

**Programmation I : TD/TP2**  
(Entrées-Sorties, Tests, Boucles)

**Exercice 1**

Ecrire un programme qui affiche le nombre d'octets occupés par les types :  
char , short, int, long, float, double, long double.

**Exercice 2**

Ecrivez un programme C qui lit trois variables réelles a, b et c. Ensuite le programme :

1. Effectue une permutation circulaire de ces variables (transfère à a la valeur de b, à b la valeur de c et à c la valeur de a) et affiche les valeurs des variables avant et après permutation.
2. Affiche la plus grande des trois valeurs.

**Exercice 3**

Soit le programme suivant :

```
#include <stdio.h>
main( )
{ short n ;
  scanf("%d",&n) ;
  switch (n)
  {
    case 0 : printf("Null\n") ;
    case 1 :
    case 2 : printf("petit\n") ; break ;
    case 3 :
    case 4 : printf("Moyen\n");
    default : printf("Grand\n");
  }
}
```

Quels résultats affiche-t-il pour les valeurs suivantes de n :

- a) 0    b) 1    c) 4    d) 10    e) -5

**Exercice 4 (T.P)**

1. Ecrivez un programme qui lit N et calcule la somme :  $1 + 1/2 + 1/3 + \dots + 1/N$
2. Ecrivez un programme qui lit une valeur epsilon et calcule la somme :

$$1 + 1/3 + 1/5 + 1/7 + \dots$$

L'arrêt de calcul est effectué quand le terme à ajouter est inférieur à epsilon

{  
while ( $\frac{A.0}{i} > \text{eps}$ )  
    {  
        S +=  $\frac{A.0}{i}$   
        i = i + 2;  
    }  
}

**Exercice 5**

Ecrivez un programme qui calcule le terme  $U_N$  de la suite de Fibonacci donnée par :

$$U_1=1, U_2=1, U_N=U_{N-1} + U_{N-2} (\text{pour } N>2)$$

### Exercice 6

On veut calculer une approximation de sinus de  $x$  par un développement limité :

$$\sin(x) = x - \frac{x^3}{6} + \frac{x^5}{5!} - \dots + (-1)^n \frac{x^{2n+1}}{(2n+1)!}$$

Écrivez un programme qui demande à l'utilisateur de saisir la valeur de  $x$  et  $n$  puis calcule une approximation de sinus de  $x$  avec  $(n+1)$  termes.

### Exercice 7

Un nombre entier  $p$  (différent de 1) est dit premier si ses seuls diviseurs positifs sont 1 et  $p$ .

1. Ecrivez un programme qui permet de déterminer si un nombre est premier ou non.
2. Ecrivez un programme qui affiche le nombre de nombres premiers entre 1 et 1000.

### Exercice 8

Soit la suite de nombres entiers strictement positifs définie par :

$$X_0 = A$$

$$X_n = \frac{X_{n-1}}{2}, \text{ si } X_{n-1} \text{ est pair}$$

$$X_n = 3X_{n-1} + 1, \text{ si } X_{n-1} \text{ est impair}$$

Cette suite atteint la valeur 1, quelle que soit la valeur  $A$  du premier terme.

Écrire un programme C qui saisie la valeur  $A$  depuis le clavier et affiche les termes de la suite, jusqu'à arriver à la valeur 1. Il faut s'assurer lors de la saisie que la valeur de  $A$  est un entier strictement positif.

### Exercice 9

Quels résultats fournit le programme suivant :

```
#include <stdio.h>
void main()
{
    long n = 0;
    do
    {
        if (n%2 == 0) { printf("%ld est pair\n", n); n += 3; continue; }
        if (n%3 == 0) { printf("%ld est multiple de 3\n", n); n += 5; }
        if (n%5 == 0) { printf("%ld est multiple de 5\n", n); break; }
        n += 1;
    }
    while (1);
}
```

Rq EX6 :

$$\begin{array}{l|l}
u=x & -x^3 = x \times -x^2 \\
\sin x & x^5 = x^3 \times -x^2 \\
& -x^7 = x^5 \times -x^2
\end{array} \quad \left| \begin{array}{l}
3! = 1+2+3 \\
5! = 3! \times 4 \times 5 \\
7! = 5! \times 6 \times 7
\end{array} \right.$$

$$\begin{cases} \sin = \sin + u \\ u = x \\ \sin x \end{cases} \quad \begin{array}{l}
i=1 \rightarrow 2i(2i+1) \\
i=2 \rightarrow 3i \cdot 2i(2i+1) \\
i=3 \rightarrow 5i \cdot 2i(2i+1)
\end{array}$$

$$u = u + \frac{-x^2}{2i(2i+1)}$$