Fiche

Un algorithme est la décomposition d'une action en instructions élémentaires.
Les calculs nécessaires à la résolution d'un problème, les consignes pour un tracé géométrique, les étapes pour trier des données constituent des algorithmes.
L'énoncé en français doit être traduit en langage machine pour effectuer un traitement sur calculatrice ou ordinateur.

1. Quelles sont les instructions d'un algorithme élémentaire ?

• Un algorithme comporte quatre étapes :

- indication des variables : une variable sert à stocker une valeur numérique ou un mot. Il faut déclarer les données de l'énoncé et désigner les variables qui permettront de stocker les résultats intermédiaires lors du traitement des instructions ;
- début : on affecte les données dans des variables, on initialise le stockage des valeurs intermédiaires et les compteurs numériques ;
- traitement : on note l'ensemble des instructions de calcul ou de tri, les boucles de répétition et les tests à effectuer ;
- sortie : lorsque toutes les opérations sont exécutées il faut communiquer les résultats, c'est-à-dire les écrire.

Exercice n°1

Exercice n°2

Exercice n°3

2. Comment insérer une condition ?

• Deux traitements sont proposés, suivant que la condition est vérifiée ou non.
La structure de l'algorithme est alors de la forme :

- indication des variables ;
- entrée des données ;
- traitement.

Si condition, alors traitement 1.
Sinon traitement 2.
FinSi
Sortie des résultats : l'instruction FinSi indique la fin du traitement conditionnel.

Exercice n°4

3. Comment insérer une boucle ?

• Il y a trois structures possibles :

- avec un compteur : pour I variant de I_0 jusqu'à N, faire traitement. FinPour
- avec une condition : tant que condition, faire traitement. FinTantque
- avec répétition jusqu'à condition : répète traitement jusqu'à condition. FinRepète

Les instructions FinPour, FinTantque et FinRepète terminent la boucle.

Exercice n°5

Exercice n°6

4. Comment résoudre un système de deux équations du premier degré ?

• Le système $\begin{cases} Ax + By = C \\ Dx + Ey = F \end{cases}$ peut avoir un couple solution, aucune solution ou une infinité de solutions.

Lorsque les coefficients A, B, D et E ne sont pas proportionnels, les deux droites représentatives sont sécantes. Il y a un seul couple solution, composé des coordonnées du point d'intersection.

Dans le cas contraire, si les nombres A, C, D et F sont proportionnels, on obtient deux équations d'une même droite, il y a une infinité de solutions. Sinon les deux droites sont strictement parallèles et il n'y a aucune solution.

Algorithme

Variables

A, B, C, D, E, F,
X, Y les valeurs du couple solution éventuel.

Entrée

? → A ; ? → B ; ? → C
? → D ; ? → E ; ? → F

Traitement

Si A × E ≠ B × D
Alors
$\frac{CE-BF}{AE-BD} \rightarrow X$
$\frac{AF-CD}{AE-BD} \rightarrow Y$
Écrire « X = »
Afficher X
Écrire « Y = »
Afficher Y
Sinon
Si A × F ≠ C × D
Alors écrire « PAS DE SOL »
Sinon
Écrire « INFINITE DE SOL »
Alors écrire « PAS DE SOL »
FinSi

Sortie

Écrire « FIN »

Traduction de la calculatrice :

Texas instrument	Casio
Input A : Input B : Input C	? → A : ? → B : ? → C
Input D : Input E : Input F	? → D : ? → E : ? → F
If A × E ≠ B × D	A × E − B × D → P
Then	A × F − C × D → Q
(C × E − B × F) ÷ (A × E − B × D) → X	If P ≠ 0
(A × F − C × D) ÷ (A × E − B × D) → Y	Then "X ="
Disp "X =" : Disp X	(C × E − B × F) ÷ P → X △
Disp "Y =" : Disp Y	"Y ="
Sinon	(A × F − C × D) ÷ P → Y △
If A × E ≠ C × D	Else If Q ≠ 0
Then	Then "PAS DE SOL"
Disp "PAS DE SOL"	Else "INFINITÉ DE SOL"
Sinon	IfEnd
Disp "INFINITÉ DE SOL"	IfEnd

End	"FIN"
End	Stop
Disp "FIN"	

Exercice n°7

5. Comment déterminer l'extremum d'une fonction ?

• Recherche du minimum

Sur l'intervalle [A ; B], on cherche les coordonnées du point le plus bas de la courbe représentative d'une fonction *f*.

Pour cela, on balaye l'intervalle avec un pas de 10^{-N} et on calcule à chaque fois l'image obtenue.

On la compare à la plus petite image obtenue précédemment. Si elle est encore plus petite, elle prend sa place dans la variable.

Algorithme

Variables

X (borne inférieure de l'intervalle)
B (borne supérieure de l'intervalle)
N (exposant de la précision)
f la fonction à étudier

Entrée

? → A
? → B
? → N
f(A) → M

Traitement

Tant que X < B faire X + 10^−N → X
Si *f* < M
Alors *f*(X) → M
Fin Si
Fin Tant que

Sortie

Afficher M

Traduction de la calculatrice :

Texas Instrument	Casio
Input A	? → A
Input B	? → B
Input N	? → N
Y1 (A)→ M	A → X
A → X	Y1 → M
While X < B	While X < B
X + 10^(−N) → X	X + 10^(−N) → X
If Y1(X) < M	If Y1 < M
Then	Then Y1 → M
Y1 (X) → M	IfEnd
End	WhileEnd
End	M △

Dips M	Stop

Recherche du maximum

Il faut remplacer l'instruction If Y1 < M par If Y1 > M.

Exercice n°8

6. Comment démarrer avec le langage Python ?

Types de variables :
Quelques types de variables abordés au lycée avec le language Python:
int : nombre entier ; float : nombre flottant ; str : chaîne de caractères ; bool : booléen
Pour déterminer le type d'une variable on peut taper la commande type dans le Shell.
type(3) va renvoyer int ; type(1/3) va renvoyer float
type(« ASP ») va renvoyer str (sans les guillemets ASP est alors le nom d'une variable mais si elle n'a pas de valeur affectée alors type(ASP) va renvoyer un message d'erreur
type(True) va renvoyer bool
Fonctions :
Une des fonctionnalités du langage Python est la possibilité de créer des fonctions. L'avantage est que une fois crée on peut réutiliser une fonction dans le même programme ou bien ultérieurement en utilisant un copier-coller. L'objectif de la création de fonctions est de gagner du temps et de pouvoir les utiliser directement dans le Shell.
Pour définir une fonction on utilise les commandes def et return.
Voici par exemple la fonction carré :

```
def carré(x) :
    return x*x
```

Bien sûr, pour des fonctions de base le gain de temps est très faible.
Le module random
Les modules contiennent des fonctions que l'on est amené à réutiliser souvent (on les appelle aussi bibliothèques ou *libraries*).
Le module random de Python donne accès aux nombres aléatoires
Pour pouvoir utiliser les fonctions du module random il faut en introduction du programme écrire :

```
def f(x):
    return x**2

max=0

for i in range(0,401):
    if f(i*0.01)>max:
        max=f(i*0.01)

print("Le maximum de f sur [0,4] est environ",max)
```

from random import* (Le symbole * permet d'importer toutes les fonctions du module random).
On peut par exemple simuler la génération d'un entier compris entre 1 et 6 (utile pour simuler le lancer d'un dé cubique) en tapant : random.randint(1,6)
Algorithme : Déterminer une valeur approchée du maximum de la fonction carré sur l'intervalle [0;4] par balayage.

À retenir

• Ne pas oublier de stocker les résultats intermédiaires et les compteurs dans des variables.
Ces variables doivent être initialisées.

• L'instruction FinSi permet de repérer la fin d'une condition. Les instructions Fin tant, FinPour et FinRepeat permettent de repérer les fins de boucles.
Lorsqu'une condition est imbriquée dans une boucle, un décalage des instructions de la condition vers la droite permet de mieux lire l'algorithme.

• Les algorithmes donnant le minimum ou le maximum d'une fonction définie sur un intervalle, le nombre de solution(s) d'un système d'équations du premier degré est à transposer et conserver sur la calculatrice.

© 2000-2024, rue des écoles