

# Les bases d'un algorithme

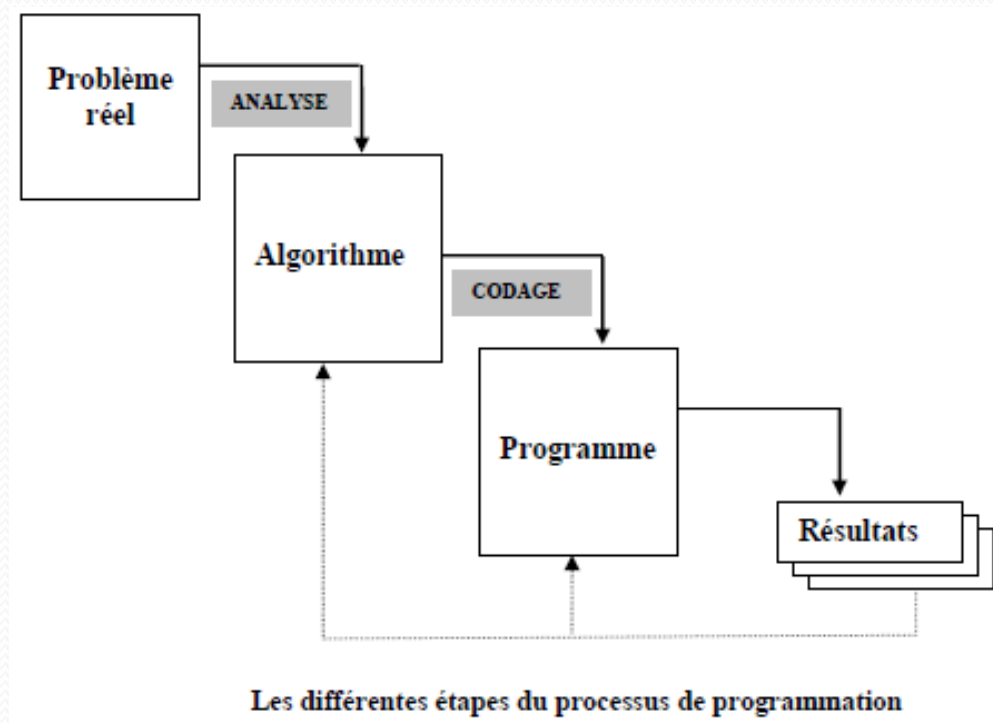
# Introduction

- Un ordinateur peut contenir plusieurs applications, par exemple: application de gestion de stock, suivi médical des patients dans un hôpital, gestion des factures,....
- Un ordinateur pour qu'il puisse effectuer des tâches aussi variées il suffit de le programmer.
- La programmation consiste, avant tout, à déterminer la démarche permettant d'obtenir, à l'aide d'un ordinateur, la solution d'un problème donné.

# Le processus de la programmation

- Le processus de la programmation se déroule en deux phases :
  1. On procède à ce qu'on appelle **l'analyse** du problème posé ou encore la recherche d'un **algorithme** qui consiste à définir les différentes étapes de la résolution du problème.
  2. on exprime dans un **langage de programmation** donné, le résultat de l'étape précédente.

# Schéma regroupant les différentes étapes de la programmation

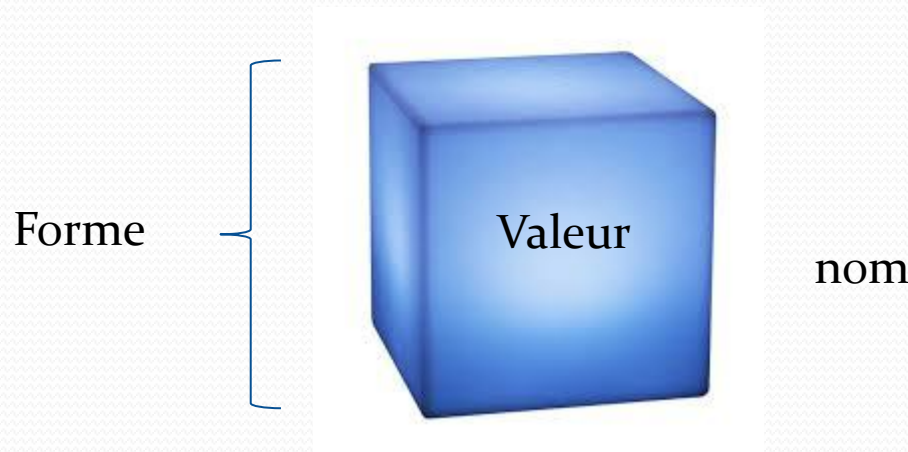


# Définition d'un algorithme

- Un algorithme est une **suite d'actions** que devra effectuer un ordinateur, dans un **ordre** bien précis, pour obtenir la **solution** d'un **problème** de départ.

# Les données

- Une donnée peut être considérée comme une boîte, portant une étiquette (**nom**), d'une certaine forme (**type**) et qui contient une information (**valeur**).



# Les types de données

- Une donnée possède un type qui peut être:
  1. Données numériques
  2. Données alphanumériques
  3. Données logiques

# Le type numérique

- Un type numérique caractérise les valeurs entières ou réelles:
  1. Entier: une données(variable ou constante) est dite entière s'elle prend des valeurs dans l'ensemble  $\mathbb{Z}$ .
  2. Réel: une données est dite réelle s'elle appartient à l'ensemble  $\mathbb{R}$ . (possède une partie entière et une partie fractionnaire)



# Le type alphanumérique

- Le type alphanumérique caractérise les valeurs caractères (noté par **char**) ou une chaîne de caractères (noté par **chaîne**).

1. **Char** : sa valeur est un caractère quelconque. Un caractère est toujours noté entre **quotes** (' ').

Exemples: 'o' ... '9', 'a' ... 'z', '£', '\$', ';', ...

Le caractère espace s'écrit: ' '

Le caractère vide s'écrit: ''

2. **Chaîne** : sa valeur est une suite finie de caractères quelconques. Une chaîne doit être délimitée par des guillemets ""

Exemples: "Bonjour", "AZERTY098Y/REA", ....

# Le type logique

- Une valeur logique ou booléenne est l'une des deux valeurs **VRAI** ou **FAUX** (**1** ou **0**).
- On utilise des valeurs logiques souvent lorsque on vérifie une condition (**a voir lus tard**)

# Les opérateurs arithmétiques

- Les opérateurs arithmétiques sont utilisés avec des variables de type numériques. Ils nous permettent de faire des calculs numériques.
  1. Addition : +
  2. Soustraction : -
  3. Multiplication : \*
  4. Division: /
  5. Modulo : c'est le reste de la division entière d'un nombre sur un autre le symbole c'est **mod**

**Exemple:  $12 \bmod 4 = 0$**

**$14 \bmod 4 = 2$**

**Remarque:** l'opérateur modulo est utilisé juste pour le type **Entier**

# Les opérateurs de comparaison

- Sont des opérateurs qu'on utilise pour comparer deux valeurs entières ou même caractère ou chaîne de caractères.
  1. Supérieur :  $>$
  2. Supérieur ou égal:  $>=$
  3. Inférieur:  $<$
  4. Inférieur ou égal:  $<=$
  5. Égal:  $=$
  6. Différent:  $!=$  ou bien  $<>$  ou  $\neq$

Exemple :  $16 > 12$  ou  $"Balon" < "Bonjour"$  ou  $'g' > 'f'$

**Remarque:** la comparaison entre les chaînes de caractères se fait selon l'ordre du dictionnaire.

# Les opérateurs logiques

- On va travailler avec 3 opérateurs logiques. Qui sont utilisés pour vérifier une expression logique (à voir par la suite ):
  1. La négation: **NON**
  2. L'intersection : **ET**
  3. L'union: **OU**

**Remarque:** vous pouvez revenir aux tables de vérité de ces trois opérateurs dans le module : l'essentiel...

# Les expressions

- Une expression est une **combinaison** de **variables** et des constantes à l'aide **d'opérateurs**. On peut distinguer deux types d'expressions:
  1. Expression arithmétique
  2. Expression logique

# Expression arithmétique

- Elles expriment un calcul. Est une combinaison de variables numériques à l'aide d'opérateurs arithmétiques:
- Exemple:  $X * 2 / (3 + y)$
- L'ordre selon lequel se déroule chaque opération est très important. On se sert de l'hierarchie des opérateurs :
  1.  $()$  parenthèses
  2.  $*$  / multiplication et division
  3.  $+$  - addition et soustraction

Exemple:

$a + b * c$  : on fait d'abord  $b * c$  puis on rajoute  $a$

# Expression logique

- Est une combinaison de **variables logiques** à l'aide **d'opérateurs logiques** ou une combinaison de variables **arithmétiques** à l'aide **d'opérateurs de comparaison**.
- **Exemples:**
  - $X > (y + 2)$
  - $(A \text{ ET } B) \text{ OU } C$
  - $X = 3 \text{ OU } y > 12$



# Choix des identificateurs

- Il est nécessaire de déclarer toute donnée utilisée dans un algorithme et de préciser son type.
- L'identificateur (ou nom ) de la variable est une suite de caractères alphanumérique qui doit commencer obligatoirement par une lettre et ne doit pas contenir des caractères spéciaux ou un espace.
- Exemples:

x, x1, ident3,....      Correctes

3y, x\$y,...      faux

# Action de déclaration

- Pour déclarer une variable on utilise la syntaxe suivante:

```
Var MaVariable : type
```

Où type est le type de la variable: **Entier**, **réel**, **char**, **chaine** ou **booléen**

Exemples:

```
Var x : Entier  
Var ok : booléen  
Var y : réel  
Var c : char  
Var ch : chaine
```

# Action d'affectation

- C'est l'action qui permet d'affecter ou d'assigner une valeur ou une expression à une variable.
- l'affectation s'effectue en se servant de d'une flèche  $\leftarrow$  ou  $:=$
- Syntaxe:

Identifiant  $\leftarrow$  expression ou valeur

Identifiant  $:=$  expression ou valeur

- Exemple:

$X \leftarrow 5$   
 $Y \leftarrow (x+3)$

# Action de lecture

- Une action de lecture ou d'entrée est une action qui permet d'introduire des données dans un programme. En utilisant généralement le clavier ou la souris
- Syntaxe:

**Lire(a)** : la valeur saisie va être stockée dans la variable a  
**Lire(a,b)** : les valeurs saisies vont être stockées successivement dans a et dans b

# Action d'affichage

- L'action d'affichage ou d'écriture va permettre d'afficher le résultat du programme à l'écran.
- Syntaxe:

**Afficher**("bonjour") : Le message Bonjour va être affiché à l'écran

**Afficher** (x): la valeur de la variable **x** va être affiché à l'écran

**Afficher** ("Le résultat est :", X): Le message est affiché suivi de la valeur de **X**

- **Remarque:** parfois on utilise le mot **Ecrire** au lieu de afficher