$(94)+GR$(95):
252 COLORN, B: PRINTGR$(96)+GR$(97
)::COLORV, B: PRINTGR$(98)::COLORN
, B: PRINTGR$(99)+GR$(100)+GR$(101
)+GR$(102)+GR$(103):
260 RETURN
1010 V$=" ": F$=CHR$(127): DEFGR$(
0)=0, 0, 0, 0, 0, 0, 0, 3: DEFGR$(1)=0, 0
, 0, 16, 56, 124, 254, 255: DEFGR$(2)=0
, 0, 0, 0, 0, 0, 0, 128
1040 DEFGR$(3)=1, 1, 1, 1, 1, 1, 1, 1: D
EFGR$(4)=255, 1, 1, 57, 57, 57, 57, 1: D
EFGR$(5)=0, 0, 0, 0, 0, 0, 63, 127: DEFGR
R$(6)=0, 0, 0, 0, 0, 0, 255, 255: DEFGR$(
7)=0, 0, 0, 0, 0, 0, 224, 16
1090 DEFGR$(8)=1, 3, 7, 15, 31, 63, 12
7, 255: DEFGR$(9)=254, 252, 249, 243,
231, 207, 159, 63: DEFGR$(10)=254, 25
2, 248, 240, 224, 192, 128, 0
1120 DEFGR$(11)=8, 4, 2, 1, 3, 7, 15, 6
3: DEFGR$(12)=0, 0, 0, 255, 255, 255, 2
55, 255: DEFGR$(13)=0, 0, 0, 254, 249,
240, 224, 192: DEFGR$(14)=0, 0, 0, 0, 0
, 1, 3, 15
1160 DEFGR$(15)=0, 0, 0, 254, 254, 25
4, 254, 254: DEFGR$(16)=128, 145, 145
, 145, 145, 128, 128, 128: DEFGR$(17)=
32, 32, 38, 38, 38, 38, 32, 32
1190 DEFGR$(18)=0, 0, 12, 12, 12, 12,
0, 0: DEFGR$(19)=1, 1, 25, 25, 25, 25, 1
, 1: DEFGR$(20)=19, 19, 16, 16, 16, 16,
16, 16: DEFGR$(21)=12, 12, 0, 15, 19, 3
3, 64, 128
1230 DEFGR$(22)=128, 128, 128, 248,
252, 254, 255, 127: DEFGR$(23)=0, 0, 0
, 3, 5, 8, 16, 32: DEFGR$(24)=4, 4, 4, 25
2, 252, 252, 124, 60
1260 DEFGR$(25)=0, 0, 8, 8, 12, 28, 60
, 62: DEFGR$(26)=0, 0, 12, 12, 12, 12, 0
, 0: DEFGR$(27)=0, 0, 0, 0, 3, 15, 63, 25
5: DEFGR$(28)=3, 15, 63, 255, 255, 255
, 255, 255: DEFGR$(30)=255, 255, 255,
255, 255, 252, 240, 224
1320 DEFGR$(31)=255, 252, 240, 192,
0, 0, 0, 0: DEFGR$(32)=16, 16, 16, 16, 1
6, 16, 16, 16: DEFGR$(33)=32, 32, 32, 3
2, 32, 32, 32, 32: DEFGR$(34)=62, 127,
127, 127, 127, 255, 255, 255
1360 DEFGR$(35)=0, 7, 15, 31, 63, 127

```

VINCENNES

100 M. DU R.E.R

le spécialiste
de l'ordinateur
à moins de
5 000 francs



SINCLAIR	ORIC
LASER	COMMODORE
TEXAS	THOMSON
MEMOTECH	COLECO

et des logiciels

500 titres disponibles
en boutique, par correspondance

entre autre pour ORIC1/ATMOS

AIGLE D'OR (aventure)	180 F
ZORGON (arcade)	120 F
MONOPOLIC (reflexion)	160 F
DRIVER (action)	120 F
TRAITEMENT DE TEXTE	120 F
MISSION DELTA (simulation)	95 F



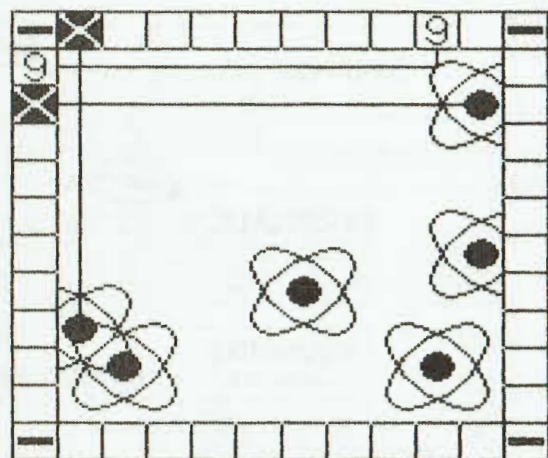
20, rue de Montreuil
94300 VINCENNES-Tel : (1) 328 22 06


```

2670 IF AY = 11 THEN AY = 10.5
2680 IF BX = 11 THEN BX = 10.5
2690 IF BY = 11 THEN BY = 10.5
2700 RETURN
2710 DATA
-6,-8,-5,-9,-4,-10,-3,-11,-2,-11
,-1,-12,0
2720 DATA
-12,1,-13,2,-13,3,-14,4,-14,5,-1
5,6,-15,7,-15
2730 DATA
8,-16,9,-16,10,-16,11,-16,12,-15
,13,-15,14
2740 DATA
-14,15,-13,15,-12,16,-11,16,-10,
16,-9,16,-8,15,-7
2750 DATA
15,-6,15,-5,14,-4,14,-3,13,-2,13
,-1,12
2760 DATA
0,12,1,11,2,11,3,10,4,9,5,8,6,7,
7
2770 DATA
0,0,1,1,1,0,0,0,1,0,0,0,1,0,1,0,
0,0,0,0,1,1,0,0,0,0,0,1,1,0,0,0,
0,0,1,1,0,0,0,0,0,1,1,0,0,0,0,0,
1,1,0,0,0,0,0,1,1,0,0,0,0,0,1,0,
1,0,0,0,1,0,0,0,1,1,1,0,0
2780 DATA
0,0,0,0,1,0,0,0,0,0,1,1,0,0,0,0,
1,0,1,0,0,0,1,0,0,1,0,0,0,0,0,

```

SPRINT CHR\$(9); "G2"



```

1,0,0,0,0,0,0,1,0,0,0,0,0,0,1,0,
0,0,0,0,0,1,0,0,0,0,0,0,1,0,0,0,
0,0,0,1,0,0,0,0,1,1,1,1,1
2790 DATA
0,0,1,1,1,0,0,0,1,0,0,0,1,0,1,0,
0,0,0,0,1,1,0,0,0,0,0,1,0,0,0,0,
0,0,1,0,0,0,0,0,1,0,0,0,0,0,1,0,
0,0,0,0,1,0,0,0,0,0,1,0,0,0,0,0,
1,0,0,0,0,0,1,1,1,1,1,1,1,1
2800 DATA
1,1,1,1,1,1,1,0,0,0,0,0,1,0,0,0,
0,0,1,0,0,0,0,0,1,0,0,0,0,0,1,1,
1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,
1,0,0,0,0,0,0,1,1,0,0,0,0,0,1,0,
1,0,0,0,1,0,0,0,1,1,1,0,0
2810 DATA
0,0,0,1,0,0,0,0,0,0,1,0,0,0,0,0,
1,0,0,0,0,0,0,1,0,0,0,0,0,1,0,0,
0,0,0,0,1,0,0,0,0,0,1,0,0,0,1,0,
0,1,1,1,1,1,1,1,0,0,0,0,1,0,0,0,
0,0,0,1,0,0,0,0,0,0,1,0,0
2820 DATA
1,1,1,1,1,1,1,1,0,0,0,0,0,0,1,0,
0,0,0,0,0,1,0,0,0,0,0,0,0,1,1,1,
1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,
1,0,0,0,0,0,0,1,1,0,0,0,0,0,1,0,
1,0,0,0,1,0,0,0,1,1,1,0,0
2830 DATA
0,0,1,1,1,0,0,0,1,0,0,0,1,0,1,0,
0,0,0,0,1,1,0,0,0,0,0,0,1,0,1,1,
1,0,0,1,1,0,0,0,1,0,1,0,0,0,0,0,
1,1,0,0,0,0,0,1,1,0,0,0,0,0,1,0,
1,0,0,0,1,0,0,0,1,1,1,0,0
2840 DATA
1,1,1,1,1,1,1,0,0,0,0,0,0,1,0,0,
0,0,0,1,0,0,0,0,0,1,0,0,0,0,0,0,
1,0,0,0,0,0,1,0,0,0,0,0,1,0,0,0,
0,0,0,1,0,0,0,0,0,1,0,0,0,0,0,1,
0,0,0,0,0,0,1,0,0,0,0,0,0
2850 DATA
0,0,1,1,1,0,0,0,1,0,0,0,1,0,1,0,
0,0,0,0,1,1,0,0,0,0,0,1,0,1,0,0,
0,1,0,0,0,1,1,1,0,0,0,1,0,0,0,1,
0,1,0,0,0,0,0,1,1,0,0,0,0,0,1,0,
1,0,0,0,1,0,0,0,1,1,1,0,0
2860 DATA
0,0,1,1,1,0,0,0,1,0,0,0,1,0,1,0,
0,0,0,0,1,1,0,0,0,0,0,1,1,0,0,0,
0,0,1,0,1,0,0,0,1,1,0,0,1,1,1,0,
1,0,0,0,0,0,0,1,1,0,0,0,0,0,1,0,
1,0,0,0,1,0,0,0,1,1,1,0,0
2870 NA = 0: IT = 0: FOR XX = 0
TO 12: FOR YY = 0 TO 12: M(XX,YY)
= 0: NEXT YY: NEXT XX
2880 PRINT
CHR$(7);CHR$(7);CHR$(7);: GET
A$: IF A$ = " " THEN 520
2890 TEXT: HOME: END

```

Titre : Chantier
Machine : T07



LE CAHIER DES AS

FIGURE 1 - IMAGE DONNEE PAR LE LOGICIEL DE JEU "CHANTIER" (E. et P. BEZEMANS - 33 Le Tivoli - 13100 Aix en Provence)

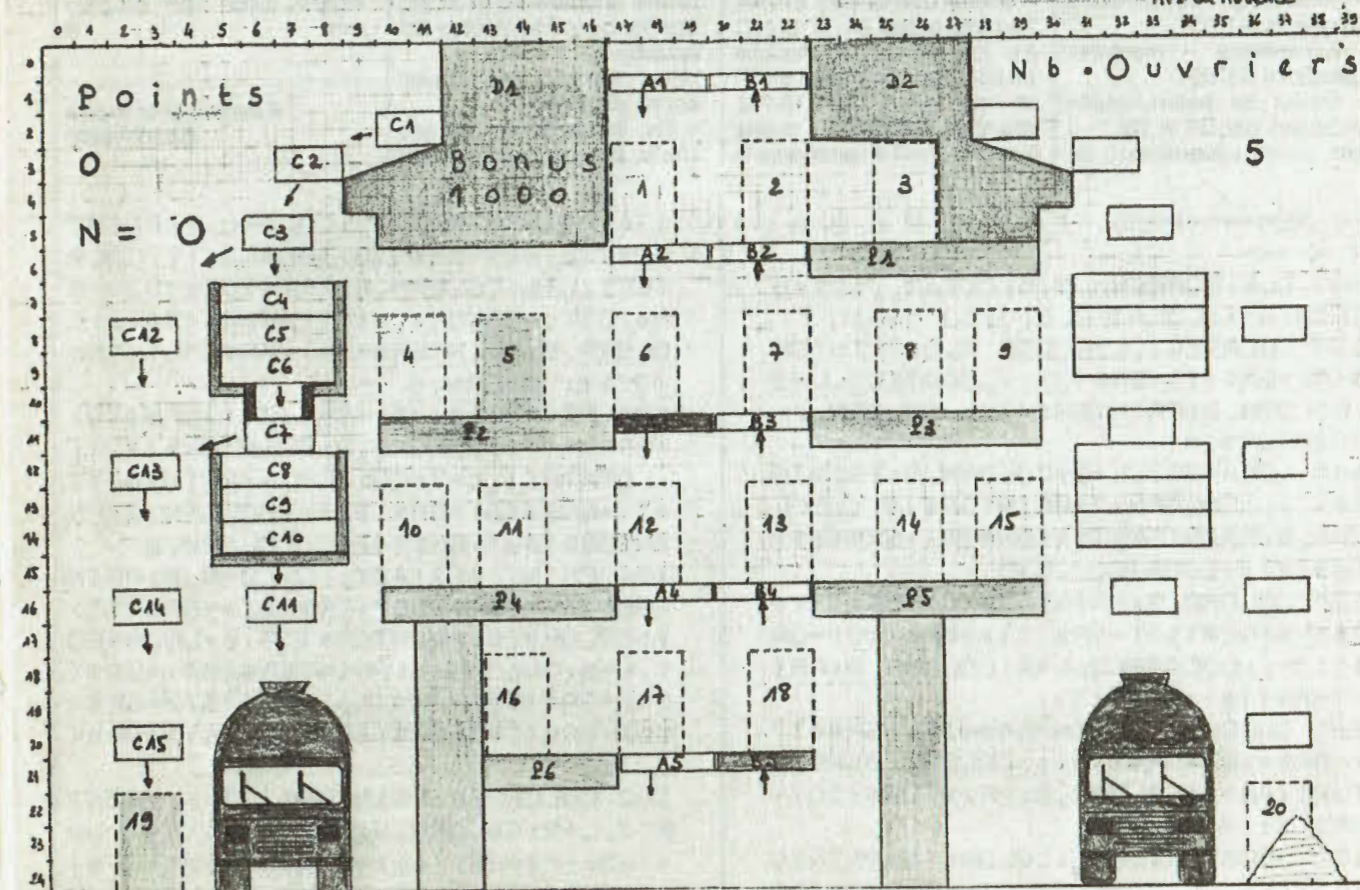


Image donnée par le logiciel de jeu « Chantier »



ATARIDEES

PROMENADE DANS LA MEMOIRE ECRAN

Avant d'aborder la question du scrolling, qui vous permettra de faire défiler un texte sur l'écran dans toutes les directions, il faut dire quelques mots de la gestion de l'écran sur les ordinateurs Atari. L'Atari a un chip spécial pour prendre les informations et les afficher sur l'écran. Ce chip s'appelle Antic.

Une Display List est ce que ANTIC examine pour indiquer à la machine comment gérer l'écran : quel mode texte ou graphique il faut afficher et où il faut prendre les data à placer sur l'écran. Les instructions de Display List sont indiquées par des POKE. Examinons maintenant en détail le listing de notre programme de scrolling.

La ligne 10 nous met en graphique 0, c'est-à-dire en mode texte. Une commande de type « GRAPHICS » crée instantanément une Display List standard et place l'adresse à laquelle elle commence dans les locations mémoire 560 et 561.

La ligne 20 nous permet d'avoir un texte sur l'écran pour la démonstration du scrolling. La ligne 30 stocke sous la variable DL la valeur de l'adresse de départ de la Display List.

La ligne 40 stocke l'octet de poids faible (DL4) et l'octet de poids fort (DL5) d'un nombre en places 5 et 6 de la Display List. DL4 et DL5 indiquent à l'ordinateur où aller chercher les data pour l'affichage. Le scrolling est obtenu en changeant ces valeurs.

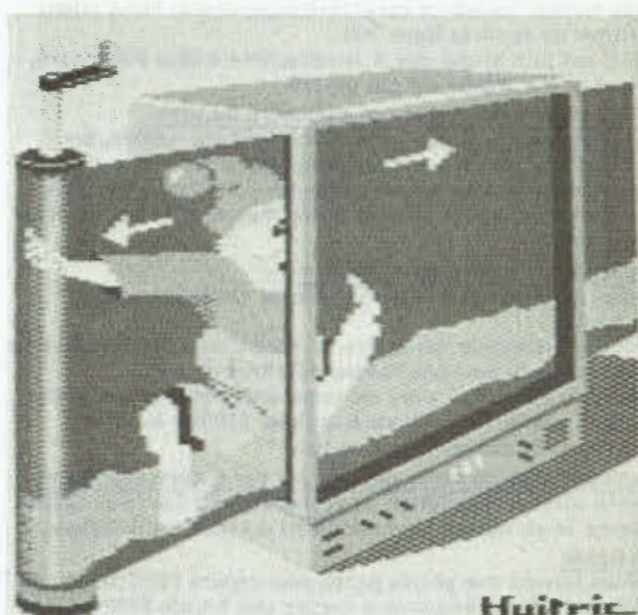
Les lignes suivantes permettent de faire varier ces

valeurs en se servant d'un joystick branché dans la première prise. Suivant les mouvements qu'on lui imprime (valeur de STICK (0)) on prend la valeur contenue par la location mémoire DL4 (NUML) et on la modifie : mouvement horizontal plus ou moins 1 ; mouvement vertical plus ou moins 40 (nombre de caractères par ligne en mode 0). Ainsi l'ordinateur sait qu'il doit commencer son affichage à un autre endroit, lorsqu'en ligne 180 on lui donne les nouvelles valeurs à stocker en DL4 et DL5.

Si on se déplaçait toujours dans la même direction, les valeurs contenues dans NUML tomberaient au-dessous de 0 ou dépasseraient 255. La ligne 180 occasionnerait alors un message d'erreur. Les lignes 100 à 177 permettent d'y remédier en changeant (-1 ou +1) l'octet de poids lourd NUMH, selon les besoins.

On peut accélérer le scrolling horizontal en remplaçant aux lignes 92 et 94 et -1 et +1 par -5 et +5 par exemple. Le scrolling vertical ne peut pas être aussi facilement accéléré. Il ne vous reste plus qu'à intégrer tout ou partie de ce programme dans vos propres créations pour obtenir un des effets les plus farfelus de l'Atari...

Nicole MASSON



```

10 GRAPHICS 0
20 LIST
30 DL=PEEK(560)+256*PEEK(561)
40 DL4=DL+4:DL5=DL+5
50 NUML=PEEK(DL+4)
60 NUMH=PEEK(DL+5)
70 ST=STICK(0)
80 IF ST=14 OR ST=10 OR ST=6 THEN
  NUML=NUML+40
90 IF ST=13 OR ST=9 OR ST=5 THEN
  NUML=NUML-40
92 IF ST=6 OR ST=7 OR ST=5 THEN NUML=
  NUML-1
94 IF ST=10 OR ST=11 OR ST=9 THEN
  NUML=NUML+1
100 IF NUML < 0 THEN 160
110 IF NUML < 256 THEN 175
120 NUML=NUML-256
130 NUMH=NUMH+1
150 GOTO 175
160 NUML=NUML+256
170 NUMH=NUMH-1
175 IF NUMH < 0 THEN 70
177 IF NUMH > 255 THEN 70
180 POKE DL4,NUML:POKE DL5,NUMH
190 GOTO 70

```

Basic : SEGA
Difficulté : *
Adaptabilité :

YENO

ATTAQUE D'UN OVNI

Des Ovnis plongés dans une guerre des étoiles sidérante !

Dans ce programme, un canon à flux électrobetamique permet de descendre les ovnis en provenance de la lointaine planète Alessuiglasse. Ne pas passer devant le jet électrobetamique c'est 2HOT4U bien que ce soit seulement 10S4ME. Après le combat vous êtes priés de vous arrêter 0 20 100 0 pour

boire un coup. Le commandement suprême de la galaxie vous souhaite bonne chance dans votre périlleuse entreprise où la seule aide que vous recevrez sera celle du capitaine Eno Duma qui n'a rien à voir avec Alexandre du même nom.

Maité AIMCEZ

```

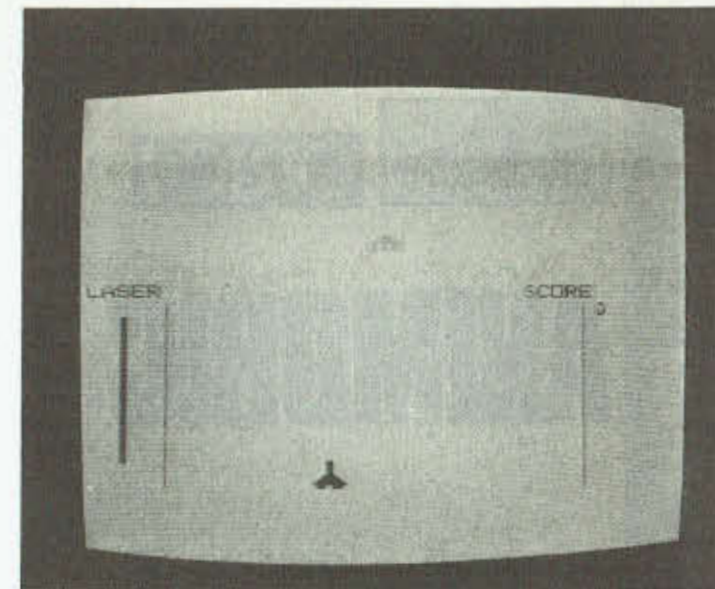
20 SCREEN 2,2:COLOR 1,15,(0,0)-(255,19
  1),15:MAG 1
30 GOSUB 130
40 IF MN=0 THEN 530
50 K*=INKEY$
60 IF K*=CHR$(28) AND MH+16<215 THEN M
  H=MH+8:GOSUB 610
70 IF K*=CHR$(29) AND MH-1>43 THEN MH=
  MH-8:GOSUB 610
80 GOSUB 320
90 FX=FX+SP:GOSUB 660
100 IF FX>235 THEN FX=0
110 IF K*=CHR$(30) THEN GOSUB 350
120 GOTO 40
130 REM **** POSITION DEPART ****
140 CLS
150 SC=0:MN=30: SX=210:SY=85
160 PATTERN S#0,"0303030303030303"
170 PATTERN S#1,"070F1F3F7FFF3C3C"
180 PATTERN S#2,"8080808080808080"
190 PATTERN S#3,"C0E0F0F6FCFE7878"
200 PATTERN S#4,"0000000000000307"
210 PATTERN S#5,"0F0F7FFF9999FF38"
220 PATTERN S#6,"000000000000C0E0"
230 PATTERN S#7,"F0F0FEFF9999FF1C"
240 LINE (43,85)-(43,189)
250 LINE (215,85)-(215,189)
260 MH=101:GOSUB 610
270 CURSOR 190,75:PRINT"SCORE"
280 CURSOR 12,75:PRINT"LASER"
290 LINE (25,93)-(27,93+MN*3),,BF
300 RETURN
310 REM **** POSITION <UFO> ****
320 FY=INT(RND(1)*55)
330 SP=INT(RND(1)*15)+5
340 RETURN

```

```

350 REM **** TIR BASE ****
360 MN=MN-1
370 BLINE (25,93+(MN+1)*3)-(27,93+MN*3)
  ,,BF
380 LINE (MH+7,170)-(MH+7,0)
390 BEEP:FOR I=1 TO 300:NEXT I
400 BLINE (MH+7,170)-(MH+7,0)
410 IF MH+7<FX OR MH+7>FX+15 THEN RETU
  RN
420 BEEP
430 LINE (FX,FY+5)-(FX+15,FY+20)
440 LINE (FX,FY+20)-(FX+15,FY+5)
450 FOR I=1 TO 300:NEXT I
460 BLINE (FX,FY+5)-(FX+15,FY+20)
470 BLINE (FX,FY+20)-(FX+15,FY+5)
480 GOSUB 710
490 SX=SX+10:IF SX=250 THEN SX=220:SY=
  SY+10
500 CURSOR SX,SY:PRINT"☠"
510 SC=SC+10:FX=0
520 RETURN
530 REM ::::: GAME OVER :::::
540 CURSOR 70,80:PRINT"*** GAME OVER *
  **"
550 CURSOR 75,100:PRINT"VOTRE SCORE : "
  ;SC
560 CURSOR 75,110:PRINT"MEILLEUR SCORE
  :";HS
570 CURSOR 60,130:PRINT"VOULEZ-VOUS RE
  JOUER (O/N)!!"
580 IF INKEY$<>"0" THEN 580
590 IF SC>HS THEN HS=SC
600 FX=0:GOTO 30
610 REM **** MOVEMENT <BASE> ****
620 SPRITE 0,(MH,172),0,1
630 SPRITE 0,(MH,172),0,1
640 SPRITE 0,(MH,173),0,1
650 RETURN
660 REM **** MOVEMENT <UED> ****
670 SPRITE 1,(FX,FY),4,1
680 SPRITE 1,(FX,FY),4,1
690 SPRITE 1,(FX,FY+1),4,1
700 RETURN
710 REM **** EFFACEMENT <UFO> ****
720 SPRITE 1,(FX,FY),4,0
730 SPRITE 1,(FX,FY),4,0
740 SPRITE 1,(FX,FY+1),4,0
750 RETURN

```



Basic : Oric
 Difficulté : *
 Adaptabilité :

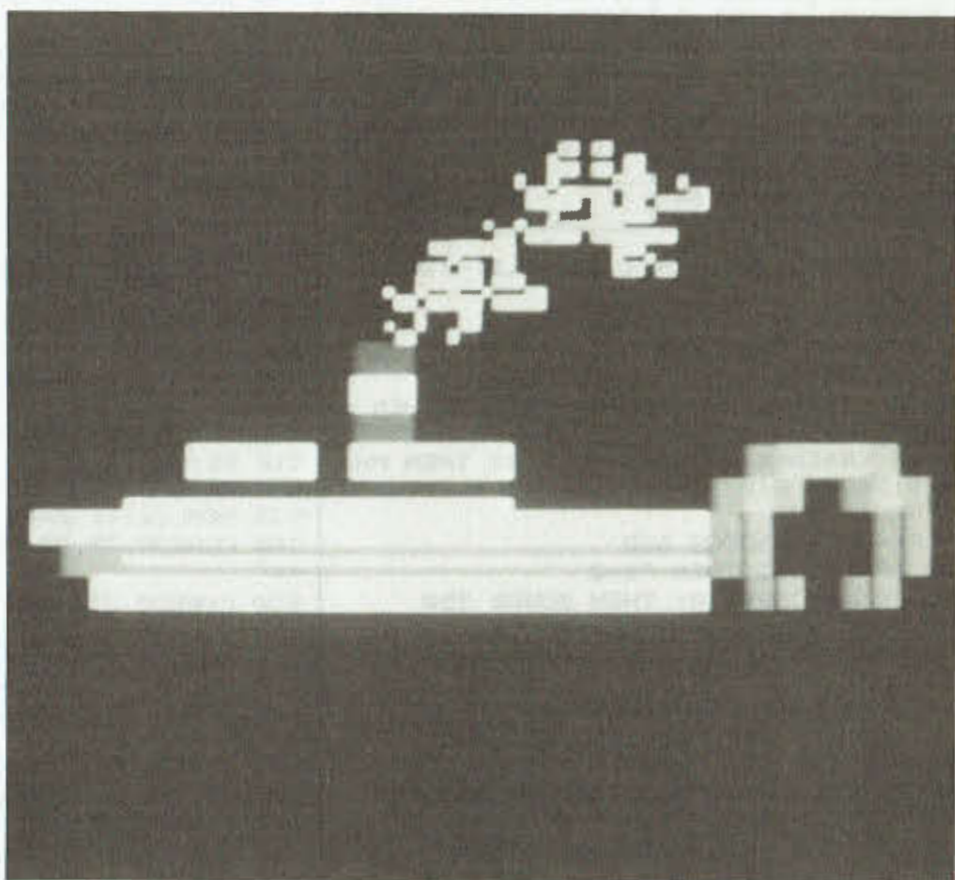
ORIC-1

ANIMORIC

Ne pas confondre « Dessin Assisté par Ordinateur » (DAO) et « Dessin Animé par Ordinateur » (DAO), quand l'Oric se prend pour un fusain...

Voici un programme sur Oric qui vous permet de créer un dessin animé. Après lancement du programme, l'ordinateur vous demande si vous voulez les explications. Si oui il vous envoie aux explications, sinon le menu principal s'affiche. Vous avez le choix entre créer une image, voir et modifier une image existante, voir le dessin animé, sauvegarder le dessin animé en mémoire, charger un dessin animé créé précédemment, revoir les explications et arrêter de jouer. Toutes les explications nécessaires sont données dans le programme. Des caractères graphiques spéciaux sont programmés dont vous trouverez facilement les correspondances. L'idée du programme peut être utilisée pour d'autres ordinateurs mais le programme en lui-même est difficilement adaptable du fait de l'utilisation du langage machine. ■

Jacques JOFFRE



```

5 REM**** DESSIN ANIME pour ORIC *
***
10 REM **COPYRIGHT Jacques JOFFRE
1984
12 REM ** 10, AVENUE DU Professeur
GRASSET 34000 MONTPELLIER
15 GRAB
18 REM POUR RETROUVER LA FIN DU PR
OGRAMME EN CAS DE CHARGEMENT D'IM
AGE
20 IF DEEK(#9E) > #2400 THEN DB=DEE
K(#450):DOKE#9C,DB:DOKE#9E,DB:DOK
E#A0,DB
30 HIMEM #2400
50 DIM A(40),C(40)
60 E#=CHR$(27):CD#=CHR$(4)
70 INK6:PAPER0
100 REM *** PROGRAMME DE TRANSFER
T EN LANGAGE MACHINE***
105 M=0
110 REPEAT

```

Traitement d'une lettre

Nous décalons le titre à la colonne E. Le ; indique que les caractères que nous afficherons ensuite seront affichés sur la même ligne.

Un passage sur chaque lettre du titre

Nous traitons tous les caractères de la chaîne A\$(titre) du premier, FOR I=1, jusqu'au dernier, TO L, un par un. STEP 1 n'est pas obligatoire. Remarquez le NEXT I de la ligne 330 qui indique la fin de la boucle. Les instructions comprises entre le FOR I et le NEXT I seront exécutées L fois.

L'instruction MID\$(sert à extraire des caractères d'une chaîne. Le premier chiffre, I, indique le premier caractère que l'on veut extraire. Le second, L, le nombre de caractères. Au premier passage nous prendrons donc le premier caractère du titre, au second, le deuxième et ainsi de suite.

Nous prenons la valeur ascii de ce caractère et nous la mettons dans la variable X.

Si D est plus petit que X cela veut dire que nous n'avons pas encore trouvé ce caractère auparavant. Nous allons traiter ce cas à la ligne 300

Si D est plus grand que X le caractère a déjà été trouvé. Nous allons traiter ce cas en 270

Nous venons de trouver un caractère du titre.

Si D est égal à X, les codes Ascii sont les mêmes, les caractères sont les mêmes. Nous avons trouvé un caractère. Nous faisons sonner la cloche pour l'annoncer. Vous pourrez changer la couleur à cet endroit si votre machine le permet.

Nous avons trouvé ce caractère

Nous ajoutons un à notre somme de caractères déjà trouvés.

Nous affichons le caractère du titre correspondant à cette colonne, puisque nous l'avons déjà trouvé

Nous en avons fini avec cette colonne, nous allons passer à la suivante. Nous allons à la ligne 310 car la ligne suivante ne concerne pas ce cas.

Le caractère de cette colonne n'a pas encore été trouvé

Nous affichons la lettre de recherche. La fonction CHR\$(donne le caractère correspondant au code ascii indiqué.

Attente

Nous faisons une petite pause pour rendre l'affichage plus agréable. Nous employons encore une boucle FOR. Cette boucle ajoutera 1 à P, jusqu'à ce que P soit égal à T (cf ligne 100).

Lorsque P=T le programme continue à l'instruction suivant le NEXT.

Ce NEXT correspond au FOR de la ligne 210. Si nous n'avons pas traité toutes les colonnes, le programme retourne à la ligne 220 pour traiter la colonne suivante.

Fin de traitement d'une lettre.

Ce PRINT permet de passer à la ligne pour le prochain passage. Jusqu'ici les ; nous ont fait écrire sur la même ligne.

Nous devons maintenant utiliser la lettre suivante. Nous augmentons D d'une unité. Le caractère correspondant à sa valeur ascii sera donc le suivant.

Nous nous demandons, avant de recommencer, si nous n'avons pas trouvé tous les caractères de notre titre. Si S est plus grand ou égal au nombre de lettres de notre titre (L), alors ce n'est pas la peine de continuer. Nous allons à la fin (380)

Nous vérifions si nous n'avons pas dépassé la dernière lettre de l'alphabet : ASC("Z") donne la valeur ascii de la lettre Z. Ceux qui ont des minuscules pourront mettre ASC("z"). la recherche continuera jusqu'à la fin des minuscules. Mais attention, entre Z et a, il peut se passer des choses bizarres.

Si nous n'avons pas fini notre alphabet, nous allons à la ligne 190, pour recommencer une rangée.

Sinon c'est fini.

```

200 PRINT TAB( E);
210 FOR I = 1 TO L
220 W# = MID# (A#,I,1)
230 X = ASC (W#)
240 IF D < X THEN GOTO 300
250 IF X < > D THEN GOTO 270
260 PRINT B#;
270 S = S + 1
280 PRINT W#;
290 GOTO 310
300 PRINT CHR# (D);
310 FOR P = 1 TO T
320 NEXT P
330 NEXT I
340 PRINT
350 D = D + 1
360 IF S > = L THEN GOTO 380
370 IF D < = ASC ("Z") THEN
GOTO 190
380 END

```

INITIATION

VENDREDI SHOW

Vous avez sans doute remarqué le générique de l'émission de télévision Vendredi (fort bonne au demeurant) sur la troisième chaîne. Nous allons reproduire ce générique sur notre machine. Le programme n'emploiera pas de couleurs ni d'effets style INVERSE, afin d'être compatible avec un maximum de machines. Vous pourrez toutefois l'améliorer par la suite. Ce programme va nous permettre de jongler avec les caractères, et même de faire des calculs avec des lettres. En effet, un ordinateur ne travaille qu'avec des zéros et des uns (binaire). Les lettres doivent donc être codées. Le code le plus employé dans les micro-ordinateurs est le code ascii. Ce code est un code étendu à 8 bits (8 zéros ou un), ce qui permet de coder 256 caractères différents. Un tableau de correspondance code/caractères doit être donné dans votre manuel. Grâce

Initialisation

Les lettres s'afficheront à l'écran l'une après l'autre. La variable T (numérique) servira de temporisation entre deux lettres. Nous mettons 30 pour le moment, mais vous pourrez augmenter cette valeur si vous désirez ralentir l'affichage, ou la diminuer pour accélérer. N indique le nombre de colonnes de l'écran. Si votre écran n'a pas 40 colonnes, changez la valeur de N. CHR\$(7) est sur de nombreuses machines le caractère qui fait sonner la cloche. Si vous n'en n'avez pas, supprimez cette ligne et la ligne 260. Nous allons commencer notre recherche à partir de la lettre A. La valeur ascii de A est 65. Nous initialisons D à 65. Nous mettons le titre dans la zone A\$(alphanumérique). Vous pouvez mettre ce que vous voulez. La fonction LEN(donne le nombre de caractères qu'il y a dans la zone A\$. Pour VENDREDI, L égal 8.

On vérifie si le titre rentre en entier dans l'écran. S'il rentre, on indique au programme d'aller directement à la ligne 180. (Certaines machines n'acceptent pas la syntaxe THEN GOTO, il faudra retirer l'un des deux; consultez votre manuel en cas de doute).

Le titre est plus long que la largeur de l'écran.

La variable L contient la longueur du titre. Il suffit de lui donner maintenant la valeur de la taille de l'écran moins un (afin de ne pas avoir de saut de ligne supplémentaire). L'idéal serait que le titre soit au milieu de l'écran. Nous calculons la colonne de début. N-L correspond au nombre de caractères qui ne seront pas utilisés par le titre. Si l'on divise par deux, on obtiendra le nombre de colonnes à gauche. Nous prenons la valeur entière (les chiffres avant la virgule) car il n'est pas possible d'écrire à une demi-colonne. E indique le début du titre.

Début du traitement

Le traitement va continuer jusqu'à ce que l'on ait trouvé toutes les lettres du mot. Nous compterons les lettres trouvées. Lorsque nous en aurons trouvé autant qu'il y en a dans le titre (L) nous arrêterons tout. Il n'est pas nécessaire d'aller jusqu'à Z.

à ce code, lorsque nous tapons le caractère A la machine stocke sa valeur ascii (65), et lorsque nous disons d'afficher un caractère, elle affiche le petit dessin correspondant au code ascii 65 (A). Pour utiliser un caractère, il faut deux informations à la machine; d'abord dire qu'il s'agit d'un caractère, en Basic on utilise les guillemets et les variables postfixées \$, ensuite lui donner la valeur ascii du caractère que l'on désire (en fait c'est ce que l'on fait lorsque l'on tape une lettre au clavier). Le programme "Vendredi show" doit afficher d'abord une série de A, puis B et ainsi de suite, à la place de chaque lettre du mot VENDREDI. Lorsque la lettre qui défile passe sur son équivalent dans le mot, elle s'affiche définitivement à cet endroit. On continuera jusqu'à ce que le mot soit entièrement trouvé.

```

100 T = 30

110 N = 40
120 B$ = CHR$(7)

130 D = 65

140 A$ = "VENDREDI"

150 L = LEN (A$)

160 IF L < N THEN GOTO 180

170 L = N - 1

180 E = INT ((N - L) / 2)

190 S = 0

```

```

120 READ D0
130 POKE #400 +M,D0
140 M=M+1
150 UNTIL D0=#FF
200 DATA #A9,#BB, #85,#09, #A2,#
04, #A0,#00, #B1,#06, #91,#08,
#88
210 DATA #D0,#F9, #E6,#07, #E6,
#09, #CA, #D0,#F0 ,#60, #FF
350 CLS
400 REM ADRESSE DU DEBUT DE RANGEM
ENT DES 35 IMAGES DIFFERENTES POS
SIBLES
410 FOR I=1 TO 35
420 A(I)=#2400+(I-1)*#400
430 NEXT
500 REM ***MINUSCULES ACCENTUEES**
*
505 DC=46080
510 FOR J=1 TO 6
515 READ C0,0(0),0(1),0(2),0(3),0(
4),0(5),0(6),0(7)
520 DATA 64,16,8,28,2,30,34,30,0
0
522 DATA 92,00,00,30,32,32,30,30,0
4
524 DATA 94,08,20,00,00,00,00,00,
00
526 DATA 123,04,08,28,34,62,32,30,
00
528 DATA 124,16,08,34,34,34,38,26,0
0
530 DATA 125,16,08,28,34,62,32,30,
00
535 D=CO*8
540 FOR I=0 TO 7
550 POKE DC+D+I,0(I)
555 NEXT I:NEXT J
600 PRINT:PRINT:PRINT
610 PRINTCD$;E$;"A";E$;"J ***
DESSIN ANIME ***":PRINTCD$
615 PRINT:PRINT:PRINT:PRINT
620 PRINT"Congaissez vous ce progr
amme de dessin anim( : (0/N)":GE
TR$
640 IF R$<>"0"THEN GOSUB 1000
900 CLS
905 POKE 618,3:PRINT
910 PRINT"Pour dessiner la PREMIER
E IMAGE";:PRINTE$;"A : 1"
930 PRINT
932 PRINT"Pour VOIR une image d(j@
dessin(e,
933 PRINT"puis la MODIFIER et la r
anger ";:PRINTE$;"B : 2":PRIN
T
940 PRINT"Four voir le DESSIN ANIM
E ";:PRINTE$;"C : 3":PRINT
:PRINT:PRINT
942 PRINT
945 PRINT "Pour SAUVEGARDER le des
sin anim("
946 PRINT"que vous venez de cr(er
: 4 " :PRINT
950 PRINT "Pour CHARGER"
951 PRINT "un dessin anim( ANCIEN
: 5 " :PRINT
955 PRINT "Pour revoir les explica
tions : 6 " :PRINT
975 PRINT "Pour ARRETER de jouer
: 0 "
980 GET R$ :IFR$="0" THEN END
982 R=VAL(R$)
985 ON R GOSUB 1500,3000,2000 ,400
0,5000,1000
995 GOTO 900
1000 REM *** EXPLICATIONS ***
1002 CLS:PRINT
1005 PRINTCD$;E$;"A";E$;"J PRINCIP
E DE REALISATION D'UN ":PRINT
1010 PRINT E$;"A";E$;"J *** DE
SSIN ANIME ***":PRINTCD$;:PRINT:
PRINT:PRINT
1015 PRINT"--On dessine d'abord une
image sur l'(cran."
1020 PRINT "Pour cela on dispose d
e petits carr(s"
1022 PRINT "qui sont color(s a l'a
ide des touches ";
1023 PRINTE$;"A0,1,2 ...@...9":PRI
NT
1024 PRINT"On peut d(placer ces pe
tits carr(s"
1025 PRINT "@ l'aide des touches f
l(ch(es" :PRINT:PRINT:PRINT:PRINT
1026 PRINT"(POUR CONTINUER, AFFUYE
R SUR UNE TOUCHE)"
1027 GETA$ :CLS
1030 PRINT"-- Quand on a fini cette
premi}re "
1032 PRINT"image, on peut la mettr
e en m(moire"
1035 PRINT"en appuyant sur la touc
he";:PRINTE$;"A$ " :PRINT:PRINT
1040 PRINT "-- La deuxi}me image (i
mage num(ro 2)"
1042 PRINT"est obtenue en appelant
@ l'(cran la"
1045 PRINT "premi}re image que l'o
n va modifier l(g)rement."
1047 PRINT:PRINT"La succession rap
ide des images"
1050 PRINT"donnera l'impression de
mouvement."
1055 PRINT:PRINT "On peut revoir

```

```

(en l'appelant par son ";
1057 PRINT"num{ro} une image d{j@
dessin{e.
"
1060 PRINT"Ensuite on la modifie p
uis on la range";
1062 PRINT"@ un emplacement pr{ala
blement d{fini.":PRINT
1065 PRINT:PRINT:PRINT:PRINT
1070 PRINT"POUR CONTINUER, APPUYER
SUR UNE TOUCHE"
1080 GETA$
1200 CLS
1205 PRINTTAB(14);E$;"A POUR
DEFINIR L'IMAGE ":PRINT
1210 PRINT"1) S{lectionner une cou
leur:
"
1211 PRINT "TOUCHE 1 : ROUGE": P
RINT "TOUCHE 2 : VERT"
1212 PRINT "TOUCHE 3 : JAUNE": P
RINT "TOWAITCHE 4 : BLEU"
1213 PRINT "TOUCHE 5 : MAGENTA":
PRINT "TOUCHE 6 : CYAN"
1214 PRINT "TOUCHE 7 : BLANC ":
PRINT
1215 PRINT "TOUCHE 0 : NOIR (EFF
ACEMENT)":PRINT
1216 PRINT "TOUCHES 8 ET 9 : INVI
SIBLE (RIEN)":PRINT
1217 PRINT"Les autres touches donn
ent des
1218 PRINT"caract}res graphiques p
our raffiner l'image":PRINT:PRI
NT
1219 PRINT"2) D{placer le curseur
avec les touches fl{ches":
PRINT:PRINT
1220 PRINT "POUR CONTINUER APPUYER
SUR UNE TOUCHE": GETA$:CLS
1230 PRINT:PRINT:PRINT:PRINT"Quand
votre image est termin{e,"
1240 PRINT "appuyez sur $ pour la
sauvegarder et/ou revenir au
menu":PRINT:PRINT
1250 PRINT"Si vous voulez recommen
cer une image,"
1260 PRINT"appuyez sur ESC pour l'
effacer ":PRINT:PRINT
1265 PRINT"On peut mettre jusqu'@
35 images "
1267 PRINT"diffr{entes dans le des
sin anim{":PRINT:PRINT:PRINT
1270 PRINT "POUR VOIR LE MENU APPU
YER SUR UNE TOUCHE": GETA$
1290 RETURN
1500 REM *** SAISIE DE LA PREMIERE
IMAGE ***
1510 I=1
1550 DOKE 06,#BB80

```

```

1560 DOKE 08,A(I)
1590 CLS:LORES1
1600 GOSUB 7000
1650 GOSUB 6000
1660 CLS:GOTO 930
1670 GOTO 930
2000 REM *** DESSIN ANIME**
2005 CLS
2010 PRINTTAB(15);E$;"A VISUA
LISATION DU DESSIN ANIME":PRINT:F
RINT:PRINT:PRINT
2015 PRINT"Pour arr{ter le dessin
anim{, appuyer sur la barre d'esp
acement"
2017 PRINT:PRINT:PRINT
2020 INPUT "COMBIEN AVEZ VOUS D'I
MAGES DANS VOTRE DESSIN ANIME":N
2030 PRINT:PRINT:PRINT:PRINT
2040 PRINT "INDIQUEZ LA VITESSE DE
SUCCESSION DES IMAGES ";
2045 PRINT"(intervalle de temps en
tre 2 images en 1/100 de second
es)":PRINT
2050 INPUT" 1 :VITESSE LA PLUS RAP
IDE ";V
2060 POKE 618,2
2100 FOR I=1 TO N
2120 DOKE 06,A(I)
2130 DOKE 08,#BB80
2200 CALL #0400
2220 WAIT V
2230 IF KEY$=" " THEN RETURN
2240 NEXT
2300 GOTO 2100
3000 REM *** MODIFICATION D'UNE I
MAGE***
3050 CLS:PRINT:PRINT:
3060 PRINT"*** MODIFICATION D'UNE
IMAGE***
3100 PRINT:PRINT:PRINT "Entrez le
num{ro de l'image que vous voul
ez voir : "
3110 INPUT NV:PRINT:PRINT:PRINT
3120 PRINT"Entrez le num{ro de l'e
mplacement o{ vous voulez mettre
votre";
3130 PRINT" image quand":PRINT"vo
us l'aurez modifi{e." :INPUT I
3200 DOKE 06,A(NV)
3210 DOKE 08,#BB80
3220 CLS :LORES 1
3250 CALL #400
3270 GOSUB 7000
3300 DOKE 06,#BB80
3310 DOKE 08,A(I)
3350 GOSUB 6000
3400 RETURN
4000 REM ** SAUVEGARDE DU D.A.**

```

```

1210 FOR T=1 TO 80 :: CALL POSITION{£3,YA,XA):: IF XA<25 OR XA>230 THEN CALL DEL
SPRITE{£3)
1220 NEXT T :: CALL DELSPRITE{£3)
1230 IF NBCHAR=0 THEN CALL HCHAR(23,10,137,12):: GOTO 1250
1240 CALL HCHAR(23,10,137,12):: PASS=0 :: CALL SPRITE{£1,96,5,157,50):: GOTO 840
1250 REM FIN PARTIE
1260 IF SCORE<HISCO THEN 1280 ELSE HISCO=SCORE
1270 IF SCORE>99999 THEN DISPLAY AT(1,17):"SCORE" :: DISPLAY AT(1,22):SCORE ELSE
DISPLAY AT(1,23):SCORE
1280 FOR T=1 TO 200 :: NEXT T
1290 DISPLAY AT(11,10):"GAME OVER"
1300 FOR T=1 TO 20
1310 CALL KEY(5,KEY,STATUS):: IF STATUS=0 THEN 1340
1320 IF KEY=6 THEN 1430
1330 IF KEY=15 THEN 1420
1340 NEXT T
1350 DISPLAY AT(20,3):"APPUYER SUR REDO OU BACK" :: DISPLAY AT(5,1):: DISPLAY AT
(11,10)
1360 FOR T=1 TO 20
1370 CALL KEY(5,KEY,STATUS):: IF STATUS=0 THEN 1400
1380 IF KEY=6 THEN 1430
1390 IF KEY=15 THEN 1420
1400 NEXT T
1410 GOTO 1290
1420 CALL DELSPRITE(ALL):: HISCO=0 :: GOTO 100
1430 DISPLAY AT(11,1):: DISPLAY AT(20,1):: DISPLAY AT(1,8)SIZE(7):"00000"
1440 DISPLAY AT(23,27)SIZE(3):"5" :: NPASS=NPASS1 :: GOTO 730
1450 REM SOUS PROGRAMMES
1460 DISPLAY AT(24,2)BEEP:"BARRE ESPACE POUR CONTINUER"
1470 CALL KEY(5,KEY,STA):: IF STA=0 THEN 1470 ELSE RETURN
1480 FOR L=3 TO 24 :: DISPLAY AT(L,1):: NEXT L :: RETURN

```



Vente Informations Services Micro-Ordinateurs
Micro-ordinateurs familiaux

84, bd Beaumarchais - 75011 Paris
 M{tro Bastille ou Chemin Vert
 De 10 h à 20 h sauf dimanche.

88, bd de Reuilly - 75012 Paris
 M{tro Daumesnil ou Dugommier
 De 14 h à 20 h sauf dimanche et lundi

BON DE COMMANDE
à retourner accompagné de votre
r{glement à VISMO - 84, bd Beau-
marchais - 75011 Paris

Contre-remboursement + 60 F
R{glement joint + 30 F

TOTAL :

Catalogue VISMO seul 20 F
(à d{duire sur 1^{re} commande)

Nom _____

Pr{nom _____

Adresse _____

T{l. _____

Signature _____

 ZX SPECTRUM <small>p{r{ital 48 K</small> 2.325 F	 ORIC ATMOS <small>48 K</small> 2.480 F	 SANYO PHC 88 <small>nouveau standard MSX</small> 3.000 F
 THOMSON MO8 <small>micro-ordinateur fran{ais</small> 2.390 F	 EKL 100 <small>command{ à distance par infrarouge micro-ordinateur fran{ais</small> 3.200 F	<p>Ordinateur ZX 81 - coffret familial 650</p> <p>Moniteur Z{nith 1050</p> <p>Imprimante GP 500 A 2450</p> <p>Imprimante GP 50 A 1350</p> <p>Lecteur K7 ordinateur 680</p> <p>K7 MONASM VISMO</p> <p>pour Oric 1 et Atmos 160</p> <p>Synth{tiseur vocal Oric 550</p> <p>Synth{tiseur vocal ZX 81 435</p> <p>Catalogue VISMO 20 F <small>remboursable à la 1^{re} commande.</small></p> <p>Interface imprimante Minitel. P{riph{riques. Accessoires. Logiciels. K7 jeux. Bibliographie. Nous consulter T{l. (1) 338.60.00</p>

```

660 BN$="03070F1F3F7F7F7F3F1F0F0703000303COE0F0F8FCFEFEFEFCF8F0E0C000COC0"
670 AR$="00000000030F39F1390F03000000000020F3FFEC6FEFEDFFEFEC6FE3F0F0200"
680 GA$="000E1B7F7F000000000000001C36FFFF030307FFFE00000000000006060EFC"
690 BO$="04060607070707070703010000000000103030F0F0F0F0F0E0C08000000000"
700 L1$="0510054014822882288114410822080240104400A20821841144114802A008A0"
710 L2$="000002080114022902280210040100000000A000482094402480288010400000"
720 CALL CHAR(132,L1$,140,L2$)
730 CALL CHAR(96,TK$,100,BL$,108,AV$,120,SC$,104,BN$,116,AR$,112,GA$,128,ET$,124,BO$):: CALL MAGNIFY(3):: CALL SPRITE(£1,96,5,157,50)
740 DATA 8,11,7,5,15,13,6,9,14,16,8,11,7,5,15,13,6,9,14,16,8
750 TOUR=0 :: NB=1 :: VDEB=10 :: NBCHAR=5 :: SCORE=0 :: RESTORE 740
760 RANDOMIZE :: ALT=INT(96*RND)+25 :: PASS=0 :: DISPLAY AT(23,3)SIZE(4):NB
770 IF TOUR=0 THEN 830
780 IF INT(TOUR/10)<>TOUR/10 THEN 790 ELSE NBCHAR=NBCHAR+1 :: DISPLAY AT(23,26)SIZE(3):NBCHAR
790 IF INT(TOUR/5)<>TOUR/5 THEN 840
800 IF TOUR<>100 THEN 820

810 NPASS=1 :: TOUR=0 :: VDEB=10 :: RESTORE 740 :: GOTO 830
820 VDEB=VDEB+5
830 READ CO :: PA=104 :: V=VDEB
840 FOR TEM=1 TO 3 :: CALL SOUND(400,1047,6,523,12,262,12):: CALL KEY(5,KEY,STA) :: IF KEY<>80 THEN 940
850 DISPLAY AT(23,11)SIZE(7):" PAUSE "
860 FOR T=1 TO 15 :: CALL KEY(5,KEY,STA):: IF STA=0 OR KEY=80 THEN 870 ELSE 920
870 NEXT T
880 DISPLAY AT(23,11)SIZE(7)
890 FOR T=1 TO 10 :: CALL KEY(5,KEY,STA):: IF STA=0 OR KEY=80 THEN 900 ELSE 920
900 NEXT T
910 GOTO 850
920 CALL HCHAR(23,10,137,12):: FOR T=1 TO 100 :: NEXT T
930 GOTO 840
940 NEXT TEM
950 CALL SPRITE(£3,PA,CO,ALT,230,0,-V)
960 CALL KEY(1,KEY,STATUS):: IF KEY=18 THEN CALL POSITION(£1,YT,XT):: CALL SPRITE(£2,100,16,141,XT)ELSE 1030
970 CALL MOTION(£2,-80,0,£1,0,0):: CALL SOUND(100,-5,2)
980 CALL COINC(£2,£3,12,C):: IF C<>-1 THEN CALL POSITION(£3,YA,XA)ELSE 1060
990 IF XA<18 OR XA>238 THEN PASS=PASS+1
1000 CALL COINC(£2,£3,12,C):: IF C<>-1 THEN CALL POSITION(£2,Y,X)ELSE 1060
1010 IF Y<175 THEN 980 ELSE CALL DELSPRITE(£2)
1020 IF PASS=NPASS THEN 1140 ELSE 960
1030 CALL POSITION(£3,YA,XA):: IF XA<18 OR XA>238 THEN PASS=PASS+1
1040 CALL JOYST(1,VT,WT):: CALL POSITION(£1,YT,XT):: IF (XT<40 AND VT<0)OR(XT>220 AND VT>0)THEN CALL MOTION(£1,0,0):: GOTO 1020
1050 CALL MOTION(£1,0,4*VT):: GOTO 1020
1060 CALL DELSPRITE(£2,£3):: CALL SPRITE(£2,132,CO,YA,XT):: CALL SOUND(350,-5,2) :: CALL SOUND(150,-5,15)
1070 CALL SPRITE(£2,140,CO,YA,XT):: CALL SOUND(100,-5,25):: CALL DELSPRITE(£2)
1080 IF XT>52 THEN VT=-10 ELSE 1090
1090 IF PASS<>0 THEN POINT=INT(10*V/PASS)+120-ALT :: PASS=0 ELSE POINT=15*V+120-ALT
1100 SCORE=SCORE+10*(INT(POINT/10)):: DISPLAY AT(1,7)SIZE(7):SCORE
1110 CALL POSITION(£1,YT,XT):: DIF=INT(50-XT)/2 :: IF ABS(DIF)<=1 THEN CALL MOTION(£1,0,0):: GOTO 1130 ELSE CALL MOTION(£1,0,DIF)
1120 GOTO 1110
1130 V=V+5 :: TOUR=TOUR+1 :: NB=NB+1 :: PA=PA+4 :: GOTO 760
1140 CALL SOUND(100,660,2):: CALL SOUND(100,770,2):: CALL MOTION(£1,0,0):: CALL POSITION(£1,YT,XT):: CALL POSITION(£3,YA,XA)
1150 DISPLAY AT(23,8)SIZE(12):"TIR IMPRECIS"
1160 CALL COINC(£3,YA,XT,8,CB):: IF CB<>-1 THEN 1160 ELSE CALL SPRITE(£4,124,11,YA+16,XT,40,0)
1170 CALL COINC(£1,£4,8,CEX):: IF CEX<>-1 THEN 1170 ELSE CALL DELSPRITE(£1,£4)
1180 CALL SPRITE(£2,132,10,YT,XT):: CALL SOUND(350,-5,2):: CALL SOUND(150,-5,15) :: CALL SPRITE(£2,140,7,YT,XT):: CALL SOUND(100,-5,25)
1190 CALL DELSPRITE(£2)
1200 NBCHAR=NBCHAR-1 :: DISPLAY AT(23,26)SIZE(3):NBCHAR :: CALL SOUND(2700,-8,2)

```

```

4005 REM POUR SAUVEGARDER A VITESS
E LENTE MODIFIER LA LIGNE 4050
4010 CLS:PRINT:PRINT
4020 PRINT" SAUVEGARDE DE VOTRE
DESSIN ANIME"
4025 PRINT:PRINT
4030 PRINT"Mettez le magn(ophone
en position ENREGISTREMENT et
":PRINT
4032 PRINT"Entrez le nom de votre
dessin anim(:INPUT NO$
4034 PRINT:PRINT:PRINT:PRINT
4036 PRINT"Attention d's que vous
appuyerez sur RETURN , il s'enre
gistrera.
4037 PRINT:PRINT:PRINT
4038 PRINT"Entrez le nombre d'imag
es de votre dessin anim("
4040 INPUT NI
4042 AF=#2400+#400*NI
4050 CSAVE NO$,A#2400,EAF :REM
SAUVEGARDE DES IMAGES A PARTIR DE
#2400
4055 PRINT:PRINT
4060 PRINT"Dessin anim( ";NO$;"
sauv(:WAIT 200
4080 RETURN
5000 REM ** CHARGEMENT DU D.A.**
5005 REM POUR CHARGER A VITESSE LE
NTE MODIFIER LA LIGNE 5070
5010 CLS:PRINT:PRINT
5020 PRINT" CHARGEMENT D'UN DESSI
N ANIME ANCIEN "
5025 PRINT:PRINT
5030 PRINT"Positionnez la bande et
mettez le magn(ophone en LE
CTURE"
5035 PRINT"Indiquez le nom de votr
e dessin anim("
5037 PRINT"(si vous avez des doute
s sur le nom, entrez !)"
5040 PRINT"Apr's le chargement du

```

```

D.A. faites RUN pour relancer le
programme
5042 INPUT NO$
5045 IF NO$="!" THEN NO$=""
5060 DOKE#450, DEEK(#9E)
5070 CLOAD NO$,A#2400,E#B000
5080 RETURN
6000 REM
6020 FOR J=0 TO 39:POKE 48000+J,32
:NEXT
6050 CALL #404
6080 POKE 618,3
6090 RETURN
7000 REM***AIDE A LA PREPARATION D
ES IMAGES***
7010 A$="Dessin de l'image No"+STR
$(I)+CHR$(1)+" Pour finir:"+"$"
7015 FOR J=1 TO LEN(A$):C(J)=ASC(M
ID$(A$,J,1))
7020 POKE 48000+J,C(J):NEXT
7110 X=0 :Y=0 :C=25
7115 PRINT CHR$(9);
7130 GET A$ :AS=ASC(A$)
7132 IF A$=" $" THEN PLOT X,Y,16 :PO
KE 618,0:RETURN
7133 IF AS=27 THEN CLS:LORES1:GOTO
7000
7135 IF AS>47 AND AS<58 THEN C=VAL
(A$)+16 ELSE IF AS>31 THEN C=AS
7140 IF C=24 OR C=25 THEN 7150
7145 PLOT X,Y,C
7150 IF ASC(A$)=9 AND X < 38 THEN
X=X +1 :PRINTCHR$(9)::GOTO7130
7155 IF ASC(A$)=8 AND X > 0 THEN
X=X -1 :PRINTCHR$(8)::GOTO 7130
7160 IF ASC(A$)=11 AND Y > 0 THEN
Y= Y-1: PRINTCHR$(11) ::GOTO 713
0
7170 IFASC(A$)=10 AND Y < 23 THEN
Y=Y+1 :PRINT CHR$(10);
7190 GOTO7130

```

LES CHAMPIONS DU MOIS

ORIC :

Didier Vidal (78 Plaisir) - Bruno Alexandrian (13 Aubagne) - Michel Vachta (21 Messigny) - Gabriel Poinignon (57 Hagondage)

T.I. :

Stéphane Biden (12 Rodez) - Laurent Pollier (73 Aix-les-Bains) - Bruno Rivière (60 Precy sur Oise) - Patrick Laurent (59 Maubeuge)

T.O.7. :

Laurent N'Guyen (91 Savigny sur Orge) - Jean-Claude Paulin (75 Paris) - Emmanuel et Pierre Bertrand (13 Aix en Provence) - Michel Boissi (83 La Valette) - Michel Thieblot (06 Le Rouret)

- André Blain (95 L'isle Adam)

CASIO :

Laurent Drocourt (93 Montreuil) - Eric François (94 Vitry).

CANON X-07 :

Guillaume Forest (07 Granges les Valence).

HECTOR :

Michel Guillout (87 Isle).

VIC 20 :

Fabrice Dimitriou (07 Satillieu).

SPECTRA VIDEO :

Pascal Chambon (71 Châlonarne).

ZX 81 :

Guillaume Macé (27 Gaillon) - Philippe

Moreau (86 Poitiers) - François Hache (24 Belves) - Régis Laigle (80 Abbeville).

Spectrum :

César Martinez-Barbosa (93 Aubervilliers).

LYNX :

Vantha Sok Cham (91 Evry) - Stéphane Butel (14 Falaise).

MC 10 :

Dominique Bon (44 Nantes).

BBC :

Yves Torre (38 Grenoble).

COMMODORE 64 :

Alain Messelot (70 Oray) - Guillaume Kollar (59 Douai).

Basic : Basic Etendu
 Difficulté : **
 Adaptabilité : ***



TEXAS
 INSTRUMENTS

ATTACK

OVPIs à l'horizon ! le jour le plus long commence pour les braves attaqués traitreusement par des biplans et autres antiquités d'avant-garde.

Ce programme vous permet de tester vos réflexes et vous dit si vous avez le compas dans l'oeil. Il s'agit en effet d'un programme de jeu où vous dirigez un char et où vous devez tirer sur des objets volants parfaitement identifiés puisqu'il s'agit d'une montgolfière, d'un vieux multiplan, d'un avion de chasse moderne, d'un vaisseau spatial de conception archaïque puis enfin d'une soucoupe volant dernier cri. Vous contrôlez le déplacement du char avec le joystick et vous faites feu avec la touche se trouvant sur le joystick. Vous pouvez rater l'objet passant au-dessus de votre tête mais, si vous le laissez vivre trop longtemps, il vous détruira d'un seul coup car il dispose d'une visée sûre à 100%. Quand l'ordinateur jugera que vous avez marqué suffisamment de points, il vous octroiera généreusement un char supplémentaire. il y a 100

tableaux, chaque tableau correspond à un passage de chacun des OVPIs cités plus haut à une vitesse de plus en plus grande. Quand vous aurez descendu tous les tableaux, vous reviendrez au premier tableau c'est-à-dire à la vitesse la plus lente mais alors, vous n'aurez plus droit à l'erreur, tout OVPI manqué rien qu'une fois vous détruira automatiquement. L'architecture du programme est assez simple, tout d'abord une procédure d'initialisation puis une routine pour générer les OPVIs, une procédure de tir et de déplacement du char, une procédure de test et de destruction des OPVIs et enfin une procédure de destruction du char. Mais n'oublions pas la routine permettant le calcul et l'affichage du score. Allez, descendez les tois, défendez votre planète contre les envahisseurs venus d'ailleurs en montgolfière. ■

Gilles LIONNET



```

100 REM ATTACK
110 CALL CLEAR :: CALL SCREEN(2)
120 CALL CHAR(96,"3F40838786868E",97,"8F9F9C9C9C80403F",98,"FF00C0E060606070",99,"FOF83838380000FF")
130 CALL CHAR(100,"FF001F1F01010101",101,"01010101010000FF",102,"FF00F8F880808080",103,"80808080800000FF")
140 CALL CHAR(104,"FF000F1F18181818",105,"181818181F0F00FF",106,"FF00F0F81800000",107,"00000018F8F000FF")
150 CALL CHAR(108,"FF00181818191B1F",109,"1E1F1B19181800FF",110,"FC023171E1C18101",111,"010181C1F17102FC")
160 CALL CHAR(112,"FF0003070606060E",113,"0F1F1C1C1C0000FF")
170 DATA 96,97,98,99,100,101,102,103,100,101,102,103,112,113,98,99,104,105,106,107,108,109,110,111

```

```

180 RESTORE 170 :: L=1 :: C=12 :: FOR COU=9 TO 11 :: CALL COLOR(COU,11,1):: NEXT COU
190 FOR T=1 TO 6 :: READ CAR1,CAR2,CAR3,CAR4
200 CALL HCHAR(L,C,CAR1):: CALL HCHAR(L+1,C,CAR2):: CALL HCHAR(L,C+1,CAR3):: CALL HCHAR(L+1,C+1,CAR4):: C=C+2 :: NEXT T

210 CALL CHAR(64,"00000000"):: FOR COU=0 TO 8 :: CALL COLOR(COU,8,1):: NEXT COU
220 DISPLAY AT(12,1)BEEP:"VOULEZ-VOUS DES EXPLICATIONS" :: DISPLAY AT(16,12):"(O/N) ?"
230 CALL KEY(5,KEY,STA):: IF STA=0 THEN 230
240 IF KEY=78 OR KEY=110 THEN 440
250 IF KEY=79 OR KEY=111 THEN 260 ELSE 220
260 GOSUB 1480 :: RESTORE 270
270 DATA "*LE JEU CONSISTE A REPOUSSER","L'ATTAQUE DES ENNEMIS.LEUR A","LTITUDE DE VOL EST ALEATOIRE"
280 DATA "*NE LES MANQUEZ PAS CAR EUX","NE VOUS MANQUERONT PAS.,"*ATTENTION LEUR VITESSE DE"
290 DATA "DEPLACEMENT AUGMENTE AU FUR","ET A MESURE QUE VOUS PASSEZ","LES TABLEAUX SUCCESSIFS."
300 FOR LI=5 TO 21 STEP 2 :: READ LI$ :: DISPLAY AT(LI,1):LI$ :: NEXT LI
310 GOSUB 1460 :: GOSUB 1480 :: RESTORE 320
320 DATA "*LES POINTS SONT EN RAPPORT","AVEC LA VITESSE,L'ALTITUDE","ET LE NOMBRE DE PASSAGES DE","L'ENNEMI."
330 DATA "*TOUS LES 10 TABLEAUX VOUS","GAGNEZ UN CHAR SUPPLEMENTAI-","-RE.,"*SI VOUS PARVENEZ AU TABLEAU"
340 DATA "100,LA VITESSE DE DEPLACE-","-MENT REDEVIENT CELLE DU DE-","PART,MAIS ATTENTION L'ERREUR"
350 DATA "N'EST PLUS PERMISE : (VOUS","JOUEZ ALORS EN FORCE 3) !!!"
360 FOR LI=7 TO 19 STEP 2 :: READ LI$ :: DISPLAY AT(LI,1):LI$ :: NEXT LI
370 GOSUB 1460 :: GOSUB 1480 :: FOR LI=8 TO 18 STEP 2 :: READ LI$ :: DISPLAY AT(LI,1):LI$ :: NEXT LI
380 GOSUB 1460 :: GOSUB 1480 :: RESTORE 390
390 DATA "ààààà*POUR OBTENIR UNE PAUSE","ATTENDRE LE SIGNAL SONORE ET","APPUYER SUR P.,"ààààà*APPUYER SUR N'IMPORTE"
400 DATA "QUELLE TOUCHE SAUF P POUR","REPARTIR","","ààààà* BONNE CHANCE *"
410 FOR LI=6 TO 20 STEP 2 :: READ LI$ :: DISPLAY AT(LI,1):LI$ :: NEXT LI :: GOSUB 1460
420 DATA "DESIREZ VOUS JOUER SOUS","àà1-CIEL ETOILE","àà2-UNE NUIT NOIRE",""
430 DATA "DEGRE DE DIFFICULTE ?","àà1- FACILE","àà2- MOYEN","àà3- DIFFICILE","" ,"àààààENTER POUR VALIDER"
440 GOSUB 1480 :: RESTORE 420 :: FOR LI=6 TO 24 STEP 2 :: READ LI$ :: DISPLAY AT(LI,2):LI$ :: NEXT LI
450 ON WARNING NEXT :: ACCEPT AT(6,26)SIZE(1)VALIDATE("12")BEEP:CIEL
460 ACCEPT AT(14,24)SIZE(1)VALIDATE("123")BEEP:FORCE
470 IF FORCE=1 THEN NPASS1=6
480 IF FORCE=2 THEN NPASS1=3
490 IF FORCE=3 THEN NPASS1=1
500 NPASS=NPASS1
510 CALL CLEAR
520 CALL CHAR(136,"00000000FFFFFFFF"):: CALL CHAR(137,"FFFFFFFFFFFFFFFF")
530 CALL COLOR(14,3,1):: CALL HCHAR(22,4,136,28):: CALL HCHAR(23,4,137,28):: CALL HCHAR(24,4,137,28)
540 DISPLAY AT(1,2):"SCOREà000000ààHIàSCOREà00000"
550 DISPLAY AT(23,21):"CHARS 5 " :: IF CIEL=2 THEN 620 ELSE RESTORE 560
560 DATA 16,40,16,40,64,16,64,72,16,88,78,16,110,60,16,144,80,16,139,115,16,40,136,16
570 DATA 64,128,16,48,176,16,72,168,16,80,192,16,104,200,16,128,192,16,24,96,9,18,164,13,26,216,11
580 DATA 56,32,5,72,232,10,96,24,7,104,136,14,144,40,8,152,160,12,140,232,4
590 ET$="0000000000000000103010000000000000000000000000000000000000000000000000000000"
600 CALL CHAR(128,ET$):: CALL MAGNIFY(3)
610 FOR NU=5 TO 28 :: READ LG,CL,COUL :: CALL SPRITE(LNU,128,COUL,LG,CL):: NEXT NU
620 TK$="01010101010101010F7FFFFFFFFF300000000000000E0FCFEFEFEFEFEFE18"
630 AV$="0000001F1F003FBF3F000C0C000000000000000868606FEFEFE00000000000000"
640 BL$="0000000000000000000000000000101010100000000000000000000000000000"
650 SC$="00000001033FFFDDDDFF7F07000000000000080C0FCFFDDDDFFEE000000000000"

```