If condition

If condition

REFERENCES

- Dr. Mostafa Saad
  - Logical Operators

- Dr. Mohamed El-Desouki
  - If Condition and Logical Operators

- Adel Nasim
  - Logical Operators
  - If condition

If condition

Normally, statements in a program are executed one after the other in the order in which they're written

- **This is called sequential execution.**

So how can we choose specific statements to execute and other not?

- Here comes the answer

The control statements are categorized in almost three groups:

1 - Selection control statements.
2 - Repetition control statements
3 – Jump statements

We will use the selection control statements to solve our problem.

If condition

Selection statements are used to choose among alternative courses of action.

- For example

Suppose the passing mark on an exam is 50.

➢ The pseudocode statement :

If student's mark is greater than or equal to 50 Then

Print "Passed".

If condition

1) Relational Operators

operator	description
==	Equal to
!=	Not equal to
<	Less than
>	Greater than
<=	Less than or equal to
>=	Greater than or equal to

If condition

Our computer use 0 for saying that expression is FALSE or 1 for saying that expression is TRUE

➢ Let's see some examples

Cout<<3>5; → Is 3 greater than 5? False so it's like printing "0"

Cout<<5>3; → Is 5 greater than 3? True so it's like printing "1"

Cout<<3==3; → Is 3 equal 3? True so it's like printing "1"

Let's see the code and it's result

If condition

05_01.cpp

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

int main() {
	cout << (3 > 5) << "\n";
	cout << (3 < 5) << "\n";
	cout << (3 == 5) << "\n";
	cout << (3 >= 5) << "\n";
	cout << (3 >= 3) << "\n";
	cout << (3 == 3) << "\n";
	cout << (3 > 1) << "\n";
	cout << (3 != 4) << "\n";
	cout << (3 != 3) << "\n";

	return 0;
}
```

Console | Problems | Tasks | Pr

<terminated> ztemp [C/C++ Application] /ho

```
0
1
0
0
1
1
1
1
0
```

If condition

- We can also use variables

05_02.cpp

```
1  #include<iostream>
2  using namespace std;
3
4  int main() {
5      int x, y;
6      x = 3, y = 5;
7
8      cout << (x > y) << "\n";
9      cout << (x < y) << "\n";
10     cout << (x == y) << "\n";
11     cout << (x >= y) << "\n";
12
13     return 0;
14 }
15
```

Console | Problems | Tasks

```
<terminated> ztemp [C/C++ Application] /ho
0
1
0
0
```

If condition

2)Logical Operators

&& (AND)

X	Y	X && Y
TRUE	FALSE	FALSE
FALSE	TRUE	FALSE
TRUE	TRUE	TRUE
FALSE	FALSE	FALSE

|| (OR)

X	Y	X || Y
TRUE	FALSE	TRUE
FALSE	TRUE	TRUE
TRUE	TRUE	TRUE
FALSE	FALSE	FALSE

If condition

1)And logical operators

- Let say Mostafa is 30 years old man and his salary = 7000

- Mostafa > 25 years and salary < 8000? True
- Mostafa > 27 years and salary > 9000? False
- Mostafa > 35 years and salary < 8500? False
- Mostafa > 35 years and salary > 9000? False

Summary

- Only 1 case is true when both conditions are true

If condition

2)OR logical operators

- Let say Mostafa is 30 years old man and his salary = 7000.

- Mostafa > 25 years or salary < 8000? True
- Mostafa > 27 years or salary > 9000? True
- Mostafa > 35 years or salary < 8500? True
- Mostafa > 35 years or salary > 9000? False

Summary

- Only 1 case is false when both conditions are false.

If condition

Precedence of logical operators

Precedence: means what to apply first. Here we apply the negation (NOT) then AND, and finally OR.

Operator	precedence
!	High
&&	Medium
||	Low

A || B && C	means	A || (B && C)
A && B || C && D	means	(A && B) || (C && D)
A && B && C || D	means	((A && B) && C) || D
!A && B || C	means	((!A) && B) || C

- **Advanced**

What if I want to force specific priority? Use (), Every group of () is computed first.

If condition

Mixing Logical Operators in C++

Let say Mostafa is 30 years old, salary = 7000 and weight = 110 kg

- Mostafa > 35 years or salary > 6000 or weight > 200kg? True
- Mostafa > 35 years and salary > 6000 or weight > 200kg? False

➢ Reduce every subgroup of ANDS first

1. F or T or … ⇒ T
2. F and … or F ⇒ F

If condition

- GUESS THE OUTPUT

05_6.cpp

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

int main() {
    int age = 30, salary = 7000, weight = 110;

    // ANDs are evaluated
    cout << ( age > 35 || salary > 6000  && weight > 200) << "\n";

    // () are evaluated FIRST even before some ANDS
    cout << ((age > 35 || salary > 6000) && weight > 200) << "\n";

    return 0;
}
```

If condition

- In c++ the syntax of IF statement is

```
if ( Expression)
    action statement ;
```

```
if ( Expression)
 {
    action statement 1 ;
    action statement 2 ;
    .
    .
    action statement n ;
}
```

- The Expression can be any valid expression including a relational expression and even arithmetic expression

- In case of using arithmetic expressions , a non-zero value is considered to be true, whereas a 0 is considered to be false

```
if ( grade >= 50 ) cout<<"Passed";
```

If condition

Write a program that accept an integer from the user and in case this integer is even print out the following message “Number is even ” .


```
#include <bits/stdc++.h>
using namespace std ;
int main()
{
  int number ;
  cin >> number;
  if (number % 2 == 0)
      cout << "Number is even\n";
  return 0 ;
}
```

If condition

IF .. Else Statement

The IF...Else selection statement allows you to specify that there is a course of actions are to be performed when the condition is true and another course of actions will be executed when the condition is false.

- For example
  - The pseudocode statement

If student's mark is greater than or equal to 50
Print "Passed"
Else
Print "Failed"

If condition

In C++ The syntax for the IF…Else statement

```
if ( Expression)
    action statement ;
Else
    action statement ;
```

```
if ( Expression)
  {
    action statements 1 ;
    .
    action statement n ;
}
Else
{
action statements 1 ;
    .
action statement n ;
}
```

```
#include <bits/stdc++.h>
using namespace std ;
int main()
{
  int grade ;
  cin >> grade;
  if ( grade >= 50 )
      cout <<"Passed\n";
  else
      cout <<"failed\n";
  return 0 ;
}
```

If condition

- Exercise

write a program that accept an integer from the user and print out whether it is Positive or Negative number

If condition

➢ ANSWER

```
#include<iostream>
using namespace std;
int main(){
    int x;
    cin>>x;
    if(x>0)
        cout<<"positive";
    else
        cout<<"negative";
    return 0;
}
```

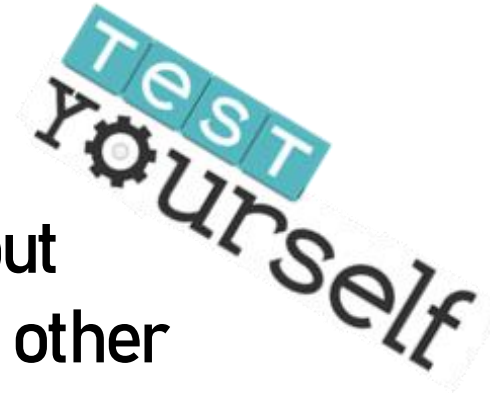
If condition

If-else if statement

```
•if (condition)
      {
      //Statements
      }
•
else if (condition) {
            //Statements
      }
•
else {
            //Statements
      }
```

If condition

➢ Example



write a program that ask the user to Enter2 numbers and print out whether they are equal or there is one which is greater than the other

```
using namespace std;
int main() {
int num1, num2;
cout << "Enter Number 1 , Number2  \n";
cin >> num1 >> num2;
if (num1 == num2)
cout << "Both Are Equal \n";
else if (num1 > num2)
cout << "Number 1 is greater than number 2 \n";
else
cout << "Number 2 is greater than number 1 \n";
}
```

If condition

➢ practice

- print out the student grade according to the following formulas

  - A for exam marks greater than or equal 90 and less than or equal 100 ,
  - B for exam marks greater than or equal 80 and less than 90 ,
  - C for exam marks than or equal to 70 and less than 80 ,
  - D for exam marks than or equal to 60, and less than 70 ,
  - F for all other marks

If condition

➢ Answer

```
#include<iostream>
using namespace std;
int main(){
    float grade;
    cin>>grade;
    if(grade>=90&&grade<=100)
        cout<<"A";
    else if(grade>=80&&grade<90)
        cout<<"B";
    else if(grade>=70&&grade<80)
        cout<<"C";
    else if(grade>=60&&grade<70)
        cout<<"D";
    else
        cout<<"F";
    return 0;
}
```

If condition

➢ Practice

- Given two numbers and a sign between them which will indicate if the user want the addition, subtraction, division or multiplication of these two numbers

  ➢ find the value of the answer.

  Inputs ⇒ outputs

  - 7 + 55 ⇒ 62
  - 7 * 10 ⇒ 70

If condition

➢ Answer

```
#include<iostream>
using namespace std;
int main(){
 float num1,num2;
    char op;
    cin>>num1>>op>>num2;
    if(op=='+')
        cout<<num1+num2;
    else if(op=='-')
        cout<<num1-num2;
    else if(op=='/'){
        if(num2==0)
            cout<<"cant be solved";
        else
        cout<<num1/num2;}
    else
        cout<<num1*num2;
    return 0;
}
```

If condition

- To be more advanced
  - INCREMENT & DECREMENT OPERATOR ++,--

- C++ also provides increment and decrement operators ++ and -- respectively.
- ++ increment the value by 1
- -- decrement the value by 1
- a++ is like a=a+1;
- a-- is like a=a-1;

If condition

- int a=5;
- a++;//a=6
- --a;//a=5
- int b=6;
- ++b;//b=7
- b=18;
- b--;//b=17
- --a;//a=5

If condition

- Pre,post- Increment & pre,post-decrement Operator ++,--

Pre-increment operator: operator used to increment the value of a variable before using it in an expression. In the Pre-Increment, value is first incremented and then used inside the expression.

```
int x=5;
a=++x;
cout<<a;//a=6
cout<<x; //x=6
int b=7;
int c=--b;
cout<<c;//c=6
cout<<b;//b=6
```

If condition

Post-increment operator: operator used to increment the value of the variable after executing the expression completely in which post-increment is used.

```
int x=5;
a=x++;
cout<<a;
cout<<x;
a=5
x=6
int b=7;
int c=b--;
cout<<c; cout<<b;
c=7
b=6
```

If condition

➢ GUESS THE OUTPUT

```
#include<iostream>
using namespace std;
int main() {
int x, y = 2;
if(x = y % 2)
cout << 1;
else
cout << 0;
return 0;
}
```

If condition

➢ GUESS THE OUTPUT

```
#include<iostream>
using namespace std;
int main() {
int x = 0;
if(++x)
cout << 1;
else
cout << 0;
return 0;
}
```

If condition

➢ GUESS THE OUTPUT

```
#include<iostream>
using namespace std;
int main() {
int x = 0;
if(x++)
cout << 1;
else
cout << 0;
return 0;
}
```

If condition

➢ GUESS THE OUTPUT

```
#include<iostream>
using namespace std;
int main() {
int x = 1, y = 0, z = 1;
if(++x && y++ && z)
cout << 1;
else
cout << 0;
return 0;
}
```

If condition

GUESS THE OUTPUT

1. OUTPUT=0
2. OUTPUT=1
3. OUTPUT=0
4. OUTPUT=0

Thank You