

algorithmme

# algorithme

- Un algorithme est donc un moyen de description des étapes à suivre pour résoudre un problème.

- Exemple: l'algorithme du café chaud.



Exemple: Comment construire une maison, les étapes à faire

## **Nous allons**

- Définir les besoins.
- Lister les opérations
- Ordonner la liste des opérations
  - L'étude de sol
  - Le terrassement
  - L'excavation
  - Les fondations
  - L'assainissement
  - Le soubassement
  - Le Plancher
  - L'élévation
  - La charpente
  - La toiture
  - Les menuiseries extérieures



- ❑ Pour résoudre un problème donné par l'informatique, l'utilisateur de l'ordinateur doit mettre au point un **programme** et le faire exécuter par la machine.
- ❑ L'ordinateur se chargera de traiter les instructions du programme et restituer les résultats demandés en fonction des données qui lui sont fournies.

- Malgré qu'il existe de nombreux langages de programmation, ils partagent tous le même point commun « la logique »
- La logique de programmation = l'algorithme
- L'algorithme est indépendant de toute architecture matérielle ou logicielle.
- Il n'est pas destiné à être exécuté par la machine

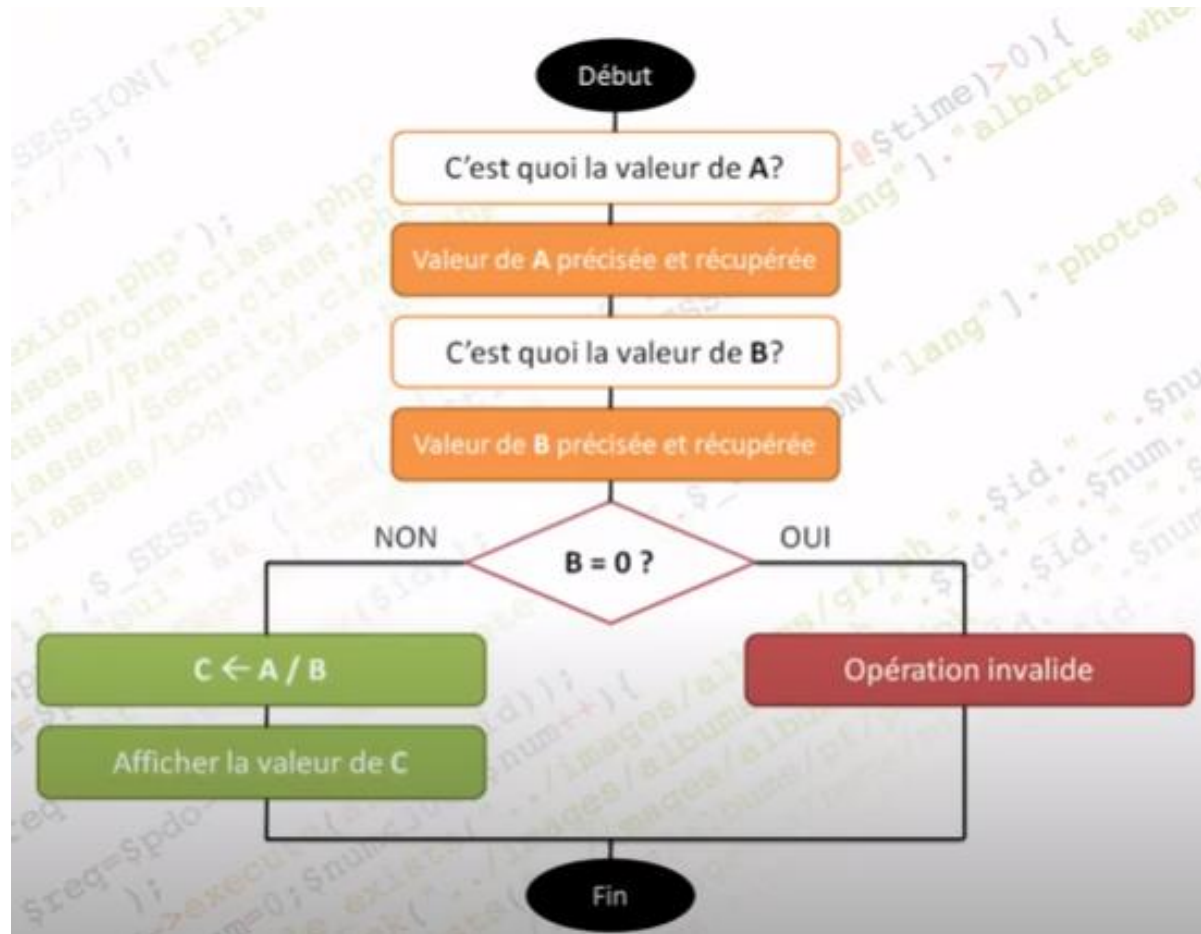
Un algorithme peut être représenté sous forme de :

- ❑ Organigramme

- ❑ Pseudo-code

**Le pseudo-code est plus utilisé car il est plus proche de la structure d'un vrai programme.**

# Organigramme





# Pseudo-code

```
1  Algorithme Division
2
3  Variables
4      A,B : Entier
5      C   : Réel
6
7  Début
8      Ecrire('Entrer la valeur de A: ')
9      Lire(A);
10     Ecrire('Entrer la valeur de B: ')
11     Lire(B);
12     Si B=0 Alors
13         Ecrire('Opération invalide!')
14     Sinon
15         C <- A/B;
16         Ecrire('Le résultat est: ',C)
17     FinSi
18 Fin
```

## Structure d'un algorithme

**Algorithme** nom-algorithme

**Entête**

**Variable** nom-variable : type\_variable

**Déclaration**

**Début**

Instruction 1 ;

Instruction 2 ;

Instruction 3 ;

.....

Instruction n ;

**Fin**

**Corps : les tâches à exécuter.**

## Algorithme Titre du Problème

### Déclaration

- ✓ Déclaration des Constantes
- ✓ Déclaration des Variables
- ✓ Déclaration des Tableaux
- ✓ Déclaration des Procédures et Fonctions

### Manipulation

Début

*Actions*

FIN

# Déclaration des constantes

## Définition:

Les données sont des informations nécessaires au déroulement d'un algorithme. On distingue deux catégories : les **constantes** et les **variables**.

### 1. Les constantes

Une constante est une donnée fixe qui ne varie pas durant l'exécution d'un algorithme. Une constante est caractérisée par son **nom** et sa **valeur** (fixe).

#### ➤ Syntaxe

**Constante** Nom \_Constante = valeur ;

#### ➤ Exemple

Constante Pi = 3,14 ;

Constante B = 10 ;

Constante Mois = 'Janvier' ;

# Déclaration des variables

## Les variables

Une variable est un objet dont le contenu peut être modifié par une action durant l'exécution d'un algorithme. Une variable est caractérisée par son nom, sa valeur et son type.

### ➤ Syntaxe

**Variable** Nom\_Variable : **Type** ;

### ➤ Exemple

**Variable** *nb\_élèves* : **entier** ;

**Variable** *note* : **réel** ;

**Variable** *Prénom* : **chaines de caractères** ;

**Variable** *B* : **booléen** ;

**Variable** *OP* : **caractère** ;

## Les types de données

Type de données	Numérique			Alphanumérique		Booléen
	<i>Entier</i> (sans la virgule)	<i>Réel</i> (Avec et sans la virgule)		<i>Caractères</i>	<i>Chaîne de caractères</i>	
<b>Exemples</b>	-345 178 2012	-123,56 56,12 3	$4,1 \times 10^{38}$ 18 -123	'A,' '2 ' '+' '@' '? ' '\$'	'Ibn Batouta' '49', '3872' 'Bonjour'	Vrai Faux

### Remarque

Une variable de type numérique ne peut pas recevoir une chaîne de caractères ou booléen.

# L'affectation

## L'affectation

L'affectation est une opération qui consiste à attribuer une valeur ou expression à une variable. Elle est représentée par une flèche orientée à gauche :  $\leftarrow$  .

### ➤ Syntaxe

**Variable  $\leftarrow$  Valeur ou expression**

### ➤ Exemple

A  $\leftarrow$  4 : la variable A reçoit la valeur 4

Nb  $\leftarrow$  A+5 : la variable Nb reçoit le contenu de A plus 5

Nom  $\leftarrow$  'Bonjour' : la variable Nom reçoit la valeur **salut**

# L'affectation

Variable  $A, B, C$ : Entier ;

$A \leftarrow 10$

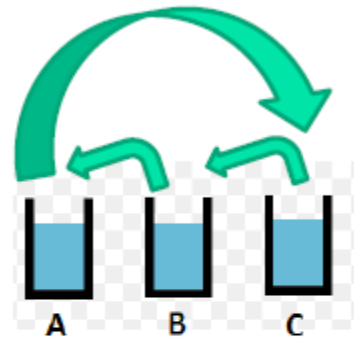
$B \leftarrow 20$

$C \leftarrow 30$

$B \leftarrow C$

$A \leftarrow B$

$C \leftarrow A$



Résultat :

$B = 30$

$A = 30$

$C = 30$



# L'écriture

L'écriture est une instruction qui permet **d'afficher** le contenu d'une **variables** ou/et un **message** sur l'écran.

## ➤ Syntaxe

```
Ecrire (variable);  
Ecrire ('message');  
Ecrire ('message', variable);
```

## ➤ Exemple

Soit A est une variable.

Ecrire (A) : signifie affiché sur l'écran le contenu de la variable A.

Ecrire ('Bonjour!') : signifie affiché sur l'écran le message suivant:  
Bonjour!

# La lecture

La lecture est une instruction qui permet de **lire** des données tapées au clavier.

## Syntaxe

```
Lire (variable1) ;  
Lire (variable2) ;  
Lire (variable1, variable2) ;
```

## Exemple

```
Lire (nombre) ;
```

```
Lire (Nb1,Nb2) ;
```

- **Exemple**

Écrire un algorithme qui demande un nombre et qui affiche un message 'Bonjour' et aussi le double de ce nombre.

- **Solution**

**Algorithme:** *nombre*

**Variable** x: entier

**Début**

**Ecrire** ('donner un nombre entier')

**Lire** (x)

**Ecrire** ('Bonjour',  $2 \times x$ )

**Fin**