

TP 5 : Les structures conditionnelles

1. la structure conditionnelle Si :

La structure conditionnelle **Si** s'écrit en langage C++ comme suit :

```

if (condition)                if (condition)
< bloc d'instructions>         <bloc d'instructions 1>
                                else
                                <bloc d'instructions 2>
    
```

Le <bloc d'instructions> peut être une seule instruction se terminant par un point virgule ";" ou un ensemble d'instructions entre accolades " { }".

Exemples :

<pre> if (Nbr < 0) Nbr = - Nbr; </pre>	<pre> if (A!=B) { A=A+2; B=B+1 } </pre>
<pre> if ((Nbr%2) == 0) cout << "Le nombre est pair"; else cout<< "Le nombre est impair"; </pre>	<pre> if (A==B) { A=A+1; B=B+2; } else { A=A+2; B=B+1; } </pre>

On peut aussi imbriquer les if-else comme suit :

Imbrication dans la partie if	Imbriquer dans la partie else
<pre> if (condition1) if (condition2) <Bloc d'instructions 1> else <Bloc d'instructions 2> else <bloc d'instructions 3> </pre>	<pre> if (condition1) <Bloc d'instructions 1> else if (condition2) <Bloc d'instructions 2> else <Bloc d'instructions 3> </pre>

Remarque :

On peut imbriquer autant de **if** qu'on veut dans la partie **if** ou dans la partie **else**.

Exemples :

Exemple 1	<pre> if (X != 0) if (X < 0) cout << "X est négatif"; else cout<<"X est positif "; else cout<<" X est nul "; </pre>
Exemple 2	<pre> if (X == 0) cout<<" X est nul "; else if (X < 0) cout<<"X est négatif"; else cout<<" X est positif "; </pre>

Question :

Quelle est la différence entre les deux exemples ?

Pratiquement il n'y a pas de différence, c'est juste un changement de priorité de teste des cas possible.

Remarque :

On peut avoir une ambiguïté à cause de telles imbrications, voici un exemple :

programme	Interprétation 1	Interprétation 2
<pre>if (A!=B) if (A>B) x=x+1; else x=x+2;</pre>	<pre>if (A!=B) if (A>B) x=x+1; else x=x+2;</pre>	<pre>if (A!=B) if (A>B) x=x+1; else x=x+2;</pre>

En langage C++, le "else" correspond au plus proche **if** qui n'a pas d'**else**.

Donc notre exemple le programme correspond à l'interprétation 2.

Si on veut que le "else" correspond au premier "if" il faut utiliser les accolades "{" }" comme suit :

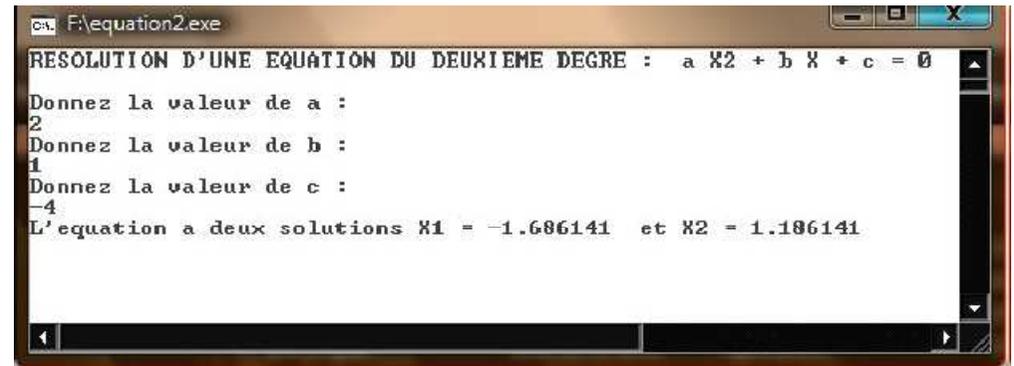
```
if (A!=B)
{
if (A>B)
x=x+1;
}
else
x=x+2;
```

Exercice 1 :

Ecrire un programme C++ qui permet de lire trois variables **A**, **B** et **C** puis affiche ces trois variables (valeurs) dans un ordre croissant.

Exercice 2 :

1) Ecrire un programme C++ qui demande à l'utilisateur les coefficients a,b,c d'une équation deuxième degré $aX^2+bX+c=0$, puis calcule et affiche les solutions de cette équation, l'exécution doit être comme suit :



2) modifier le programme précédant pour qu'il accepte même l'équation du premier degré.

=====**solution**=====

Exercice 1 :

Cet exercice a été fait dans la série de TD 3, il existe deux solutions (1 et 2), j'ai implémenté la deuxième solution, voici le code :

```
#include<iostream>
using namespace std;
    int A, B, C ;
int main () {
cout <<"donnez trois nombres SVP :";
cin >>A>>B>>C;
if ((A<B) && (B<C))
    cout << "l'ordre est:"<<A<<" "<<B<<" "<<C;
else // A>=B ou B>=C
    if ((A<C)&&(C<B))
        cout << "l'ordre est:"<< A<<" "<< C<<" "<< B;
    else
if ((B<A)&&(A<C))
    cout<< "l'ordre est:"<<B<<" " << A<<" "<< C;
    else
if ((B<C)&&(C<A))
    cout << "l'ordre est:"<<B<<" "<< C<<" "<< A;
else
if ((C<A)&&(A<B))
    cout << "l'ordre est:"<<C<<" "<< A<<" "<< B ;
    else
```

```
if ((C<B)&&(B<A))
    cout << "l'ordre est:"<<C<<" "<< B<<" "<< A;
getchar();
getchar();
}
```

Exercice 2:

1)

Équations du second degré → a non nul.

Pour pouvoir résoudre une telle équation, il faut tout d'abord calculer le **discriminant Δ** .

Pour le calculer, il suffit d'appliquer cette formule :

$$\Delta = b^2 - 4ac$$

On le calcule. Ensuite, selon le résultat, on va pouvoir connaître le nombre de solutions qu'il y a, et les trouver s'il y en a.

Si $\Delta < 0$: il n'y a pas de solution.

Si $\Delta = 0$, il y a **une seule solution** à l'équation :

c'est $x = -b/(2a)$

Si $\Delta > 0$ il y a **deux solutions** qui sont :

$$x_1 = (-b - \sqrt{\Delta})/(2a) \text{ et } x_2 = (-b + \sqrt{\Delta})/(2a)$$

Le code est le suivant :

```
#include <iostream>
#include <cmath>
```

```

using namespace std;
int main()
{
// Declaration des variables
float a,b,c; // Les coefficient reels de l'equation
float x1,x2; // x1 et x2 sont les solutions de l'equation
// La solution unique sera stockee dans x1
float delta; // Le discriminant
// Presentation du programme
cout << "RESOLUTION D'une EQUATION DEUXIEME DEGRE :aX2+bX+c=0";
cout << endl;
// Saisie les donnees
cout << "Donnez la valeur de a : ";
cin >> a;
cout << "Donnez la valeur de b : ";
cin >> b;
cout << "Donnez la valeur de c : ";
cin >> c;
if (a == 0)
cout << "Erreur ! Votre equation n'est pas du second degre.";
else
{
delta = b*b - 4*a*c;
if (delta < 0)
cout << "Aucune solution reelle." << endl;
else
if (delta == 0)
{
x1 = -b/(2*a);
cout << "Une seule solution reelle : " << x1;
}
else // Cas ou delta > 0
{
x1 = (-b+sqrt(delta))/(2*a);
x2 = (-b-sqrt(delta))/(2*a);
cout << "l'equation a deux solutions : x1="<< x1 << " ,x2="<<x2 ;
}
}
getchar();
getchar();
}

```

2)

La modification concerne la première condition **if** , on va changer le message d'erreur par les code suivant :

$x1 = -c/b$;

cout<< "c'est une equation du premier degre la solution est : x="<< x1 ;