

Un pot commun pour toutes les machines

Le nombre secret de la TI 57

□ Dans cette version du *Nombre secret*, le nombre que la machine vous invite à deviner est compris entre 0 et 100. Ce nombre aléatoire est "fabriqué" dans la petite loterie mathématique du sous-programme Lbl 0 aux pas 33 à 46. Comme sur les autres ordinateurs de poche, le jeu consiste à deviner quel est le nombre secret en proposant une série de fourchettes (50-80 par exemple). A chaque essai, la machine indique si la fourchette

. contient le nombre à deviner : affichage 2
. est trop basse : affichage -1

```

NBR SECRET
AUTEUR : BRUNO GARREAU
COPYRIGHT L'ORDINATEUR
DE POCHE ET L'AUTEUR
*****
0 81 R/S
1 32 1 STO 1
2 32 7 STO 7
3 81 R/S
4 32 2 STO 2
5 66 2ND X = T ?
6 51 1 GTO 1
7 -76 INV 2ND X >= T ?
8 51 2 GTO 2
9 33 4 RCL 4
10 22 X ECHANGE T
11 76 2ND X>=T
12 51 3 GTO 3
13 33 2 RCL 2
14 -76 INV 2ND X >= T ?
15 51 4 GTO 4
16 02 2
17 71 RST
18 86 1 2ND LBL 1
19 33 4 RCL 4
20 -66 INV 2ND X=T ?
21 03 3
22 51 9 GTO 9
23 86 2 2ND LBL 2
24 15 CLR
25 71 RST
26 86 4 2ND LBL 4
27 01 1
28 84 +/-
29 71 RST
30 86 3 2ND LBL 3
31 01 1
32 71 RST
33 86 0 2ND LBL 0
34 55 MULTIPLIE PAR
35 06 6
36 07 7
37 85 =
38 -49 INV 2ND INT
39 55 MULTIPLIE PAR
40 02 2
41 -18 INV 2ND LOG
42 85 =
43 49 2ND INT
44 32 4 STO 4
45 15 CLR
46 71 RST

```

. est trop haute : affichage 1
. ou si elle n'a pas de sens (le joueur a proposé, par distraction ou par malice, une fourchette dont la borne haute est plus petite que la borne basse), l'affichage indique alors 0.

Le joueur propose autant de fourchettes qu'il le désire jusqu'au moment où il pense avoir découvert la solution. Il ne lui reste plus qu'à proposer une "fourchette" dont les deux extrémités sont identiques (il demande en quelque sorte : "le nombre est-il compris entre 23 et 23 ?"); la machine lui répond 3 s'il s'est trompé et fait clignoter la bonne solution à l'affichage s'il a deviné juste.

Après avoir déclenché la génération du nombre au hasard (afficher un nombre compris entre 0 et 1 et faire SBR 0), le joueur introduit au clavier la borne basse de sa fourchette. En appuyant sur R/S (pas 00), il conduit le programme à engranger ce nombre dans les mémoires 1 et 7 (pas 01 et 02). Il introduit alors la borne haute qui est stockée dans la mémoire 2 au pas 04. Le programme examine ensuite si ces deux bornes sont égales : $x = t$? du pas 05 (1). Si c'est bien le cas, c'est que le joueur a proposé une solution : il pense avoir deviné le nombre secret et le programme se poursuit au sous-programme Lbl 1 (GTO 1 du pas 06) où l'on regarde si cette solution est différente de la bonne (RCL 4, $x \neq t$ des pas 19 et 20). Si la solution est différente, c'est le pas 21 qui est exécuté et l'affichage indique 3 : le joueur s'est trompé. Sinon, on passe au pas 22 où l'instruction impossible GTO 9 provoque le clignotement de l'affichage : le joueur a gagné.

Au pas 07, on regarde si la borne haute qui se trouve à l'affichage (registre x) est plus petite que la borne basse, auquel cas le joueur a proposé une fourchette impossible et le programme se poursuit à l'étiquette 2 (GTO 2 du pas 08); on efface l'affichage et l'on redemande

(1) Sur le fonctionnement des tests de la TI 57, le lecteur pourra se reporter aux explications qui sont fournies dans l'article des pages 45 à 49.

une fourchette : CLR et RST. Si la fourchette est plausible, on place le nombre secret dans le registre 7 (RCL 4 et $x > t$ des pas 10 et 11) et l'on compare la borne basse (à l'affichage) et le nombre secret : $x \geq t$. Si la borne basse est supérieure ou égale à la bonne solution, on se rend au pas 30 (GTO 3) pour afficher 1 et l'on retourne au début du programme pour l'introduction d'une nouvelle fourchette. Dans le cas contraire, on rappelle la borne haute à l'affichage (RCL 2 du pas 13) et l'on examine si elle est plus petite que la solution, $x < t$? du pas 14. La borne haute est-elle bien plus

Utilisation du programme

- 1 - afficher un nombre quelconque compris entre 0 et 1 et faire SBR 0
 - 2 - introduire la limite inférieure de la fourchette et presser sur R/S.
 - 3 - introduire la limite supérieure de la fourchette et presser sur R/S.
 - 4 - 0 à l'affichage : la fourchette n'a pas de sens, on appuie sur CLR et on recommence à l'étape n° 2.
 - 1 à l'affichage : la fourchette est trop haute, on appuie sur CLR et l'on recommence en 2.
 - 1 à l'affichage : la fourchette est trop basse, on appuie sur CLR et on repart en 2.
 - 3 à l'affichage : la solution proposée était fautive. On appuie sur R/S et l'on est en 2. La solution que l'on a proposée clignote à l'affichage : on a découvert le nombre secret.
- Pour une autre partie, reprendre à l'étape n° 1.

petite que la solution ? Si oui, c'est le pas 15 qui est exécuté et l'on passe donc au sous-programme 4 des pas 26 à 29 qui affichent -1 et renvoient au début du programme.

Lorsque la réponse à tous ces tests a été négative, on est certain que le nombre secret est compris à l'intérieur de la fourchette : les pas 16 et 17 affichent 2 et renvoient en 00 pour l'introduction d'une nouvelle fourchette.

□ Bruno Garreau