

Stage algorithmique 1

TI graphiques (82, 83, 84)

Algorithmes mystérieux

Après s'être familiarisé avec les algorithmes et la (ou les) syntaxe(s) correspondant aux logiciels ou calculatrices dont on dispose, l'élève devra savoir lire un petit programme. Au début, on peut proposer des petits programmes simples et courts.

Premier algorithme mystère

Que fait ce programme ?

Saisir(x) Saisir(y) $z := x$ $x := y$ $y := z$ Afficher(x) Afficher(y)	Facile : permutation des valeurs contenues dans x et dans y .
--	---

Deuxième algorithme mystère

Que fait ce programme ? Le fait-il dans toutes circonstances ?

Saisir(x) Saisir(y) $x := x + y$ $y := x - y$ $x := x - y$ Afficher(x) Afficher(y)	Ce programme, plus compliqué, se contente aussi de permuter les valeurs contenues dans x et dans y ... sauf dans le cas où y , par exemple, est très petit devant x de telle sorte que $x + y \approx x$...
--	--

Il est intéressant de montrer que l'on eut obtenir le même résultat avec des algorithmes différents. Pour les différencier, on peut effectivement examiner le cas où $x = 5$ et $y = 10^{-13}$. Dans le premier algorithme, il y a bien échange ; dans le second, on obtient 0 et 5.

Troisième algorithme mystère

Cet algorithme est proposé sous forme de programme en langage machine graphique TI.
Combien de nombres vont-ils être affichés ? Quelle est la suite de nombres ainsi obtenue ?

$1 \rightarrow A$ $1 \rightarrow B$ For(I, 1, 6) $A + B \rightarrow C$ $B \rightarrow A$ $C \rightarrow B$ Disp C End	Il y a six passages (For de 1 à 6), donc six nombres affichés. Ce sont 2, 3, 5, 8, 13, 21. On obtient effectivement les termes successifs de la suite de Fibonacci. On pourra prolonger en compliquant le programme (affichage de la suite des quotients successifs, par exemple, pour obtenir des valeurs de plus en plus proches du nombre d'or).
--	--

Le professeur pourra, s'il le souhaite et si sa classe s'y prête, proposer ce dernier programme en langage TI-Nspire :

```
Prgm
Local  $a, b, c, i$ 
 $a:=1$ 
 $b:=1$ 
For  $i, 1, 6$ 
 $c:=a+b$ 
 $a:=b$ 
 $b:=c$ 
Disp  $c$ 
EndFor
EndPrgm
```