

# Chapitre 1

## Qu'est ce que programmer ?

### 1.1 Les algorithmes et leur structure

#### 1.1.1 Définition et propriété d'un algorithme

1. Un algorithme est une suite d'opérations élémentaires permettant s'obtenir le résultat final lié à un problème déterminé.
2. Un algorithme, dans des conditions d'exécution similaires, avec des données identiques, fournit toujours le même résultat

#### Remarque 1 :

1. Les algorithmes ont pour fonction de nous faire réfléchir, mais, à priori, pas de s'exécuter sur un ordinateur. L'algorithme décrit, sur papier, un traitement. Il est donc nécessaire d'en simuler le traitement.
2. Un algorithme peut très bien être décrit en langage ordinaire. Dans la plupart des cas, on utilise un langage plus précis, mieux adapté à la nature des instructions ultérieures; on appelle cela un **langage de programmation**, mieux adapté à la nature des instructions utilisées. Nous utiliserons, sur [mathinfovannes](#) le langage Python. Ce n'est, à proprement parler, pas un langage d'apprentissage. Cependant, il est massivement utilisé; nous nous adaptons donc
3. On peut assimiler les algorithmes à des **méthodes**, des **procédures** ou encore des **recettes de cuisine** ou un mode d'emploi pour monter des meubles
4. Un **algorithme** prend des données en entrée, exprime un traitement particulier et fournit des données en sortie.
5. Un algorithme ne dépend pas du langage dans lequel il est implanté, ni de la machine qui exécutera le programme correspondant
6. **Les besoins pour créer un algorithme :**
  - ▷ Savoir expliciter son raisonnement
  - ▷ Savoir formaliser son raisonnement
7. Un **programme** est une série d'instructions pouvant s'exécuter en séquence, ou en parallèle (*parallélisme matériel*) qui réalise (*implémente*) un algorithme

#### Remarque 2 :

##### Structure d'un algorithme

1. Il y a souvent des règles de programmation qu'il faut respecter; l'apprentissage de la programmation propose souvent 2 parties :
  - (a) Une première partie qui est un « bloc de déclarations »
  - (b) Une seconde partie qui est un « bloc d'instructions »

2. Que contiennent ces différents blocs ?
  - (a) Le **bloc de déclarations** contient le nom du programme, les définitions de variables sous forme de commentaires
  - (b) Le **bloc d'instructions** est une partie de traitement d'un algorithme, constituée d'opérations élémentaires.
3. En Python, il est très possible de s'affranchir de ces **blocs**. Même si nous nous en affranchissons (*on le verra dans la suite du cours*) il faudra bien savoir sur quoi nous travaillons, sur quelles données le programme devra agir.
4. **Un programme se doit d'être commenté.** Les commentaires sont utiles aux programmeurs qui veulent comprendre ou modifier l'algorithme. La complexité des algorithmes impose de les commenter judicieusement : ni trop, ni trop peu. En langage Python, un commentaire est toujours précédé de #

### Exemple 1 :

Ecrivons l'algorithme qui permet d'afficher « Bonjour tout le monde » ; c'est l'algorithme « Hello World » lequel est d'un classicisme béat.

```
#Algorithme "Bonjour tout le monde"
#Cet algorithme écrit à l'écran "Bonjour tout le monde"
>>> print("Bonjour tout le monde")
Bonjour tout le monde
>>>
```

### Remarque 3 :

Pour la clarté de sa lecture, il est bon [d'indenter le programme](#). L'indentation est essentielle et naturelle en langage Python