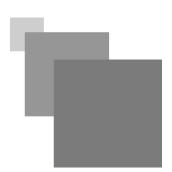
### LEÇON 2: LES OPÉRATEURS ET INSTRUCTIONS





## Table des matières

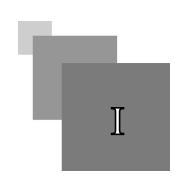
Ob	ojectifs	3
I -	LES OPÉRATEURS ET INSTRUCTIONS	4
	1. Prérequis	4
	2. Les Opérateurs	
	2.1. Les opérateurs arithmétiques	4
	2.2. Les opérateurs de comparaisons ou relationnels	5
	2.3. Les opérateurs logiques	5
	2.1. Les opérateurs arithmétiques 2.2. Les opérateurs de comparaisons ou relationnels 2.3. Les opérateurs logiques 2.4. Les opérateurs de concaténations	6
	3. Les instructions	
	3.1. L'instruction d'affectation	7
	3.2. 4.2 L'instruction Écrire ou Afficher	7
	3.3. L'instruction Lire ou Saisir	8
	3.4 Rappel	8

### **Objectifs**



- Connaître les opérateurs existant ;
- Connaître les différentes priorités entre les opérateurs ;
- Connaître les différentes instructions pour la construction d'IHM (Interface Homme Machine).

# LES OPÉRATEURS ET INSTRUCTIONS



#### 1. Prérequis

Avant de suivre cette leçon, vous devez avoir pour acquis la connaissance de la structuration d'un algorithme et la connaissance des cinq propriétés d'un algorithme de la leçon 1.

#### 2. Les Opérateurs

Un opérateur est un signe qui relie au moins deux valeurs, pour produire un résultat. On distingue à cet effet :

- Les opérateurs arithmétiques
- Les opérateurs de comparaisons ou relationnels
- Les opérateurs logiques
- Les opérateurs de concaténations

#### 2.1. Les opérateurs arithmétiques

Ce sont les opérations arithmétiques classiques :

Opérateur Arithmétique	Représentation en Algorithme
Addition	+
Soustraction	-
Multiplication	*
Division	/
Division entière	Div
Modulo	Mod
Puissance	۸

#### Remarque

- Les opérateurs arithmétiques sont toujours utilisés avec des nombres.
- L'addition, la soustraction, la multiplication, la division et la puissance sont les opérateurs que vous utilisez habituellement.
- L'opérateur div ramène le quotient d'une division entière. Exemple : 9 div 2 = 4 (il ne conserve que la partie entière du quotient qui est 4).
- L'opérateur mod ramène le reste d'une division entière. Exemple : 9 mod 2 = 1 (il conserve le reste qui est 1).

#### 2.2. Les opérateurs de comparaisons ou relationnels

Ces opérateurs donnent toujours un résultat de type Booléen (vrai ou faux).

Opérateur comparaisons	Représentation en Algorithme
Supérieur	>
Inférieur	<
égalité	=
Supérieur ou égal	>=
Inférieur ou égal	<=
Différent	<> ou !=



#### 👉 Exemple

8 < 7 = Faux ; 8 > 7 = Vrai.

#### 2.3. Les opérateurs logiques

Opérateur logiques	Représentation en Algorithme
La négation	Non
L'intersection	Et
L'union	Ои



#### Remarque

Une condition est une expression de type logique. Ils lui correspondent deux valeurs possibles VRAI et FAUX.



#### 👉 Exemple : Tableau d'évaluation des opérateurs logiques

- Opérateur de négation

A	Non A
Vrai	Faux
Faux	Vrai
В	Non B
Vrai	Faux

#### - Opérateur d'intersection

A	В	A ET B
Vrai	Vrai	Vrai
Faux	Faux	Faux
Vrai	Faux	Faux
Faux	Vrai	Faux

#### - Opérateur d'union

A	В	A OU B
Vrai	Vrai	Vrai
Faux	Faux	Faux
Vrai	Faux	Vrai
Faux	Vrai	Vrai

#### 2.4. Les opérateurs de concaténations

La concaténation s'effectue toujours entre les chaînes de caractères. Une concaténation est une association de plusieurs caractères ou chaînes de caractères. Pour concaténer deux chaînes de caractères, on utilise l'opérateur de concaténation, cet opérateur se note avec le signe "+" ou "&".

En algorithme, un caractère ou chaîne de caractère (plusieurs caractères) est (sont) toujours entre guillemets.

#### Remarque

2 est un nombre entier et "12" est une chaîne de caractère. Un caractère vide est désigné par un espace entre guillemets " ".

#### F Exemple: Exemple 1

10+25 donne 35

#### 👉 Exemple : Exemple 2

A ─ "Bon" ;

B **←** "jour" ;

 $C \leftarrow A + B$ ;

La variable C contient maintenant la chaîne : "Bonjour".

#### Exemple: Exemples 3

В ← " " ;

 $D \leftarrow A + B + C$ ;

E **←** A+C ;

La variable D contient maintenant la chaîne : "Bonjour cher étudiant".

La variable E contient maintenant la chaîne : "Bonjourcher étudiant".

#### 👉 Exemple : Exemples 4

A **←** "1" ;

B **←** "2" ;

 $C \leftarrow A + B$ ;

La variable C contient maintenant la chaîne : "12".

#### 3. Les instructions

Une instruction est un ordre. Toutes les lignes qui se trouvent à l'intérieur d'un algorithme sont des instructions.

#### 3.1. L'instruction d'affectation

L'instruction d'affectation se note avec le signe  $\leftarrow$ .

#### **Exemple**

Nombre ← 24 (Attribue la valeur 24 à la variable Nombre).

Une instruction d'affectation ne modifie que la variable situé à gauche de la flèche.

Exemple : Som  $\leftarrow$  Nombre + 4 (Attribue la valeur de la variable Nombre plus 4 donc 24 + 4 = 28 à la variable Som).

#### 3.2. 4.2 L'instruction Écrire ou Afficher

Elle permet d'envoyer l'information qui a été emmagasiné quelque part dans la mémoire.

#### - Exemple

1er cas : Afficher un message

Afficher "Nous sommes des étudiants de l'UVCI " => ramène le message délimiter par les griffes : Nous sommes des étudiants de l'UVCI

2e cas : Afficher un résultat

Soit la variable : nombre  $\leftarrow 15$ 

Afficher nombre => ramène le contenu de nombre en occurrence 15.

3e cas : Afficher un message et un résultat

Soit la variable : som  $\leftarrow 15 + 9$ 

Afficher"La somme est : "som=>ramène le message et le contenu de som : La somme est 24

#### **S** Fondamental

On peut utiliser soit l'instruction Afficher ou Écrire

#### 3.3. L'instruction Lire ou Saisir

Elle attend jusqu'à ce que l'utilisateur entre une valeur et qu'il est validé cette valeur entrée.

#### 👉 Exemple

Soit la variable nombre;

Saisir nombre ou *Lire* nombre => La valeur qui sera saisit pas l'utilisateur ira dans le contenu de la variable nombre. En clair, lorsque l'utilisateur saisi la variable *nombre* aura comme valeur 2017. Et lorsque l'ordinateur lit la variable *nombre* il lit en fait la valeur 2017.

#### 3.4. Rappel

#### Rappel

L'algorithme doit avoir une structure bien définie et comportant :

- 1. L'en-tête qui comprend le nom de l'algorithme pour identifier l'algorithme.
- 2. Les déclarations des variables et des constantes.
- 3. Le corps de l'algorithme qui contient les instructions.

Toutes les instructions doivent être situer entre le mot *Début* et le mot *Fin*, et chaque instruction doit comporter un point-virgule à la fin.

Algorithme nom_d'algorithme	L'en-tête
Variable nom_variable : type_variable  Constante nom_constante = valeur	Les déclarations
Début	
Instruction 1 Instruction 2	Le corps
Instruction 3	