

TP - Programmation

Série de TP N°4 – Procédures et fonctions

Exercice N°01 :

Soit le programme C suivant :

```
#include <stdio.h>

void sous_prog1(float x, float y, float s) {
    s = x + y;
}
void sous_prog2(float x, float y, float* s) {
    *s = x + y;
}
//Début du programme principal
int main() {
    float a = 10, b = 5, c = 0; //Variables globales du
                               //programme

    sous_prog1(a, b, c);
    printf("La somme est : %f\n", c);

    a = 10; b = 5; c = 0;
    sous_prog2(a, b, &c);
    printf("La somme est : %f\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) Dérouler le programme.
- 7) Exécuter le programme en donnant le type « int » à la variable c. Que se passe-t-il ? pourquoi ?

Exercice N°2 :

Ecrire une procédure qui prend deux entiers en entrée et qui échange leurs valeurs.

Exercice N°03 :

Soit le programme C suivant :

```
#include<stdio.h>
int maximum(int a, int b)
{
    int max; //Variables locales de la fonction maximum
    if (a>b)
        max=a ;
    else
        max=b ;
}
int main() //Début du programme principal
{
    int a, b, results ; //Variables globales du programme
    printf("Donner la première valeur : ");
    scanf("%d", &a) ;
    printf("Donner la deuxième valeur : ");
    scanf("%d", &b) ;
    results= maximum (a, b) ;
    printf("Le maximum entre %d et %d est : %d \n ", a, b,
    results) ;
    return 0 ;
} //Fin du programme principal
```

Questions :

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

Exercice N°4 :

Écrire une fonction qui calcule la somme de deux nombres entiers passés en paramètres et retourne le résultat. Écrire un programme qui utilise cette fonction pour calculer et afficher la somme de deux nombres saisis par l'utilisateur.

TP - Programmation

Série de TP N°4 – Procédures et fonctions

Exercice N°05 :

Soit le programme C suivant :

```
#include <stdio.h>

int Fact(int m) {
    int j, f = 1;
    for(j = 1; j <= m; j++) {
        f *= j;
    }
    Fact=f;
}

int main() {
    int i, n;
    float S;
    printf("Introduire n : ");
    scanf("%d", &n);
    S= 1;
    for(i = 1; i <= n; i++) {
        S =S+ Fact(i);
    }
    printf("La somme = %.2f", S);
    return 0;
}
```

- 1) Exécuter le programme pour n = 3.
- 2) Dérouler le programme pour n = 3.
- 3) Réécrire le programme en transformant la fonction Fact en une procédure Fact.
- 4) Soit la procédure « Puiss » suivante :

```
void Puiss(double t, int k, double *P) {
    int i;
    *P = 1;
    for(i = 1; i <= k; i++) {
        *P =P* t;
    }
}
```

- 4-a) Dérouler la procédure Puiss pour t=2 et k=4 et déduire ce qu'elle fait.
- 4-b) Transformer la procédure Puiss en une fonction.
- 5) On propose de calculer la valeur approximative de l'exponentiel de x avec la formule suivante :

$$e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots + \frac{x^n}{n!}$$

En utilisant les deux fonctions Fact et Puiss, écrire un programme C pour calculer e^x .

TP - Programmation

Série de TP N°4 – Exercices supplémentaires sur les procédures et fonctions

Exercice N°01 :

- Écrire une fonction « Max » qui calcule le maximum entre deux nombres réels X et Y.
- En utilisant la fonction « Max », écrire un programme Pascal permettant de calculer et afficher le maximum entre trois nombres réels X, Y et Z.

Exercice N°02 :

- Écrire un programme qui lit un **tableau** T de N réels, fait appel à une **procédure** qui détermine le plus grand élément du tableau ainsi que sa position (son rang dans le tableau). Écrire cette procédure et l'insérer dans le programme. Afficher les résultats dans le programme principal.

Exercice N°03 :

- Écrire une procédure « Permuter » qui réalise la permutation entre deux réels X et Y.
- En utilisant la procédure « Permuter », écrire un programme C permettant de permuter les deux diagonales d'une matrice carrée A(N×N) de type réel.

Exercice N°04 :

- Écrire en C une fonction « Nombre_Premier » qui vérifie si un nombre entier N est premier ou non.

N.B : Un nombre est dit « premier » s'il est divisible uniquement par 1 et lui-même.

- Insérer la fonction « Nombre_Premier » dans un programme Pascal complet permettant de lire une matrice A (N×M) de type entier et de calculer et afficher le **nombre total des nombres premiers** présents dans la matrice A.
- Réécrire le programme précédent en transformant la fonction « Nombre_Premier » en une procédure du même nom.