

Examen : Informatique : Langage C
SMI /S3
Documents non Autorisés

Durée : 1 h

Exercice 1: Qu'affiche le programme suivant :

```
#include <stdio.h>
#include <stdlib.h>
#include <conio.h>

void main() {
    int i=18;
    i=i--i;
    switch(i) {
        case 1 : (void)printf("Premier\n");
        case 2 : (void)printf("Deuxième\n");
        case 3 : (void)printf("Troisième\n");
        default : (void)printf("Non classe\n");
    }
    getch();
}
```

Solution :

.....

.....

.....

.....

.....

.....

Exercice 2:

Qu'affichera à l'écran l'exécution de ce programme ?

```
#include <stdio.h>
#include <stdlib.h>
#include <conio.h>

void main()
{
  int i;
  for (i=0;i<5;i++)
  {
    printf("le resultat est : %d\n",(i+9)/(i+1));
  }
  getch();
}
```

Solution :

A large rectangular area containing horizontal dotted lines for writing the solution. The lines are evenly spaced and cover the entire width of the box.

Exercice 3: Triangle d'étoiles

Compléter la fonction `afficherTriangle` dans le programme ci-dessous : cette fonction devra afficher un triangle rempli d'étoiles (*) sur un nombre de lignes donné passé en paramètre, exemple :

Lignes ? 8

```
*
**
***
****
*****
*****
*****
*****
*****
```

```
#include <stdio.h>
#include <stdlib.h>
#include <conio.h>
void afficherTriangle(int nbLignes)
{ // Partie à compléter
.....
.....
}
// Fonction principale pour test
void main()
{
    int nbLignes = 0;
    printf("Lignes ? ");
    scanf("%d", &nbLignes);
    if (nbLignes > 0)
        afficherTriangle(nbLignes);
    else
        printf("Erreur : Vous devez entrer un entier strictement positif !\n");
    getch() ;
}
```

Solution :

```
void afficherTriangle(int nbLignes)
{
.....
.....
.....
.....
.....
.....
.....
.....
}
```

Exercice 4 : Qu'imprime le programme suivant ?

```
void main()
{
    int x=3,y=4,z=5;

    int *ptr1=&x,*ptr2=&y;

    *ptr1 += 1;
    printf("x=%d\n",x);

    ++*ptr1;

    printf("x=%d\n",x);

    (*ptr1)++;

    printf("x=%d\n",x);

    *ptr2 = (*ptr1)++;

    printf("x=%d , y=%d\n",x,y);

    *ptr2 *= *ptr1 /= *ptr2++;

    printf("x=%d , y=%d\n",x,y);

    ptr2=&y;

    z += -*ptr1++ + ++*ptr2;

    printf("x=%d , y=%d, z=%d\n",x,y,z);
}
```

Solution :

WWW.TALIB24.COM

Examen de Programmation : LANGAGE C
SMI /S3
Documents non Autorisés

Réservé à l'étudiant	Réservé au professeur
<p>Nom :</p> <p>Prénom :</p> <p>Code apogée :</p> <p>CIN :</p>	<p>Note :</p>
<p>Réservé au surveillant</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p>	

Exercice 1

Note : /4

1. Soit les deux programmes suivants :

Programme 1

```

1. #include <stdio.h>
2. main()
3. {
4.     int A[50][50]; /* matrice initiale */
5.     int B[50][50]; /* matrice résultat */
6.     int N, M;      /* dimensions des mat*/
7.     int I, J;      /* indices courants */
8.     /* Saisie des données */
9.     printf("Nombre de lignes (max.50) : ");
10.    scanf("%d", &N );
11.    printf("Nombre de colonnes (max.50) : ");
12.    scanf("%d", &M );
13.    for (I=0; I<N; i++)
14.    for (J=0; J<M; J++)
15.    {
16.        printf("Elément[%d][%d] : ", I, J);
17.        scanf("%d", &A[I][J]);
18.    }
19.    /* Affichage de la matrice */
20.    printf("Matrice donnée :\n");
21.    for (I=0; I<N; I++)
22.    {
23.        for (J=0; J<M; J++);
24.        printf("%7d", A[I][J]);
25.        printf("\n");
26.    }
27.    /* Traitement */
28.    for (I=0; I<N; I++)
29.    for (J=0; J<M; J++)
30.        B[J][I]=A[I][J];
31.    /* Edition du résultat */
32.    /* affichage */
33.    printf("Matrice résultat :\n");
34.    for (I=0; I<M; I++)
35.    {
36.        for (J=0; J<N; J++)
37.            printf("%7d", B[I][J]);
38.        printf("\n");
39.    }
40.    return 0;
41. }

```

Programme 2

```

1. #include <stdio.h>
2. main()
3. {
4.     /* Déclarations */
5.     int A[50][50]; /* matrice donnée */
6.     int N, M; /* dim de la matrice */
7.     int I, J; /* indices courants */
8.     int AIDE; /* pour la permutation */
9.     int DMAX; /* la plus grande des 2 dim*/
10.    /* Saisie des données */
11.    printf("Nombre de lignes (max.50) : ");
12.    scanf("%d", &N );
13.    printf("Nombre de colonnes (max.50) : ");
14.    scanf("%d", &M );
15.    for (I=0; I<N; I++)
16.    for (J=0; J<M; J++)
17.    {
18.        printf("Elément[%d][%d] : ", I, J);
19.        scanf("%d", &A[I][J]);
20.    }
21.    /* Affichage de la matrice */
22.    printf("Matrice donnée :\n");
23.    for (I=0; I<N; I++)
24.    {
25.        for (J=0; J<M; J++)
26.            printf("%7d", A[I][J]);
27.        printf("\n");
28.    }
29.    /* traitement */
30.    DMAX = (N>M) ? N : M;
31.    for (I=0; I<DMAX; I++)
32.    for (J=0; J<I; J++)
33.    {
34.        A[I][J]= AIDE;
35.        A[I][J] = A[J][I];
36.        A[J][I] = AIDE;
37.    }
38.    /* affichage */
39.    printf("Matrice résultat :\n");
40.    for (I=0; I>M; I++)
41.    {
42.        for (J=0; J<N; J++)
43.            printf(A[I][J], "%7d");
44.        printf("\n");
45.    }
46.    return 0;

```

1. Détecter et corriger les erreurs des programmes ci-dessus

programme	N° de ligne	Type d'erreur : syntaxique/logique	erreur	Correction

2. si l'utilisateur saisit une valeur de nombre de ligne ou de colonne supérieure à 50,

2. 1. Que va-t-il se passer ?

.....
.....
.....

2. 2. Que proposez-vous ?

.....
.....
.....
.....
.....
.....

3. que font le programme 1 et le programme 2 ?

Programme 1 :

.....
.....
.....

Programme 2 :

.....
.....
.....

4. donner un exemple d'exécution pour les deux en affichant le résultat final.

Programme 1 :

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

Programme 2 :

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

Exercice 2 (les fonctions)

Note : /4

1. Dans le programme C suivant, donnez les erreurs et les corrections à effectuer.

```
int addition(int a, int b);  
  
{  
int a, b, c;  
c = a + b;  
return c;  
}
```

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

2. Ecrire une fonction qui calcule le minimum des deux entiers :

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

3. Ecrire une fonction qui trie un tableau a une dimension de N entiers

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

Exercice 4 (les tableaux)

Note : /4

Ecrire un programme qui permet de :

1. Remplir les **4** premières lignes et **4** colonnes de la matrice M[4][5] de type **int**
2. Placer le Maximum des nombres Impaires de chaque ligne dans la dernière case de la même ligne
3. Afficher le numéro de ligne qui contient le plus grand Nombre Impaire

Exemple :

1	5	3	4	5
5	3	8	7	7
7	6	11	4	11
5	9	5	8	9

Réponse

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

Exercice 5 (les pointeurs)

Note : /4

Qu'affiche le programme suivant :

```
void main()
{
int A = 10 , B = 10, C , D;
int *ptr1, *ptr2;

ptr1 = &A;
ptr2 = &B;

C = *ptr1 + *ptr2;

ptr1 = &C;

D = *ptr1;

*ptr2 = D -10;

printf("A = %d \nB = %d \nC = %d \nD = %d", A, B, C, D);
}
```

Réponse

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Bonne Chance

Examen de Programmation : LANGAGE C
SMI /S3
Documents non Autorisés

Durée : 1 H 30

Réservé à l'étudiant	Réservé au professeur
Nom :	Note :
Prénom :	
Code apogée :	
CIN :	

Exercice 1	Note : /4
<p>1. donner la syntaxe de l'opérateur conditionnel : « ? : »</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>2. Ecrire plus simplement l'instruction suivante :</p> <p style="text-align: center;">$z = (x1 > x2 ? x1 : x2) - (x1 \leq x2 ? x1 : x2).$</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>3. n étant de type int, écrire une expression qui prend la valeur :</p> <ul style="list-style-type: none">-1 si n est négatif0 si n est null1 si n est positif. <p>Note : utiliser l'opérateur de comparaison « ? : »</p> <p>.....</p> <p>.....</p> <p>.....</p>	

Exercice 2 (les fonctions)

Note : /4

1. Que fait le programme suivant :

```
#include<stdio.h>
#include<conio.h>
void main()
{
int a,b,c,temp;
clrscr();
printf("a=");scanf("%d",&a);
printf("b=");scanf("%d",&b);
printf("c=");scanf("%d",&c);
if (b>a)
    {temp=a; a=b; b=temp;}
if (c>a)
    {temp=a; a=c; c=temp;}
if (c>b)
    {temp=b; b=c; c=temp;}
printf("%d %d %d",a,b,c);
}
```

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

3. Donner pour un exemple de données d'entrée a, b, c l'état de sortie.

.....

.....

.....

.....

Exercice 3

Note : /4

Soit le programme suivant :

Ecrire un programme réalisant exactement la même chose avec :

Une instruction while

Une instruction do ... while

```
#include <stdio.h>
#include <conio.h>
void main()
{ int i, n, som;
  som = 0;
  for (i = 0; i < 4; i++)
  {
  printf(" donnez un entier ");
  scanf ("%d", &n);
  som+=n;
  }
  printf ("somme : %d\n", som);
}
```

```
#include <stdio.h>
#include <conio.h>
void main()
{
}
```

```
#include <stdio.h>
#include <conio.h>
void main()
{
}
```


Exercice 5 (les pointeurs)

Note : /4

Qu'affiche le programme suivant :

```
#include <stdio.h>

#include <stdlib.h>

void main()

{

int A = 20 , B = 20, C , D;

int *ptr1, *ptr2;

ptr1 = &A;

ptr2 = &B;

C = *ptr1 + *ptr2;

ptr1 = &C;

D = *ptr1 ;

*ptr2 = D -10;

D = *ptr2 +20;

printf("A = %d \nB = %d \nC = %d \nD = %d", A, B, C, D);

}
```

Réponse

.....

.....

.....

.....

Bonne chance