

## Test TD - Corrigé

### Algorithmique et Structures de Données

Durée 1h15

#### Exercice 1 : (03 points)

Questions (répondez par une simple phrase )

1) Quel est le rôle de `#include <stdio.h>`

Inclure la bibliothèque standard contenant les fonctions d'entrées/sorties notamment `printf`

2) Quelle est la différence entre les constantes `'7'` et `7` ?

La différence se situe dans le type ; `'7'` est du type caractère quant au `7` il est du type entier.

Soit le QCM suivant (cochez la bonne réponse):

1) Quel sera le résultat de l'exécution du code c suivant ?

```
#include<stdio.h>
#include<math.h>
int main(){
    int c, a= 6,b=3;
    if (a>b && b!=0)
        c = a % b ;
    else c= a + b ;
    printf("%d",c) ;
    return 0 ;
}
```

- ☐ 2  
☒ 0  
☐ 9  
☐ Erreur de compilation

2) Quel sera le résultat de l'exécution du code c suivant ?

```
#include<stdio.h>
int main(){
    char tab[11] ="The African Queen" ;
    printf("%s",tab) ;
    return 0 ;
}
```

- ☐ The African Queen  
☐ The  
☒ The African  
☒ Erreur de compilation

3) Quel sera le résultat de l'exécution du code c suivant ?

```
#include<stdio.h>
int main(){
    int X=0;
    switch(5/4/3){
        case X:printf("Clinton"); break;
        case X+1:printf("Gandhi"); break;
        case X+2:printf("Gates"); break;
        default: printf("Brown");
    }
    return 0;
}
```

- ☐ Clinton  
☐ Gandhi  
☐ Gates  
☐ Brown  
☒ Erreur de compilation

4) Après l'exécution du code c suivant ,combien de fois s'afficher "Hello" ?

```
#include<stdio.h>
int main(){
    for(int i=0; i<=6; i++) {
        printf("Hello") ; }
    return 0 ;
}
```

- ☐ 5 fois  
☐ 6 fois  
☒ 7 fois  
☐ Erreur de compilation

NOM :.....  
PRENOM :.....  
Groupe :.....

## Exercice 2 : ( 07 points)

soit le tableau de 50 notes Tab[50]. Ecrire un programme en c qui permet de :

1. Saisir les notes dans le tableau par l'utilisateur (les notes doivent être entre 0 et 20).
2. Compter le nombre de notes supérieur ou égale à 10 , puis l'afficher
3. Calculer la moyenne des notes saisis, ensuite l'afficher.

```
include <stdio.h>
int main()
{
    float tab[50];
    int n,i,j,nbrMot,som=0;
    float Moy=0;
    printf("entrer 50 notes (entre 0 et 20\n)");
    for( i=0;i<50; i++){
        printf("tab[%d]= ",i);
        scanf("%f",&tab[i]);
        if(tab[i]<0 || tab[i]>20 ){
            //saisie erronée, recommencer la saisie ;
            printf("tab[%d]= ",i);
            scanf("%f",&tab[i]);
        }
    }
    // compter le nombre de note au dessus de la moyenne et l'afficher
    for( i= 0;i<50; i++){
        if (tab[i]>= 10 ){
            nbrMot++;
        }
    }
    printf("le nombre de note au dessus de la moyenne= %d",nbrMot);
    // calculer la moyenne et l'afficher

    for(i=0;i<50;i++){

        som=som+tab[i];
    }
    Moy=som/nbrMot;
    printf("la moyenne des notes saisis = %.2f",Moy);
    return 0;
}
```

### Exercice 3 : ( 08 points)

Soit Mat[n][n] une matrice carrée d'entiers. Ecrire un programme en c qui permet de :

1. Demander à l'utilisateur la taille de la matrice.
2. Remplir la matrice.
3. Compter le nombre des valeurs négatifs puis les mettre à zéro .
4. Inverser les éléments de la diagonale principale.

Exemple :

1	-2	5	6		3	0	5	6
7	9	-1	-4		7	7	0	0
8	2	7	-1	→	8	2	9	0
4	-2	5	3		4	0	5	1

```
#include <stdio.h>
int main()
{
    int mat[100][100];
    int n,i,j,nNeg=0,temp;
    printf("entrer la taille de la matrice \n");
    scanf("%d",&n);
    // remplir la matrice
    for( i=0;i<n; i++){
        for( j= 0; j<n; j++){
            printf("mat[%d][%d]=",i,j);
            scanf("%d",&mat[i][j]);
        }
    }

    // compter le nombre des val négatives et les remettre a zéro
    for( i=0;i<n; i++){
        for( j= 0; j<n; j++){
            if (mat[i][j]<0){
                nNeg++;
                mat[i][j]=0;
            }
        }
    }

    printf("le nombre des valeurs negatives= %d",nNeg);
    // inverser les éléments de la diagonale principale
    for( i=0;i<=n/2; i++){
        for( j= 0; j<=n/2; j++){
            temp= mat[i][i];
            mat[i][i]= mat[n-1-i][n-1-i];
            mat[n-1-i][n-1-i]= temp;
        }
    }

    //affichage de la matrice après inversement
    for( i=0;i<n; i++){
        for( j= 0; j<n; j++){
            printf("%d",mat[i][j]);
        }
    }

    return 0;
}
```

