

## TP - Programmation

### Série de TP N°3 – Sous-Programmes : Procédures et Fonctions

**Exercice N°1** : Soit le programme suivant :

```
#include <stdio.h>
float a= 10,b= 5,c= 0 ;

void sous_prog1(float x, float y, float s)
{
    s = x + y;
}
void sous_prog2(float x, float y, float* s)
{
    *s = x + y;
}
int main() //Début du programme principal
{
    sous_prog1 (a, b, c);
    printf("La somme est : %.2f \n", c);
    float a = 10, b = 5, c = 0;
    sous_prog2 (a, b, &c);
    printf("La somme est : %.2f \n", c);
    return 0;
} //Fin du programme principal
```

#### Questions :

- 1) Exécuter le programme.
- 2) Quelle est la différence entre les deux procédures **sous\_prog1** et **sous\_prog2** ?
- 3) Quels sont les paramètres à passage par valeur et ceux à passage par variable ?
- 4) Quels sont les paramètres formels des deux procédures ?
- 5) Quels sont les paramètres effectifs ?
- 6) Quels sont les variables locales et globales
- 7) Dérouler le programme.
- 8) Exécuter le programme en donnant le type « **int** » à la variable **c**. Que se passe-t-il ? pourquoi ?

**Exercice N°2** : Soit le programme suivant :

```
#include<stdio.h>

int maximum(int a, int b) {
    if (a>b)
        return a ;
    else
        return b ; }

int main() {
    int x, y, results ;
    printf("Donner la première valeur de x: ");
    scanf("%d", &x) ;
    printf("Donner la deuxième valeur de y: ");
    scanf("%d", &y) ;
    results= maximum (x, y) ;
    printf("Le maximum entre %d et %d est : %d \n ", x, y, results) ;
    return 0 ; }
```

#### Questions :

- 1) Exécuter le programme pour  $x = 5$  et  $y = 3$ .
- 2) Dérouler le programme pour  $x = 5$  et  $y = 3$ .
- 3) Quels sont les paramètres formels de la fonction ?
- 4) Quels sont les paramètres effectifs ?
- 5) Réécrire le programme pour déterminer le maximum entre trois nombres entiers  $x$ ,  $y$  et  $z$ .

#### Exercice N°3 :

Écrire un programme en C qui lit les 10 valeurs entières d'un tableau à une dimension, comportant une procédure permet d'ordonner le tableau dans l'ordre croissant.

#### Exercice N°4 :

Écrire un programme en C qui permet de calculer  $N!$ , en utilisant la fonction **fact**. Ensuite réécrire le programme en remplaçant la fonction par une procédure.

## TP - Programmation

### Série de TP N°3 – Sous-Programmes : Procédures et Fonctions

#### Exercice Sup - 01:

Écrire un programme en C qui effectue le calcul de la somme **S** en utilisant deux fonctions distinctes :

- Une fonction qui calcule le factoriel d'un nombre entier.
- Une fonction qui calcule la puissance d'un nombre réel élevé à un exposant entier.

Le programme doit ensuite utiliser ces deux fonctions pour calculer la **somme S** suivante:

$$S = -\frac{2!}{x^3} + \frac{3!}{x^4} - \frac{5!}{x^6} + \frac{7!}{x^8} - \frac{11!}{x^{12}} + \dots - \frac{N!}{(x)^{N+1}}$$

Avec : **N** est un nombre entier positif non nul et **x** un nombre **flottant (reel)** non nul

#### Exercice Sup - 02 :

Écrire un programme en C qui permet de lire un tableau **T** de **N réels**. Le programme doit faire appel à une procédure qui détermine le plus grand élément du **tableau T** ainsi que sa position.

#### Exercice Sup – 03 :

Écrire un programme en C qui calcule une approximation de **ln(1+x)** en utilisant la **série de Taylor**.

1. Écrire une **fonction Puissance** qui calcule **a<sup>n</sup>** pour un nombre **réel a** et un entier **n**.
2. Écrire une **fonction fact** qui calcule le factoriel d'un entier **n**.
3. Écrire une **fonction ln\_approximation** qui calcule une approximation de **ln(1+x)** pour un nombre réel **x** (où  $-1 < x \leq 1$ ) et un entier **n**, en utilisant la série suivante :

$$\ln(1+x) \approx x - \frac{x^2}{2} + \frac{x^3}{3} - \frac{x^4}{4} + \dots + \frac{(-1)^{n-1} \cdot x^n}{n}$$

4. Dans le programme principal, demander à l'utilisateur de saisir un nombre réel **x** (où  $-1 < x \leq 1$ ) et un entier **n**, et afficher l'approximation de **ln(1+x)**.

#### Exercice Sup – 04 :

Écrire un programme en C qui lit les valeurs d'une matrice (**N × N**) de type **entier**, comportant une fonction qui permet de retrouver le nombre de fois qu'un nombre (choisi par l'utilisateur) apparaît.

#### Exercice Sup – 05 :

Écrire un programme en langage C qui permet de calculer la **moyenne** des notes de **10** étudiants. Le programme doit :

1. Demander à l'utilisateur de saisir les notes des 10 étudiants.
2. Stocker ces notes dans un tableau.
3. Utiliser une **fonction** pour calculer la moyenne des notes.
4. Utiliser une **procédure** pour insérer les notes dans le tableau.
5. Afficher la **moyenne** calculée à la fin.