

Chapitre n°2 :

## Les actions élémentaires simples

Objectifs du cours :

- L'affectation.
- Les opérations d'entrée / sortie.

### Leçon 1

### L'affectation

I - Définition (voir livre page 46)

II - Syntaxe (voir livre page 47)

**Activité :** H et B sont deux entier. Soit la séquence d'instruction suivante :

1.  $H \leftarrow 10$
  2.  $H \leftarrow H + H$
  3.  $B \leftarrow - 2 * H$
  4.  $H \leftarrow H + B \text{ DIV } H$
  5.  $B \leftarrow \text{ABS}(B)$
- Dresser le tableau de la trace de la séquence précédente.

Séquence	H	B
1	10	-
2	20	-
3	20	-40
4	18	-40
5	18	40

*Remarque :*

- Une affectation peut ne pas s'exécuter à cause d'un problème de compatibilité de Type.

## Leçon 2

### Les opérations d'entrée / sortie

#### I - Les opérations de sortie

1. **Définition** (voir livre page 54)
2. **Syntaxe** (voir livre page 56)
3. **Formatage de l'affichage des résultats**

Exemples :

- Affichage des entiers : `writeln (n:5);`
- Affichage des réels : `writeln (x:5:2);`
- Affichage des caractères : `writeln (c:3);`

4. **GotoXY ( )**

#### II - Les opérations d'entrée

1. **Définition** (voir livre page 64)
2. **Syntaxe** (voir livre page 66)

##### Activité :

Écrire un programme qui lit un entier  $n$ , supposé de 3 chiffres distincts. Calculer puis afficher l'entier  $m$  obtenu par l'inversion des chiffres de  $n$ .

*Exemple :*  $n = 273$  alors  $m = 372$ .

$$\begin{aligned}
 m &= 300 + 70 + 2 \\
 &= \mathbf{3} \times 100 \quad + \mathbf{7} \times 10 \quad + \mathbf{2} \\
 &= n \text{ MOD } 10 \times 100 + (n \text{ DIV } 10) \text{ MOD } 10 \times 10 + (n \text{ DIV } 10) \text{ DIV } 10
 \end{aligned}$$

##### a. Analyse :

Nom : <b>inverse</b>		
S	L.D.E	O.U
6	Résultat = Ecrire ("l'inverse est : ", m)	m
5	$m \leftarrow u * 100 + d * 10 + c$	u
2	$u \leftarrow n \text{ MOD } 10$	d
3	$d \leftarrow (n \text{ DIV } 10) \text{ MOD } 10$	c
4	$c \leftarrow (n \text{ DIV } 10) \text{ DIV } 10$	n
1	n = Donnée ("saisir un entier à 3 chiffre : ")	
7	Fin <b>inverse</b>	

**T.D.O**

Objet	Type	Rôle
n	Entier	
m	Entier	
u	Entier	
d	Entier	
c	Entier	

**b. Algorithme**

- 0) Début **inverse**
- 1) écrire ("saisir un entier à 3 chiffre : ")  
Lire (n)
- 2)  $u \leftarrow n \text{ MOD } 10$
- 3)  $d \leftarrow (n \text{ DIV } 10) \text{ MOD } 10$
- 4)  $c \leftarrow (n \text{ DIV } 10) \text{ DIV } 10$
- 5)  $m \leftarrow u * 100 + d * 10 + c$
- 6) Ecrire ("l'inverse est : ", m)
- 7) Fin **inverse**

**c. Traduction en Pascal** (voir fichier : inverse.pas)*Constatations :*

Dans cette activité il n'y a que des ordres de lectures d'affectation et d'écritures.

D'une façon général on peut extraire les chiffres d'un nombre comme suit :

```

program extraction_chiffres;
uses wincrt;
var
  n:integer;
begin
  read(n);
  writeln ('dizaine de millier : ',n div 10000);
  writeln ('millier : ',(n mod 10000)div 1000 );
  writeln ('centaine : ',((n div 10)div 10)mod 10);
  writeln ('dizaine : ',(n div 10)mod 10);
  writeln ('unité : ',n mod 10);
end.

```

- Erreur lors de la compilation :
  1. Type mismatch
  2. Expected
  3. Unknown identifier
  4. Syntax error
  5. Error in Statement
  6. Constant out of rang
  
- Erreur lors de l'exécution :
  1. Runtime error