

Jumpman, le saute-fantômes

Toujours atteint d'une maladie rare appelée la "pacmanie aiguë", je vous présente ici une nouvelle version du jeu de saute-mouton...

Le jeu est simple : au début les jumpmen sont à gauche, les fantômes à droite, le but est d'inverser cette situation. Pour ce faire, lire la suite de l'article (si, si !).

Le terrain est représenté par dix cases numérotées de gauche à droite et de 0 à 9 (une barre horizontale représente une case vide).

Les jumpmen ne peuvent se déplacer que de gauche à droite et les fantômes de droite à gauche ; voici (pour le jumpman) les mouvements autorisés (et les seuls !):



Lorsque les jumpmen occupent les cases 7 à 9 et les fantômes les cases 0 à 2, le jeu est fini ; répondez 10 au point d'interrogation vous demandant un numéro de case.

L'architecture du programme à présent :

- Lignes 1 à 3 : c'est toujours nous...
- Lignes 5 à 15 : initialisation et création du terrain
- Ligne 16 : impression du terrain
- Ligne 17 : entrée du mouvement
- Lignes 18 à 202 : analyse de la validité du coup ; s'il est valable, l'exécuter

```

1:REM (C) Daniel
      GLAZMAN
2:REM (C) TRACE
3:REM PARIS 83
5:WAIT 0: CLEAR :
  RESTORE
10:DATA "7E3B7F7F
      3B7E0000000000
11:DATA "40404040
      404040400000
12:DATA "1C3E7F7F
      772200000000
13:DIM A$(2)*21:
  READ A$(2),A$(
    0),A$(1)
14:DIM A(9):FOR I
  =1 TO 3:A(1)=1:
  NEXT I
15:FOR I=6 TO 8:A(
  I)=2:NEXT I
16:CLS :FOR I=0 TO
  9:GPRINT A$(A(
  I));:NEXT I
17:CURSOR 18:
  INPUT A:IF A=1
  0GOTO 1000
18:W=W+1:IF A(A)=
  0GOTO 16
19:GOTO 100*A(A)
100:IF A(A+1)=0LET
  A(A+1)=1:A(A)=
  0:GOTO 16
101:IF A(A+2)=0AND
  A(A+1)<>0LET A
  (A)=0:A(A+2)=1
  :GOTO 16
102:GOTO 16
200:IF A(A-1)=0LET
  A(A-1)=2:A(A)=
  0:GOTO 16
201:IF A(A-2)=0AND
  A(A-1)<>0LET A
  (A)=0:A(A-2)=2
  :GOTO 16
202:GOTO 16
1000:FOR I=0 TO 2:
  IF A(I)=0:
  GOTO 2000
1001:NEXT I
1002:FOR I=7 TO 9:
  IF A(I)=0
  GOTO 2000
1003:NEXT I
1010:WAIT :
  PRINT "En";W
  :;" coups...
1011:GOTO 5
2000:CLS :WAIT 75
  :PRINT "Vous
  moquez-vous
  de moi ?
2001:WAIT 0:GOTO
  16
  
```

Lignes 1000 à 2000 : en cas de fin de partie

Et voilà... En sachant qu'il suffit de préciser à votre poquette la case où se trouve le jumpman (ou le fantôme) pour que celui-ci se déplace automatiquement, amusez-vous bien...

Daniel Glazman

Voici les 2 découvertes que j'ai faites :

1) Gain de 197 octets :

Il suffit de faire :

- POKE &7866,0
- POKE &7868,0

MEM affichera alors (sans extension) :
2047

NEW ne changera rien mais NEW 0 redonnera MEM=1850.

2) Il est bien connu que l'on peut protéger ou déprotéger un programme par LOCK et UNLOCK ce qui n'empêche le listage que si le mode est différent de PRO, et l'exécution que si il est différent de RUN. Faire LOCK en mode "Réserve" empêche donc les 2 possibilités mais laisse encore à l'utilisateur la faculté de détruire les touches préprogrammées... Il existe une autre façon d'empêcher le listage et l'exécution grâce à un POKE bien placé :

POKE &7867,63 (ou tout nombre positif et inférieur à 63) ;
LIST et RUN ne donneront plus rien ;

MEM retrouvera des valeurs très grandes (si l'on fait POKE &7867,0 on aura MEM=18409), mais ces indications sont fantaisistes.

En effet, l'utilisateur sera dans l'impossibilité de programmer (tout essai donnera ERROR 13 bien que l'on ait MEM > 0)

Pour rappeler le programme il suffira de faire POKE &7867,64 (même le pointeur est gardé intact).

Exemple :

```

Mode "PRO" NEW 0
10 PRINT "O.K., CA MARCHE"
RUN donnera ce que l'on attend.
  
```

Si l'on repasse en mode "PRO" et que l'on appuie sur [↓] ou [↑] on verra que le signe - est placé après le deuxième guillemet.

Faire POKE &7867,63 puis [↓] ou [↑] : rien n'apparaît ! De plus, MEM=2281 (!!!) et RUN ne donne rien. Faire POKE &7867,64 puis RUN et "O.K., CA MARCHE" apparaît sur l'afficheur.

Cette astuce plus un LOCK sur RUN, laisse seulement à l'utilisateur la possibilité d'utiliser le PC.2 en calculette...

Paul Courbis