

## TD : Les fonction en C++

### Rappel.

En C++ le retour de plusieurs valeur avec **return** n'est pas permis. En C++, on distingue deux types de passage d'argument par valeur ou par variable (par référence) :

```
void somme_produit(int a,int b, int & s, int & p)
{
    s=a+b ;
    p=a*b ;
}
```

Utilisation : (Il faut créer deux variables S et P)

```
int S,P ;
somme_produit(5,6,S,P) ;
cout<<S;
cout<<P;
```

0.1

Quelle modification faut-il apporter au programme suivant pour qu'il devienne correct :

```
#include <iostream>
using namespace std ;
main()
{
    int p=5 ;
    triple(p) ;
    cout << "le triple de " << p << " est égale à " << n ;
}
int triple(int r)
{
    int n; n=3*r ;
    return n;
}
```

0.2 Quels résultats fournira ce programme :

```
#include <iostream>
using namespace std ;

int n=10, q=2 ;

int fct (int) ;
void f (void) ;
main()
{
    int n=0, p=5 ;
    n = fct(p) ;
    cout << "A : dans main, n = " << n << " p = " << p << " q = " << q << "\n" ;
    f() ;
}
int fct (int p)
{
    int q ;
    q = 2 * p + n ;
    cout << "B : dans fct, n = " << n << " p = " << p << " q = " << q << "\n" ;
    return q ;
}

void f (void)
{
    int p = q * n ;
    cout << "C : dans f, n = " << n << " p = " << p << " q = " << q << "\n" ;
}
```

1-

a- Ecrire une fonction ***est\_premier*** qui détermine si un nombre entier positif a est premier. La fonction doit retourner un booléen.

b- Ecrire une fonction qui imprime tous les nombres premiers entre une valeur min et une valeur max. (tester dans un main)

c- Tester dans un main

2-

a- Ecrire une fonction qui affiche les solutions de l'équation d'une fonction  $a \cdot x^2 + b \cdot x + c = 0$ . Ecrire un main permettant de tester la fonction. Il y a plusieurs cas : 0 solutions, 1 solution, 2 solutions et une infinité de racines.

```
si a=0  
    si b=0  
        si c=0 alors une infinité de solutions  
        sinon aucune de solutions  
    sinon S={-c/b}  
  
    sinon  
        Delta=b2-4ac  
        si (Delta > 0) alors 2 solutions ....  
        si (Delta < 0) alors 0 solutions  
        si (Delta = 0) alors une solution ...
```

b- Ecrire une fonction « resoudre » qui retourne (qui n'affiche pas) les solutions de l'équation

$$a \cdot x^2 + b \cdot x + c = 0$$

La fonction ne doit faire aucun affichage ni saisie. Elle doit envoyer suffisamment d'information pour qu'une fonction d'affichage puisse afficher des messages tels que :

**Votre équation admet 2 solutions : 2.3 et 6.7**

ou

**Votre équation admet 1 solution : 4.5**

ou

**Votre équation n'admet pas de solutions**

....

b.1 Donner le prototype de la fonction ... resoudre(.....)

b.2 Ecrire la fonction resoudre

c- Ecrire une fonction « affiche » qui utilise le retour de la fonction «resoudre» pour afficher les solutions

```
resoudre(.....)  
afficher(.....)
```

3-

A partir d'une date exprimée comme trois numéros j,m,a (jour, mois, année), déterminer le jour de la semaine. Voici une méthode :

a- Ecrire une fonction qui reçoit un entier à quatre chiffres et qui retourne deux entiers : un correspondant aux deux premiers chiffres et un correspondant aux deux derniers chiffres.

b-

Soit la structure :

```
struct date
{
    int j ;
    int m ;
    int a ;
};
```

Soient **m'** et **a'** définis comme suit:

$$m' = \begin{cases} m - 2, & \text{si } m \geq 3 \\ m + 10, & \text{si } m < 3 \end{cases}$$

$$a' = \begin{cases} a, & \text{si } m \geq 3 \\ a - 1, & \text{si } m < 3 \end{cases}$$

Soit **ns** les deux premiers chiffres de **a'** et **as** les deux derniers chiffres de **a'**. On peut déterminer le jour de la semaine correspondant à une date donnée grâce à la formule suivante :

$$f = (j + a_s + \frac{a_s}{4} - 2 \cdot n_s + \frac{n_s}{4} + \frac{26 \cdot m' - 2}{10}) \bmod 7$$

**f=0** indiquera que c'est dimanche, **f=1** indiquera que c'est lundi, ...etc

Ecrire une fonction reçoit une date **d** et qui retourne un entier compris entre **0** et **6** : **0** indique dimanche, **1** indique lundi ....

c- Ecrire une fonction qui prend en argument un tableau de dates et qui affiche les jours correspondants aux dates qui y sont stockées.

d- Ecrire la fonction **ant** :

**bool ant(date &d1, date & d2)**

qui renvoie vrai si **d1** est antérieure à **d2**

e- Soit MinMax la fonction qui prend en argument un tableau de dates et sa taille et qui renvoie deux dates : la plus récente et la plus ancienne.

e.1 Peut on renvoyer les deux dates par un return ?

e.2 Ecrire la fonction MinMax (aucun cin, aucun cout)

e.3 Tester la fonction dans un main.