

Série des Exercices N° : 02

Exercice N°1

A : Soit le programme C suivant :

```
#include <stdio.h>
main()
{ int A,B,C,D,E;
  printf("Donner un nombre :"); scanf("%d",&A);
  B=(A<0)?1,2,3:10,20,30; printf("B = %d\n", B);
  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 pour une valeur positive et autre négative de A ?

B : Soit le programme C suivant :

```
#include <stdio.h>
main()
{ int X;
  do {
    printf("Donner un nombre positif:"); scanf("%d",&X);
  }
  while (X <= 0);
  printf("Valeur donnée X = %d \n", X);
  getch();
}
```

1. Que fait ce programme ? 2. Ecrire un programme équivalent en utilisant la boucle **for**.

Exercice N°2

Ecrire un programme en C permettant d'afficher la table de multiplication par 9.

Exercice N°3

Ecrire un programme en C qui permet de lire une suite d'entiers saisis par l'utilisateur et d'afficher le plus grand et le plus petit nombre ainsi que ses rangs dans cette suite. On tape 0 pour terminer la suite.

Exercice N°4

Ecrire un programme en C qui permet de lire une suite de 10 valeurs entières positives saisis par l'utilisateur et d'afficher leur somme. Le programme doit afficher un message d'erreur dans le cas de saisis d'une valeur négative.

Exercice N°5

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

Exercice N°6

Ecrire un programme en C qui permet de calculer les sommes suivantes.

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

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

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

$$S = 1 - 4 + 9 - 16 \dots + (-1)^{n+1} n^2 \quad \text{tel que } n \text{ donné par l'utilisateur et } n > 0.$$

Exercice N°7

Ecrire un programme en C qui permet de simuler la fonction d'une petite calculatrice. C-à-d, qu'il demande deux valeurs de l'utilisateur ainsi que l'opération à exécuter sur ces nombres. Ensuite, il réalise cette opération.

NB : le programme ne s'arrête pas à répéter une nouvelle demande jusqu'à ce que l'utilisateur donne deux nombres nuls.

Exercice N°8

Soit U_n et V_n deux suites définies comme suit :

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

Ecrire le programme qui permet de calculer la somme de 100 premiers termes de la suite V_n

Exercice N°9

Ecrire un programme en C qui permet d'afficher tous les nombres premiers dans l'intervalle $]2 .. N]$. Tel que $n > 2$ et donné par l'utilisateur.

Exercice N°10

Ecrire un programme en C qui répète de calculer le carré d'un nombre entier non nul donné par l'utilisateur. Le programme s'arrête lorsque l'utilisateur donne une valeur nulle.

Exercice N°11

Un nombre est dit parfait s'il est égal à la somme de ses diviseurs sauf lui-même.

Ecrire un programme en C qui permet de trouver les nombres parfaits compris entre deux nombres M et N donnés. Par Exemple: $6=1+2+3$.

Exercice N°12

On dit que deux nombres sont équivalents si les sommes des chiffres formant ces nombres sont égales. Ecrire un programme en C qui permet de lire deux nombres entiers positifs et de vérifier si sont équivalents.

Exercice N°13

Ecrire un programme en C qui permet de convertir et d'afficher la valeur équivalente en décimal d'un nombre donné par l'utilisateur en binaire pur.

Exercice N°14

Ecrire un programme qui permet de remplir les éléments d'un tableau T par des valeurs entières entrées au clavier et affiche le tableau T. Ensuite, effacer toutes les occurrences de la valeur 0 dans le tableau T et tasser les éléments restants. Puis, afficher le tableau résultant.

Tableau entré : N=10

11	0	-5	23	57	0	2	8	0	3
----	---	----	----	----	---	---	---	---	---

Tableau résultant : N=7

11	-5	23	57	2	8	3
----	----	----	----	---	---	---

Exercice N°15

Ecrire un programme en C qui permet de lire les éléments entières d'un tableau T (taille maximale de T est $N \leq 50$) et affiche le tableau T. Ranger ensuite les éléments du tableau T dans l'ordre inverse sans utiliser de tableau d'aide et afficher le tableau résultant.

Exercice N°16

Ecrire un programme en C qui permet de lire les éléments entières d'un tableau T (taille maximale de T est $N \leq 50$) et affiche le tableau T. Ensuite faire le tri de T comme suit :

Methode 1

Pour chaque place i dans le tableau T, on parcourt le sous-tableau T[i+1 .. nbElements] de gauche à droite, et on échange T[i] avec le plus petit élément trouvé dans ce sous-tableau.

Methode 2

Pour chaque place i dans le tableau T, on parcourt le sous-tableau T[i+1 .. nbElements] de la fin au début en échangeant tout couple d'éléments consécutifs non ordonnés.