

University of Batna 2
Institute of safety and security
1st year LMD

Chapter 3: Conditional statements

Introduction:

- A conditional instruction is an instruction which allows you to carry out a test (condition) and to execute or not a group of instructions depending on the value of the test carried out. The condition is a Boolean expression (which evaluates to either true or false).
- There are 3 forms:
 1. simple conditional: **if**
 2. alternative conditional: **if else**
 3. multiple choice conditional: **switch**

Simple conditional: if

- If we have code that we sometimes want to execute and sometimes we want to skip we can use the if statement.

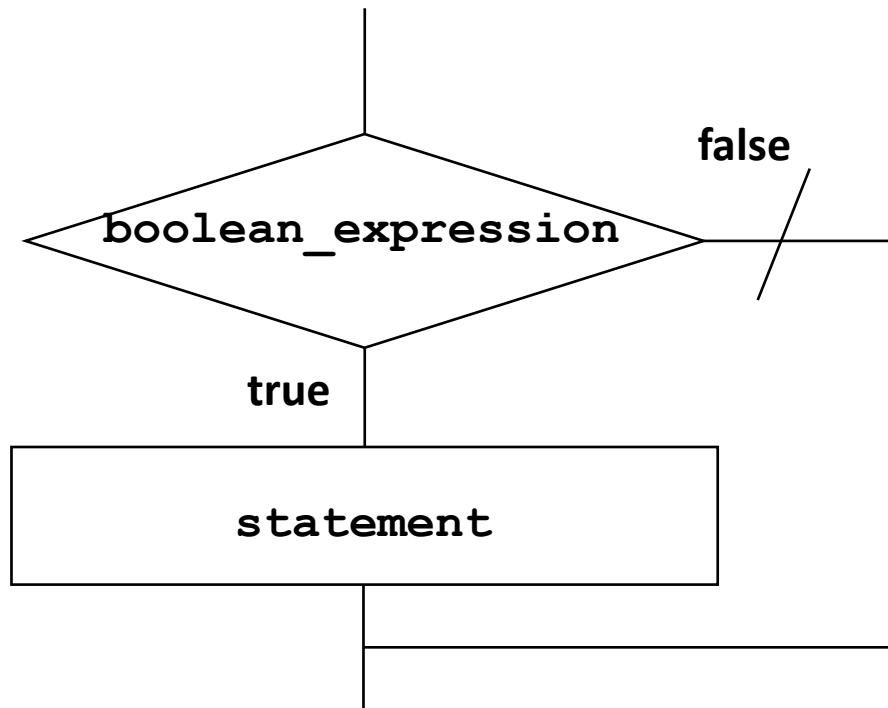
- **Syntaxe :**

```
if (boolean_expression) {  
    statement;  
}
```

- If `boolean expression` evaluates to true, then `statement` is executed.
- If `boolean expression` evaluates to false, then `statement` is skipped.
- Note that the `boolean expression` enclosed in parentheses must evaluate to true or false.

Simple conditional: if (2)

- The if statement flowchart:



Simple conditional: if (3)

- **Example:**

```
#include<iostream>
using namespace std;
int main(){
    int n;
    cout<<"please enter an integer number:";
    cin>> n;
    if(n>10){
        cout<<"The number entered is greater than 10 "<<endl;
    }
    cout<<"End of the program!";
    return 0;
}
```

Simple conditional: if (3)

- **Example:**

```
#include<iostream>
using namespace std;
int main(){
    int n;
    cout<<"please enter an integer number:";
    cin>> n;
    if(n>10){
        cout<<"The number entered is greater than 10 "<<endl;
    }
    cout<<"End of the program!";
    return 0;
}
```

Alternative conditional Instructions :

- If we want to choose between two alternative we use the if/else statement:

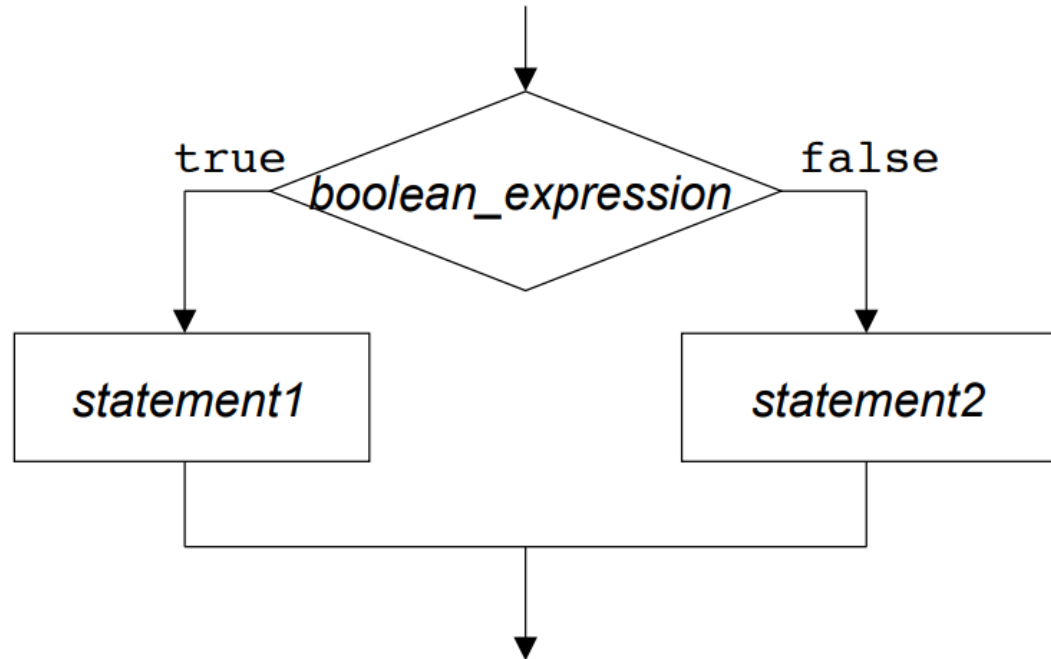
Syntaxe :

```
if (boolean_expression) {  
    statement1  
}else{  
    statement2  
}
```

- If boolean expression evaluates to true, then statement1 is executed.
- If boolean expression evaluates to false, then statement2 is executed.

Alternative conditional Instructions(2) :

- The if else flowchart:



Alternative conditional Instructions(3) :

- **Example:**

```
#include<iostream>
using namespace std;
int main(){
    int n;
    cout<<"Please print an integer:";
    cin>> n;
    if(n>0){
        cout<<"Positive number"<<endl;
    } else{
        cout<<"Negative number"<<endl;
    }
    return 0;
}
```

Multiple Alternatives

- A test with if therefore opens two paths, corresponding to two different treatments. But in some cases, these two paths are not enough for all possible solutions.

Syntaxe :

```
if (boolean_expression1) {
    statement1;
}else if (boolean_expression 2) {
    statement2;
}
}else {
    statement3;
}
```

-

The selective statement (switch statement)

- The switch statement objective is to check several possible constant values for an expression. Something similar to what we did at the beginning of this section with the concatenation of several if and else if instructions. Its form as the following:

- **Syntaxe :**

```
switch (identifier)
{
  case c1:statement1;break;
  case c2:statement2;break;
  case c3:statement3;break;
  ...
  default: statement;break;
}
```

Exercice:

- Write a program that allows you to Solve a first-degree equation: $ax+b=0$, where a and b are real numbers given by the user.