

## TD3:

## Langage C

## Exercice 1 :

Quelles erreurs ont été commises dans chacun des groupes d'instructions suivants :

<pre>if (a&lt;b) printf ("ascendant") else printf ("non ascendant") ;</pre>	<pre>int n ; ... switch (2*n+1) { case 1 : printf ("petit") ;   case n : printf ("moyen") ; }</pre>
<pre>#define LIMITE 100 int n ; ... switch (n) { case LIMITE-1 : printf ("un peu moins") ;   case LIMITE : printf ("juste") ;   case LIMITE+1 : printf ("un peu plus") ; }</pre>	<pre>const int LIMITE=100 int n ; ... switch (n) { case LIMITE-1 : printf ("un peu moins") ;   case LIMITE : printf ("juste") ;   case LIMITE+1 : printf ("un peu plus") ; }</pre>

## Exercice 2 :

<pre>main() { int n ;   scanf ("%d", &amp;n) ;   switch (n)   { case 0 : printf ("Nul\n") ;     case 1 :     case 2 : printf ("Petit\n") ;               break ;     case 3 :     case 4 :     case 5 : printf ("Moyen\n") ;     default : printf ("Grand\n") ;   } }</pre>	<p>Quels résultats affiche-t-il lors qu'on lui fournit en donnée :</p> <p>a) 0 b) 1 c) 4 d) 10 e) -5</p>
---	--

## Exercice 3 :

<p>Quelles erreurs ont été commises dans chacune des instructions suivantes :</p> <p>a) do c = getchar() while (c != '\n') ;</p>	<p>b) do while ( (c = getchar()) != '\n') ; c) do {} while (1) ;</p>
--	--

## Exercice 4 :

Ecrire plus lisiblement :

```
do {} while (printf("donnez un nombre >0 "), scanf ("%d", &n), n<=0) ;
```

## Exercice 5 :

<p>Soit le petit programme suivant :</p> <pre>#include &lt;stdio.h&gt; main() { int i, n, som ;   som = 0 ;   for (i=0 ; i&lt;4 ; i++)   { printf ("donnez un entier ") ;     scanf ("%d", &amp;n) ;     som += n ; }   printf ("Somme : %d\n", som) ; }</pre> <p>Ecrire un programme réalisant exactement la même chose, en employant, à la place de l'instruction for :</p> <p>a) une instruction while, b) une instruction do ... while.</p>	<p>Quels résultats fournit le programme suivant :</p> <pre>#include &lt;stdio.h&gt; main() { int n=0 ;   do   { if (n%2==0)     { printf ("%d est pair\n", n) ;       n += 3 ;       continue ; }     if (n%3==0)     { printf ("%d est multiple de 3\n", n) ;       n += 5 ; }     if (n%5==0)     { printf ("%d est multiple de 5\n", n) ;       break ; }     n += 1 ; }   while (1) ; }</pre>
---	---

## Exercice 6 :

## TD3:

## Langage C

Ecrivez un programme qui calcule les solutions réelles d'une équation du second degré «  $ax^2 + bx + c = 0$  » en discutant la formule.

Utilisez une variable d'aide **D** pour la valeur du discriminant «  $b^2 - 4ac$  » et décidez à l'aide de **D**, si l'équation a une, deux ou aucune solution réelle. Utilisez des variables du type **int** pour **A**, **B** et **C**.

Considérez aussi les cas où l'utilisateur entre des valeurs nulles pour **A**; pour **A** et **B**; pour **A**, **B** et **C**.

Affichez les résultats et les messages nécessaires sur l'écran.

**Exercice 7 :**

Calculez la somme des N premiers termes de la série harmonique :

$$1 + 1/2 + 1/3 + \dots + 1/N$$

**Exercice 8 :**

Afficher un triangle isocèle formé d'étoiles. La hauteur du triangle (c'est-à-dire le nombre de lignes) sera fournie en donnée, comme dans l'exemple ci-dessous. On s'arrangera pour que la dernière ligne du triangle s'affiche sur le bord gauche de l'écran.

```
combien de lignes ? 10
      *
     ***
    *****
   *********
  ***********
 *****
*****
*****
*****
*****
```

**Exercice 9 :**

Ecrire un programme qui détermine la  $n^{\text{ème}}$  valeur un (**n** étant fourni en donnée) de la "suite de Fibonacci" définie comme suit :

$$u_1 = 1$$

$$u_2 = 1$$

$$u_n = u_{n-1} + u_{n-2} \text{ pour } n > 2$$

**Exercice 10 :**

Ecrire un programme qui trouve la plus grande et la plus petite valeur d'une succession de notes (nombres entiers entre 0 et 20) fournies en données, ainsi que le nombre de fois où ce maximum et ce minimum ont été attribués. On supposera que les notes, en nombre non connu à l'avance, seront terminées par une valeur négative (**ne pas utiliser de "tableau"**). L'exécution du programme pourra se présenter ainsi :

```
donnez une note (-1 pour finir) : 12
donnez une note (-1 pour finir) : 8
donnez une note (-1 pour finir) : 13
donnez une note (-1 pour finir) : 7
donnez une note (-1 pour finir) : 11
donnez une note (-1 pour finir) : 12
donnez une note (-1 pour finir) : 7
donnez une note (-1 pour finir) : 9
donnez une note (-1 pour finir) : -1
note maximale : 13 attribuée 1 fois
note minimale : 7 attribuée 2 fois
```