

# Les algorithmiques

1.0

*Université Constantine*



Dr.CHAIB AOUATEF



# Table des matières



<b>Objectifs</b>	<b>5</b>
<b>Introduction</b>	<b>7</b>
<b>I - Initiation à l'algorithmique</b>	<b>9</b>
<b>A. Quelques notions de bases</b> .....	<b>9</b>
1. <i>Algorithme</i> .....	<b>9</b>
2. <i>Programme</i> .....	<b>10</b>
3. <i>Langage de programmation</i> .....	<b>10</b>
<b>B. Propriétés d'un algorithme</b> .....	<b>10</b>
<b>C. Format générale d'un algorithme</b> .....	<b>10</b>
1. <i>Nom de l'algorithme</i> .....	<b>11</b>
2. <i>Notion de variable</i> .....	<b>11</b>
3. <i>Notion de constante</i> .....	<b>12</b>
4. <i>Les instructions</i> .....	<b>12</b>
<b>D. Les instructions</b> .....	<b>13</b>
1. <i>L'instruction d'affectation</i> .....	<b>13</b>
2. <i>Les instructions de lecture et d'écriture</i> .....	<b>13</b>
3. <i>L'instruction conditionnelle</i> .....	<b>14</b>
4. <i>Les instructions itératives (les boucles)</i> .....	<b>16</b>



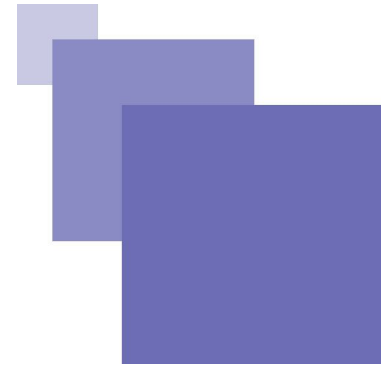
# Objectifs

A la fin de ce cours vous serez capable de :

1. Reconnaître les concepts de base de l'algorithmique et de la programmation
2. Mettre en œuvre ces concepts
3. Analyser des problèmes simples
4. Écrire les algorithmes correspondants



# Introduction



## Public cible

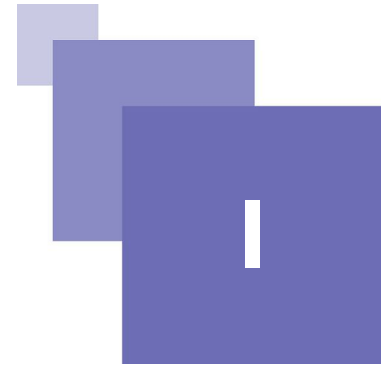
Les étudiants du 3<sup>ème</sup> année LMD : Physio Toxicologie, Génétique, Entomologie, Immunologie

L'**algorithmique** désigne la discipline qui étudie les **algorithmes** et leurs applications en Informatique. Une bonne connaissance de l'algorithmique permet d'écrire des algorithmes exacts et efficaces. Le mot **algorithme** vient du nom du célèbre mathématicien arabe **Al Khwarizm (Abu Ja'far Mohammed Ben Mussa Al-Khwarismi)**





# Initiation à l'algorithmique



Quelques notions de bases	9
Propriétés d'un algorithme	10
Format générale d'un algorithme	10
Les instructions	13

## A. Quelques notions de bases

### 1. Algorithme



#### Définition

Un algorithme est une suite de raisonnements ou d'opérations qui fournit la solution de certains problèmes, Il permet d'expliciter clairement les idées d'une solution d'un problème indépendamment d'un langage de programmation



### 2. Programme

Un programme est un assemblage et un enchaînement d'instructions élémentaires écrites dans un langage de programmation, et exécuté par un ordinateur afin de traiter les données d'un problème et renvoyer un ou plusieurs résultats.

### 3. Langage de programmation

Un langage de programmation fournit un ensemble de mots-clés et de règles de syntaxe qui permettent de créer des instructions formant des programmes et qui peuvent s'exécuter, sans souci, sur une machine.

## B. Format générale d'un algorithme



*Structure général d'un algorithme*

### 1. Nom de l'algorithme

chaque algorithme doit avoir un nom qui permet son identification.

## 2. Notion de variable

**Variables :**  
**Ident** : Type

*La déclaration d'une variable*



*Exemple*

**Variables :**

A, B : entier

Val1, Val2 : réel

Test : booléen

## 3. Notion de constante

**Constantes :**  
**Ident = Valeur**

*Déclaration d'une constante*



*Exemple*

**Constantes :**

D= 55

QTE1= 45 .05

ElemTRV= Vrai

Ch= 'K'

## 4. Les instructions

Un algorithme est formé de quatre types d'instructions considérées comme des petites briques de base :

- L'affectation de variables
- La lecture et l'écriture
- Les tests
- Les boucles

Dans la section suivante on va détailler chaque type d'instruction.

## C. Les instructions

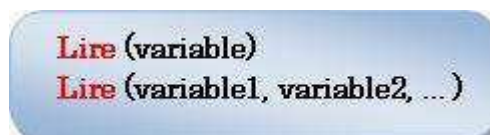
### 1. L'instruction d'affectation



*Structure générale de l'instruction d'affectation*

### 2. Les instructions de lecture et d'écriture

a) L'instruction de lecture(entrée)



*La structure de l'instruction de Lecture*

b) L'instruction d'écriture (sortie)

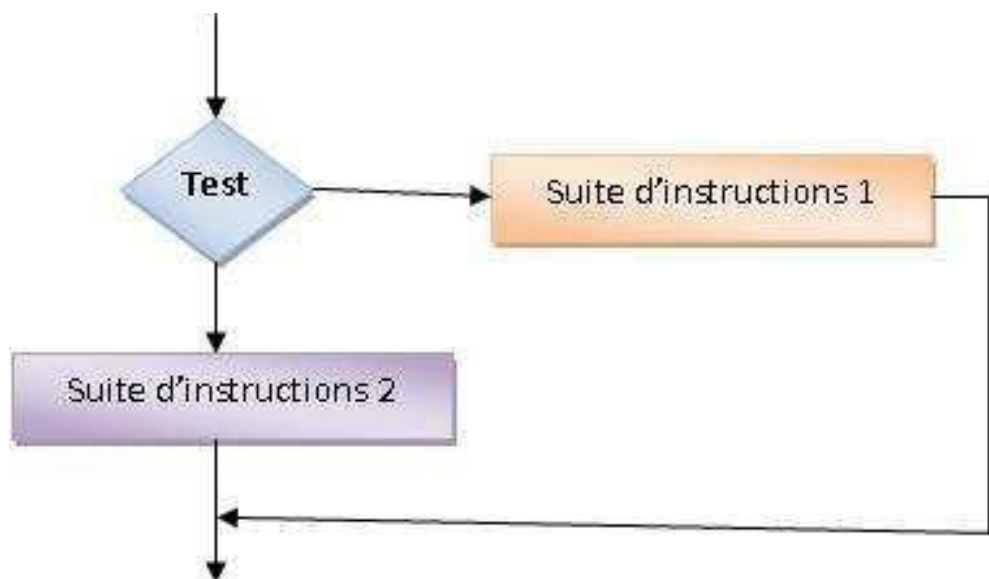


Structure de l'instruction d'Écriture

3. L'instruction conditionnelle



Structure générale de l'instruction conditionnelle



Organigramme de l'instruction conditionnelle



### Remarque

---

On peut avoir une instruction conditionnelle simple. Ce type d'instructions incorpore un bloc d'instructions dont son exécution dépend de la condition qui lui a été associée.



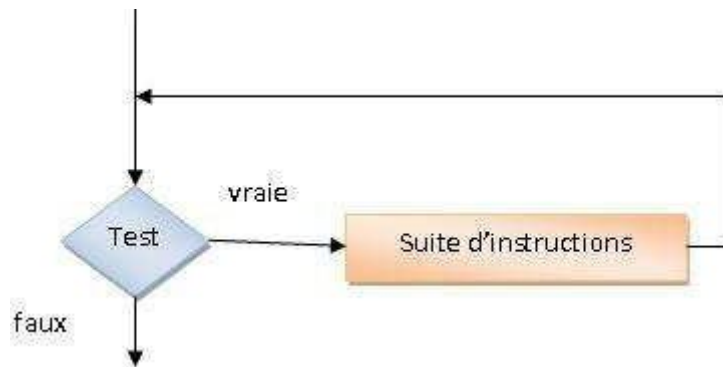
*L'instruction conditionnelle simple*

## 4. Les instructions itératives (les boucles)

### a) Les boucles tant que



*L'instruction tant que*



*Organigramme de l'instruction tant que*



### Remarque

Le nombre d'itérations dans une boucle TantQue n'est pas connu au moment d'entrée dans la boucle. Il dépend de l'évolution de la valeur de la condition



### Remarque

Une des instructions du corps de la boucle doit absolument changer la valeur de la condition de vrai à faux (après un certain nombre d'itérations), sinon le programme va tourner indéfiniment



### Exemple

Comment afficher le message «Bonjour tous le monde » 20 fois ?

```
Algorithme : affichage
Déclaration :
    i : entier
Début
    i ← 1
    Tant que (i ≤ 20) faire
        Ecrire ('Bonjour tous le monde')
        i ← i + 1
    Fin TQ
Fin
```

Exemple

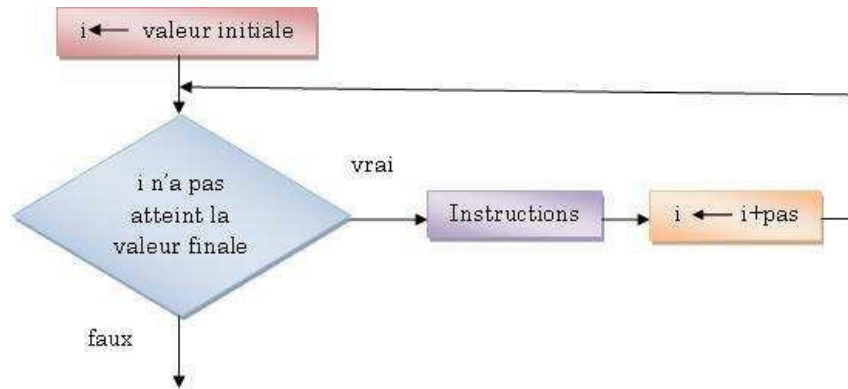
### b) Les boucles pour ou avec compteur

```
Pour i allant de (valeur initiale) à (valeur finale) (pas= valeur) faire :
    Instruction 1
    Instruction 2
    ...
    Instruction N
Fin pour
```

L'instruction POUR







Organigramme de l'instruction POUR

**Remarque**

Le nombre d'itérations dans une boucle Pour est connu avant le début de la boucle

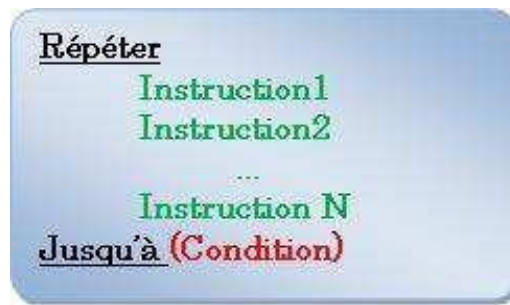
**Exemple**

Comment afficher le message «Bonjour tous le monde » 20 fois ?

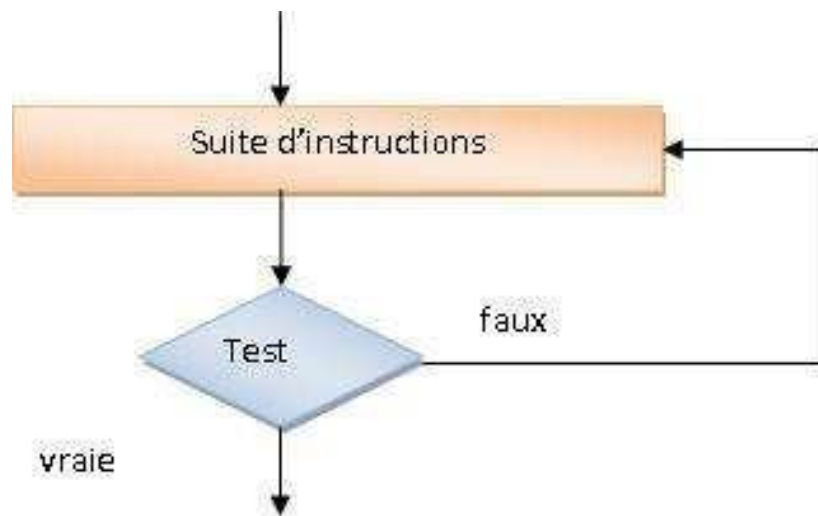


Exemple

c) Les boucles RÉPÉTER



*L'instruction RÉPÉTER*



*Organigramme de l'instruction RÉPÉTER*



*Exemple*

Comment afficher le message «Bonjours tous le monde » 20 fois ?

**Algorithme** : affichage

**Déclaration** :

$i$  : entier

**Début**

$i \leftarrow 1$

Répéter

Ecrire ('Bonjour tous le monde')

$i \leftarrow i+1$

Fin Jusqu'à ( $i > 20$ )

**Fin**

*exemple*