

# INFO-H-100

## Séance d'exercices 2

Traitement conditionnel

# Bloc

- Bloc = instruction composée
- Regroupement d'instructions

```
{  
    x = 3 ;  
    y = 7 ;  
    z = x * y ;  
}
```

- **Visibilité** : une entité déclarée dans un bloc est visible à partir de sa déclaration et jusqu'à la fin du bloc

# Portée (scope) : exemple

```
int main()
{
    int x = 0;
    // x est visible
    {
        int y = 0;
        // x et y sont visibles
    }
    // x est visible
    // y est invisible
}
```

Portée de x

Portée de y

# Condition

- Expression booléenne
  - Ex :  $x \leq y$ ,  $a == 3$ ,  $i < j$  and  $i \geq k$
- Si sa valeur est vraie, on dit que la condition est vérifiée

a	b	a and b	a or b
true	true	true	true
true	false	false	true
false	true	false	true
false	false	false	false

# Instruction if

- Permet d'exécuter une instruction **si une condition est vérifiée**.

- Exemple :

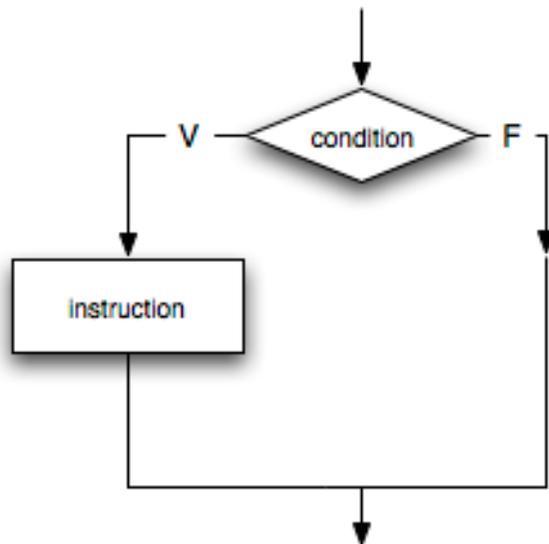
```
if (cote==20)
    cout << "Bravo" << endl;
```

- Convention : **indentation**

# Instruction If

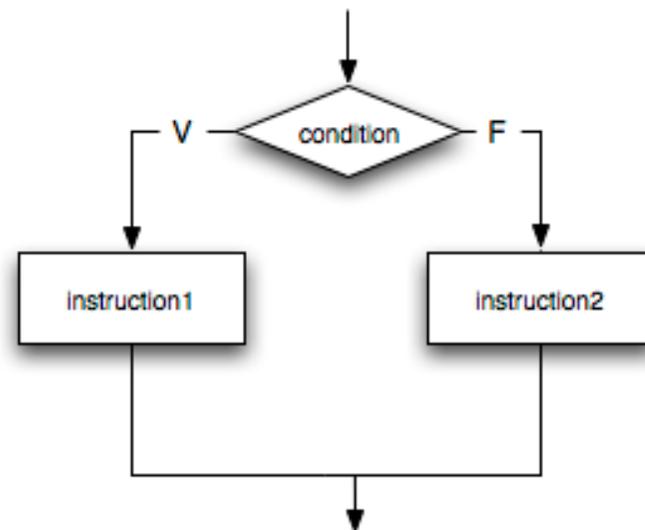
Format 1 :

```
if (condition)  
    instruction
```



Format 2 :

```
if (condition)  
    instruction1  
else  
    instruction2
```



# Instruction If

- Si le traitement conditionnel est une séquence d'actions, regrouper cette séquence dans un **bloc** :

```
if (i==20)
{
    cout << "Bravo" << endl;
    compteur += 1;
}
```

# Compositions

```
if(condition1)
    cout << "cas 1";
else if(condition2)
    cout << "cas 2";
else
    cout << "cas 3";
```

```
if(condition1)
{
    if(condition2)
        cout << "cas a";
}
else
    cout << "cas b";
```

# Erreurs courantes

## Erreurs

- `if(condition) ;  
    instruction;`
- `if(condition)  
    instruction1;  
    instruction2;`
- `if(a = b) ...`
- `if(a == 2) ...  
else if(a != 2) ...`
- `if(a < b < c) ...`

## Corrections

- `if(condition)  
    instruction;`
- `if(condition) {  
    instruction1;  
    instruction2;  
}`
- `if(a == b)...`
- `if(a == 2) ...  
else ...`
- `if(a < b and b < c)`

# Indentation

- Les instructions conditionnelles doivent être indentées (décalées vers la droite).

## Code mal indenté

```
if(a == b)
cout << "a == b" << endl;
else{
if (b == c)
cout << "b == c" << endl;
else if(a == c)
cout << "a == c" << endl;
cout << "a != b" << endl;
}
```

## Code bien indenté

```
if(a == b)
    cout << "a == b" << endl;
else{
    if (b == c)
        cout << "b == c" << endl;
    else if(a == c)
        cout << "a == c" << endl;
    cout << "a != b" << endl;
}
```

# Commentaires

- Sur une ligne :

```
if(cote==20) //si la cote est 20, j'affiche "bravo"  
    cout << "bravo" << endl;
```

- Sur plusieurs lignes :

```
/*  j'inverse les valeurs de  
    a et de b  
*/  
int tmp = a;  
a = b;  
b = tmp;
```

# Exercices

- 8, 10, 12 : sur papier
- 15, 9, 17 : sur machine
- (7, 13)

# Rappel :

## Structure d'un programme

Code source : fichier.cpp

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

int main()
{

    ... instructions ...

}
```

# Rappel : Compilation

- Pour qu'un programme puisse s'exécuter, il faut le compiler.

```
> g++ -Wall fichier.cpp -o programme
```

`-Wall` permet d'afficher les avertissements

- Exécution :

```
> ./programme
```