

## Correction TD 3 programmation 1

### Exercice 1 :

- Fonction remplirTab :

```
void remplirTab(float tab[ ],int taille)
{
    int i;
    for (i =0;i <taille;i++)
    {
        printf("Saisir la valeur de la case %d ", i);
        scanf("%f",&tab[i]);
    }
}
```

- Fonction afficheTab :

```
void afficheTab(float tab[],int taille)
{
    int i=0;
    for (i=0;i<taille;i++)
        printf("La valeur de la case %d est %f \n",i,t[i] );
}
```

- Fonction incTab :

```
void incrTab(float tab[],int taille )
{
    int i=0;
    for (i=0; i< taille;i++)
        t[i]=t[i]+1;
}
```

### Exercice 2 :

```
/* Fichier incrémente
* Ce fichier contient le programme qui permet
* d'incrémenter les valeurs saisies par
* l'utilisateur
* Auteur : SMI2017
* Version : v1
```

```

/* Date création : aujourd'hui*/
#include<stdlib.h>
#include<stdio.h>
void remplirTab(float t[ ],int taille)
{
    int i;
    for (i =0;i< taille;i++)
    {
        printf("Saisir la valeur de la case %d ", i);
        scanf("%f",&tab[i]);
    }
}
void afficheTab(int t[],int taille)
{
    int i=0;
    for (i =0;i <taille;i++)
        printf("La valeur de la case %d est %d \n",i,t[i]);
}
void incrTab(float t[],int taille)
{
    int i=0;
    for (i =0; i < taille; i++)
        t[i]=t[i]+1;
}
int main()
{
    int n;
    float tab[10] ;
    printf("Saisir le nombre des notes ( max est 10 )");
    scanf ("%d",&n);
    remplirTab(tab,n);
    incrTab(tab,n);
    afficheTab(tab,n);
    return EXIT_SUCCESS;
}

```

### **Exercice 3 :**

```

/* Fichier chercher.c
* Ce fichier contient le programme qui permet la recherche
* d'un élément parmi les valeurs saisies par
* l'utilisateur
* Version : v1

```

```

* Date création : aujourd'hui
*/
#include< stdlib.h>
#include< stdio.h>
int chercherMotif(int T[], int taille, int motif )
{
    int i;
    for (i =0;i < taille;i++)
    {
        if( t[i] == motif)
            return 1;
    }
    return 0;
}
void remplirTab(float tab[ ],int taille)
{
    int i;
    for (i =0; i < taille; i++)
    {
        printf("Saisir la valeur de la case %d ", i);
        scanf("%f",&tab[i]);
    }
}
int main( )
{
    float n,motif,tab[20] ;
    int result;
    printf("Saisir le nombre d'éléments( max est 20 )");
    scanf ("%d",&n);
    remplirTab(tab,n);
    printf("Elément recherché ? :");
    scanf ("%d",&motif);
    result = chercherMotif(tab,n,motif);
    if (result == 1)
        printf("L'élément n'existe pas ");
    else
        printf("L'élément existe ");

    return EXIT_SUCCESS;
}

```

**Exercice 4 :**

```
#include<stdlib.h>
#include<stdio.h>
/* Fonction qui calcule le max de 2 réels*/
float max2( float a, float b )
{
    int max;
    if (a>= b)
        max=a;
    else
        max=b;
    return max;
}
/* Fonction qui calcule le max de 3 réels */
float max3(float a, float b, float c )
{
    int m, max;
    m = max2 (a,b);
    max =max2(m,c)
    return max;
}
/* Fonction principale qui lance l'exécution */
int main()
{
    float x, y, z, max ;
    printf("Saisir trois réel : ");
    scanf ("%f%f%f",&x,&y,&z);

    max = max3(x,y,z);
    printf("La max est %f : ",max);
    return EXIT_SUCCESS;
}
```

**Exercice 5 :**

```
void tri(int T[],int taille )
{
    int i,j ;
    int tmp;
    for (i =0;i<taille;i++)
    {
        for (j=i+1; j<taille;j++)
```

```

        {
            if( T[j]<T[i] )
            {
                tmp = T[i];
                T[i] = T[j];
                T[j] = tmp;
            }
        }
    }
}

```

### **Exercice 6 :**

*/\* Calcul du minimum, maximum et moyenne d'un tableau \*/*

```

#include<stdio.h>
#include<stdlib.h>
/* Saisie au clavier des éléments d'un tableau */
void saisie(double *tab,int nb)
{
    int i;
    for (i=0;i<nb;i++)
    {
        printf ("Valeur de tab[%d] = ",i);
        scanf ("%lf",&tab[i]);
    }
    return;
}

```

*/\* Affichage des éléments d'un tableau \*/*

```

void affiche(double *tab,int nb)
{
    int i;

    for (i=0;i<nb;i++)
    {
        printf ("tab[%d] = %f\n",i,tab[i]);
    }
    printf("\n");
    return;
}

```

*/\* Calcul de la moyenne \*/*

```
double calculer_moyenne (double *tab,int nb)
{
    double moyen, somme;
    int i;
    somme = 0;
    for (i=0;i<nb;i++)
    {
        somme = somme + tab[i];
    }
    moyen = somme / nb;
    return moyen;
}
```

*/\* pmin et pmax contiendront resp. le min et le max du tableau tab \*/*

```
void chercher_min_max (double *tab, int nb, double *pmin, double *pmax)
{
    double val_min, val_max;
    int i;
    val_min = tab[0];
    val_max = tab[0];
    for (i=0; i<nb; i++)
    {
        if (tab[i] < val_min)
        {
            val_min = tab[i];
        }
        else if (tab[i] > val_max)
        {
            val_max = tab[i];
        }
    }
    *pmin = val_min;
    *pmax = val_max;
}
```

```
int main ()
{
    int nb_val;
    double min, max,moyenne, table[10];

    printf("Nombre d'\elements ? ");
    scanf ("%d", &nb_val);
```

```

    saisie(table, nb_val);
    affiche(table, nb_val);

    moyenne = calculer_moyenne(table, nb_val);
    printf("Moyenne = %f\n", moyenne);

    chercher_min_max(table, nb_val, &min, &max);
    printf("Min = %f\tMax = %f\n", min, max);

    return EXIT_SUCCES;
}

```