

## 1 . Qu'est-ce qu'un algorithme

Un algorithme est une suite finie d'opérations élémentaires, à appliquer dans un ordre déterminé, à des données.

Les trois phases d'un algorithme sont :

- 1 . l'entrée des données
- 2 . le traitement des données
- 3 . la sortie du résultat

### Exercice 1

Choisir un nombre x.  
Le multiplier par 2.  
Ajouter 3 au résultat.  
Elever le résultat au carré

- a) Appliquez ce programme de calcul, en donnant à x les valeurs 5 , - 1 et 100.
- b) Pouvez-vous choisir un nombre qui donne en sortie 0 ? 100 ?
- c) Donnez l'expression de la fonction correspondant à ce programme de calcul

## 2 . Variables et affectation

▪ Dans tout algorithme, on commence par l'**entrée des données** nécessaires au traitement.

Chacune de ces données est stockée à un emplacement nommé **variable** et repérée par un nom.

▪ Dans le déroulement de l'algorithme, il s'avère souvent nécessaire d'utiliser de nouvelles variables : pour des calculs intermédiaires, pour fournir des données en sortie, etc.

Les variables peuvent contenir différents types de données : des nombres, mais aussi des listes, des chaînes de caractères (notamment pour l'affichage de messages ),...

▪ En résumé, les instructions de base que l'on peut pratiquer avec une variable sont :

- **La saisie (appelée aussi instruction d'entrée)**: on demande à l'utilisateur de donner une valeur à une variable ;
- **L'affectation** : on donne à la variable le résultat d'un calcul, d'une suite d'instructions ;
- **L'affichage (appelé également instruction de sortie)** : On affiche le contenu de la variable

### Exercice 2

On considère l'algorithme suivant :

Lire X  
Lire Y  
a reçoit X+Y  
b reçoit X-Y  
c reçoit a×b  
Afficher c

1. Identifiez les parties « Entrée », « Traitement » et « Sortie » de cet algorithme.
2. Identifiez les différentes variables utilisées et leurs utilisations (Saisie –affectation- Affichage)
3. Pouvait-on faire l'économie de variables ?

## 3 . L'instruction conditionnelle

▪ La résolution de certains problèmes conduit parfois à une situation dans laquelle la décision prise est soumise à condition :

- Si la condition est vérifiée, on effectue une tâche précise
- Si elle n'est pas vérifiée, on effectue une autre tâche .

▪ Cela se traduit dans un algorithme par le schéma suivant :

▪ Le « Sinon » n'est pas systématique.

Si **condition**  
alors **tâche 1**  
sinon **tâche 2**  
FinSi

### Exercice 3

Ecrire un algorithme qui précise l'appartenance (ou non) d'un point, choisi par l'utilisateur, à la parabole d'équation  $y=x^2 - 5x + 6$ .

### 4. La boucle itérative

Dans certains programmes, on est parfois amené à réaliser plusieurs fois de suite la même tâche.

Si l'on sait le nombre de fois **N** que l'on doit exécuter notre tâche, on exécute ce qu'on appelle une boucle à l'aide de l'instruction suivante :

```
Pour i allant de 1 à N
    faire tâche
FinPour
```

Avec cette instruction, on répète alors un nombre de fois (fixé) notre tâche : Ici, pour i allant de 1 à N soit N fois en tout.

La variable i est appelé un compteur.

### Exercice 4

Ecrire un algorithme qui permet de calculer la somme des N premiers entiers naturels non nuls.

### 5. La boucle conditionnelle

Dans certains programmes, on est parfois amené à réaliser plusieurs fois la même tâche mais sans savoir combien de fois celle-ci devra être réalisée.

En fait, on doit répéter une même instruction **tant qu'**une certaine condition est remplie.

On utilise alors une boucle conditionnelle : le passage dans cette boucle s'arrête lorsque la condition n'est plus remplie.

Les instructions sont alors les suivantes :

```
Tant que condition
    faire tâche
Fin Tantque
```

#### **Remarques :**

- Il est possible de ne jamais rentrer dans une boucle conditionnelle : en effet, si la condition n'est jamais réalisée.
- Attention, lors de l'écriture d'un algorithme avec une boucle conditionnelle, de bien vérifier que la boucle se termine !

### Exercice 5

Voici un algorithme contenant une boucle conditionnelle :

```
Saisir A
n prend la valeur de 0
Tant que  $n^2 < A$ 
    Afficher n
Fin Tantque
```

Malheureusement, lorsque j'ai rentré mon programme sur ordinateur, celui-ci m'a affiché un message d'erreur.

Trouve quelle est cette erreur.

**Exercice 4 -**

**#Algorithme qui permet de calculer la somme des N premiers entiers naturels non nuls.**

Demander N entier naturel

S = 0

Pour i allant de 1 à N

S = S + i

FinPour

Afficher ( 'La somme des entiers de 1 à', N , 'est égale à', S )

**Exercice 5 – Erreur dans le programme**

