

## Série des Exercices N° : 02

### 1. Les opérateurs

#### Exercice N°1

Soit le programme C suivant :

```
#include <stdio.h>
main()
{ int A,B,C,D,E;
  A = 5;   B =(A<0)?1,2,3:10,20,30; printf("B = %d\n", B);
  A = -5;  B =(A<0)?1,2,3:10,20,30; printf("B = %d\n", B);
  A = 5;   C =(A<0)?(1,2,3):(10,20,30); printf("C = %d\n", C);
  A = -5;  C =(A<0)?(1,2,3):(10,20,30); printf("C = %d\n", C);
  D = -1,-2,-3; printf("D = %d\n", D);
  E =(-1,-2,-3); printf("D = %d\n", E); getch();
}
```

1. Que seront les valeurs de B, C, D et E après l'exécution de ce programme ?

#### Exercice N°2

Soit le programme C suivant :

```
#include <stdio.h>
main()
{ int A, B, C, D;
  A = 4, B = 1, C = A++ > B || B++ != 2;
  printf ("Resultat 1 : A = %d B = %d C = %d\n", A, B, C);
  A = 4; B = 1; C = ++A == 5 && ++B == 2;
  printf ("Resultat 2 : A = %d B = %d C = %d\n", A, B, C);
  A=1, D = ++A == (B = C =2) ;
  printf ("Resultat 3 : A = %d B = %d C = %d D = %d\n", A, B, C, D);
  A = B = C = 10 ; A += B += C ;
  printf ("Resultat 4 : A = %d B = %d C = %d\n", A, B, C);
  A = 5; C = A << 1; B = A >> 2;
  printf ("Resultat 5 : A = %d B = %d C = %d\n", A, B, C);
  A = 4, B = 1;
  C = A & B; A = B | 2; B ^= 4 ;
  printf ("Resultat 6 : A = %d B = %d C = %d\n", A, B, C);
  getch();
}
```

1. Que seront les résultats affichés par le programme ci-dessus après son exécution ?

## 2. Les boucles

### Exercice N°3

Soit les boucles suivantes :

<u>Boucle 1</u>	<u>Boucle 2</u>
<pre>int n; do { printf ("donnez un nb &gt;0 : "); scanf ("%d", &amp;n) ; printf("valeur donnée %d\n", n); } while (n&lt;=0) ; printf ("valeur correcte") ;</pre>	<pre>int n; do { } while ( printf ("donnez un nb &gt; 0 :"), scanf ("%d", &amp;n), printf ("valeur donnée %d", n), n &lt;= 0 ) ; printf ("valeur correcte") ;</pre>

1. Y-à-t-il une différence entre ces boucles ?
2. Que fait chacune de ces boucles ?
3. Utiliser la boucle **for** pour écrire une boucle équivalente à Boucle 1.

### Exercice N°4

Écrire un programme en C qui permet de lire un nombre positif donné par l'utilisateur, puis il calcule et affiche sa factorielle.

- Modifier votre programme pour calculer les factorielles des 10 premiers nombres entiers strictement positifs

### Exercice N°5

Écrire un programme en C indépendant pour calculer chacune des sommes suivantes.

$S = 1 - 3 + 5 - 7 + 9 - 11 + \dots \pm n$  tel que  $n > 50$  et donné par l'utilisateur.

$S = 1 + \frac{2}{3} + \frac{3}{5} + \frac{4}{7} + \dots + \frac{n}{2n-1}$  tel que  $n$  est un nombre positif donné par l'utilisateur.

$S = x + \frac{x+1}{2!} + \frac{x+2}{3!} + \dots + \frac{x+i}{(i+1)!} + \dots + \frac{x+n}{(n+1)!}$  tel que  $n, x$  données par l'utilisateur et  $n > 0$ .

$S = \sum_{k=1}^n A^k * X^{n-k}$  tel que  $A, X$  et  $n$  sont données par l'utilisateur et  $n \geq 10$ .

### Exercice N°6

Un nombre est dit parfait s'il est égal à la somme de ses diviseurs sauf lui-même.  
Exemple:  $6 = 1 + 2 + 3$ .

Écrire un programme en C qui permet de trouver les nombres parfaits compris entre deux nombres  $M$  et  $N$  positifs donnés par l'utilisateur.

### Exercice N°7

Soit  $U_n$  et  $V_n$  deux suites définies comme suit :

$$U_0 = 2 \quad U_n = 1 + \frac{1}{V_n} \quad \text{et} \quad V_n = 1 - \frac{1}{U_{n-1}} \quad (n > 0)$$

Écrire le programme qui permet de calculer la somme de 100 premiers termes de la suite  $V_n$ .