

## Conditional control structure

Conditional Control Structure is organized in such a way that there is always a condition that has to be evaluated first. The condition will either evaluate to a true or false.

Two type to implement:

1. if statement (including if-else and nested if)
2. switch case statement

### Operators

#### *Conditional Operators*

- use for expressions that evaluates to true or false.

Operator	Use	Result
>	op1 > op2	true if op1 is greater than op2
>=	op1 >= op2	true if op1 is greater or equal to op2
<	op1 < op2	true if op1 is less than op2
<=	op1 <= op2	true if op1 is less or equal to than op2
==	op1 == op2	true if op1 is equal to op2
!=	op1 != op2	true if op1 is not equal to op2

#### *Logical Operators*

- used in Boolean expressions and consists of logical "and", "or" and "not".

Operator	Use	Result
&&	op1 && op2	true if op1 and op2 are both true
	op1    op2	true if either op1 or op2 is true
!	!op1	op1 is false if its original value is true and vice versa

### TYPES OF IF STRUCTURES

1. if (single alternative)
2. if-else (double alternatives)
3. if-elseif-else (multiple alternatives)
4. Nested if

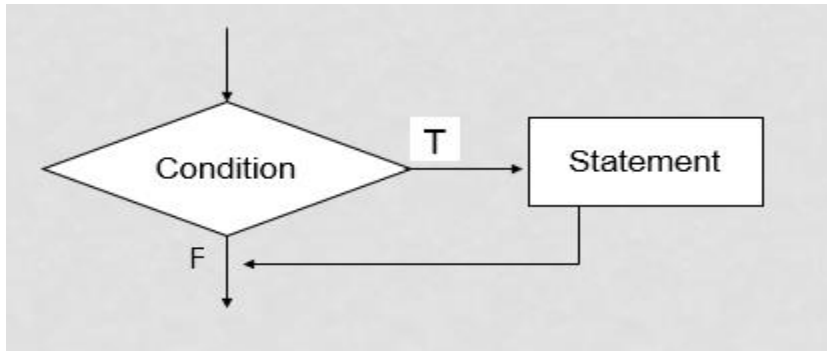
## If (single alternative)

Definition:

It performs an indicated action only when the condition is *true*, otherwise the action is skipped.

Syntax: **if** (<expression>){

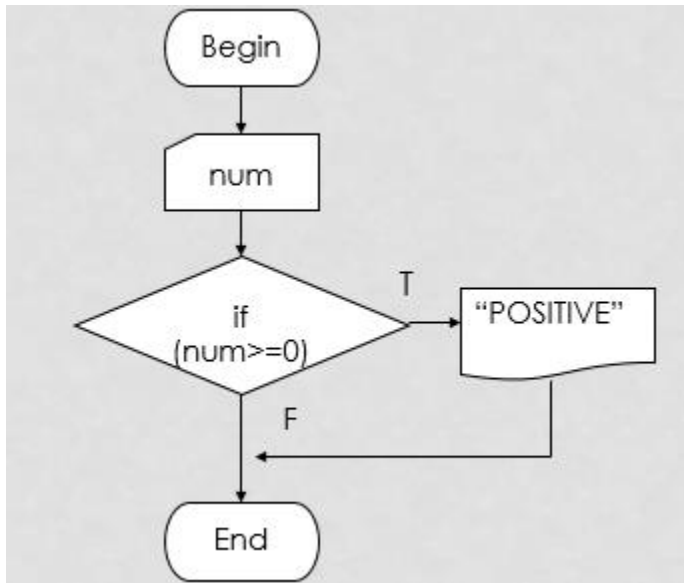
<statement>}



Example:

1. Write a program that will allow the user to input an integer value. If the value is greater than or equal to zero, print the word "POSITIVE".

Flowchart:



Program:

```
namespace IfExample1
{
    class Program
    {
        static void Main(string[] args)
        {
            //Write a program that will allow the user to input an integer value.
            //If the value is greater than or equal to zero, print the word "POSITIVE".
            int num;

            num = Convert.ToInt32(Console.ReadLine());
            if (num >= 0){
                Console.WriteLine("POSITIVE");
            }

            Console.ReadKey();
        }
    }
}
```

2. Write a program to input two integers. Thereafter, the program should determine if these two numbers are equivalent. If they are equivalent, print the word EQUIVALENT.

Program:

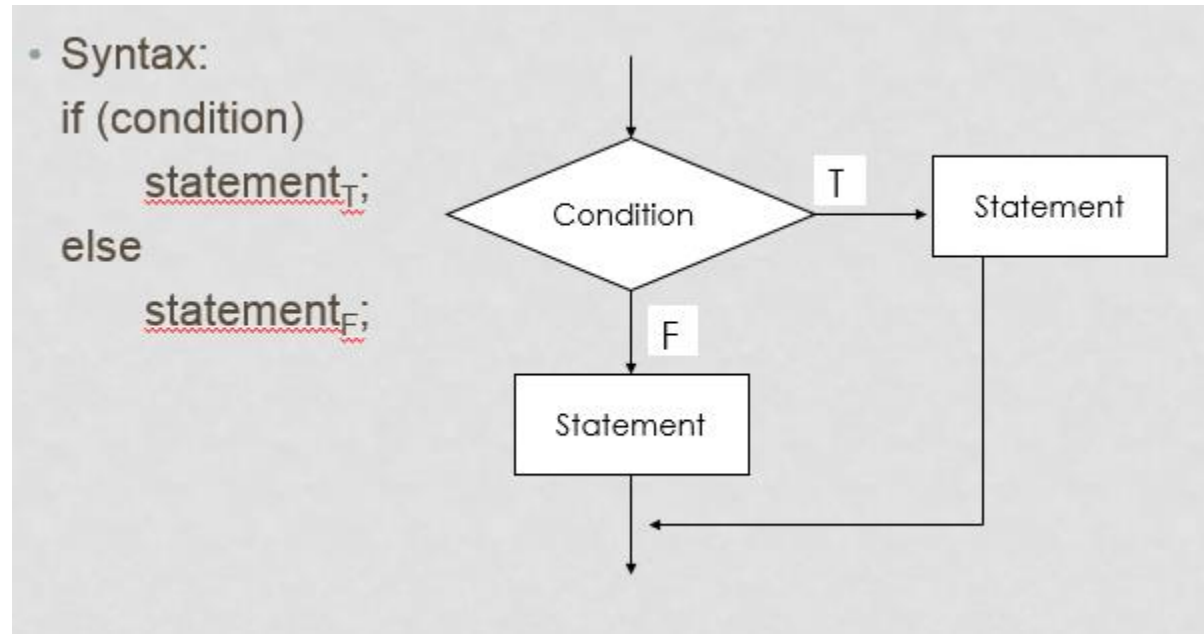
```
namespace IfExample2
{
    class Program
    {
        static void Main(string[] args)
        {
            /* Write a program to input two integers. Thereafter, the program should
            determine if these two numbers are equivalent.
            If they are equivalent, print the word EQUIVALENT.*/

            int num1, num2;
            num1 = Convert.ToInt32(Console.ReadLine());
            num2 = Convert.ToInt32(Console.ReadLine());

            if (num1 == num2) {
                Console.WriteLine("EQUIVALENT");
            }
            Console.ReadKey();
        }
    }
}
```

## If-else (double alternatives)

- It allows the programmer to specify that different actions are to be performed when the condition is true than when the condition is false.
- If condition evaluates to true, then statement true is executed and statement false is skipped; otherwise, statement true is skipped and statement false is executed.



## Good Programming Practice

- Indent both body statements of an if...else statement.
- If there are several levels of indentation, each level should be indented the same additional amount of space.

## Example:

3. Write a program to input two integers. Thereafter, the program should determine if these two numbers are equivalent. If they are equivalent, print the word EQUIVALENT otherwise print the word NOT EQUIVALENT.

```
namespace IfExample3
{
    class Program
    {
        static void Main(string[] args)
        {
            int num1, num2;

            num1 = Convert.ToInt32(Console.ReadLine());
            num2 = Convert.ToInt32(Console.ReadLine());
```

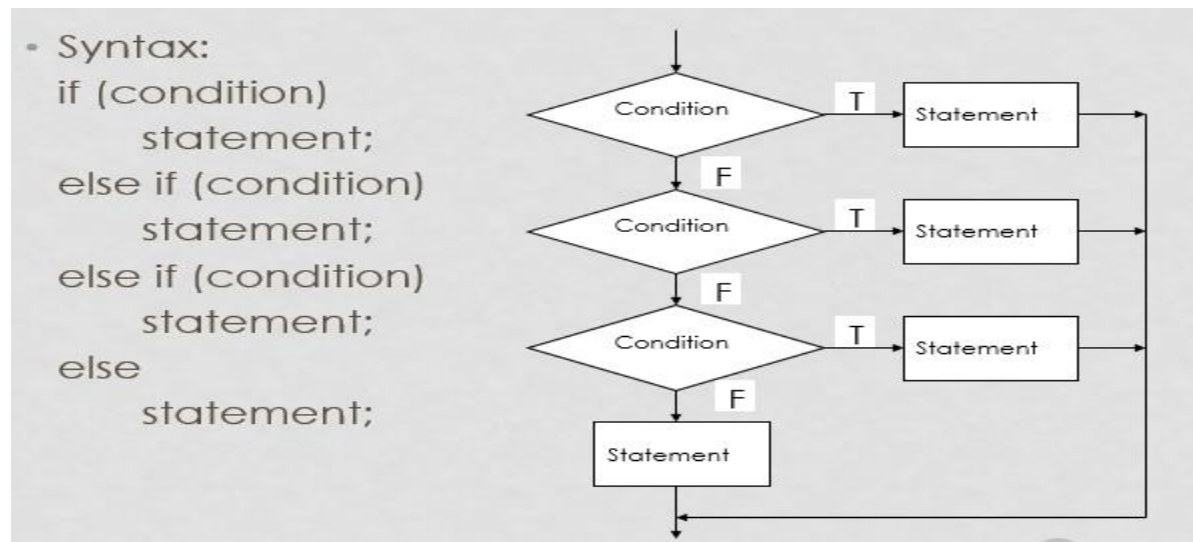
```

// boolean expression that test if the two numbers are equal
if (num1 == num2)
{
    Console.WriteLine("EQUIVALENT");
}
else {
    Console.WriteLine("NOT EQUIVALENT");
}
Console.ReadKey();
}
}
}

```

### If-elseif-else (multiple alternatives)

- The conditions in a multiple-alternative decision are evaluated in sequence until a true condition is reached.
- If a condition is true, the statement following it is executed, and the rest of the multiple-alternative decision is skipped.
- If a condition is false, the statement following it is skipped, and the condition next is tested. If all conditions are false, then the statement following the final else is executed.



Example:

4. Make a program that will accept a score and display based on the following conditions:

Score	Display
86 – 100 (inclusive)	Very Good
75 – 85 (inclusive)	Fair
Below 75	Failed

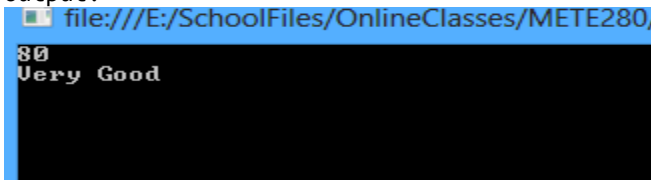
Program:

```
namespace IfExample4
{
    class Program
    {
        static void Main(string[] args)
        {
            int score;

            score = Convert.ToInt32(Console.ReadLine());

            // this boolean expression satisfy the condition 86 - 100 (inclusive)
            if (score >= 86 && score <= 100)
            {
                Console.WriteLine("Very Good");
            }
            // this boolean expression satisfy the condition 75 - 85 (inclusive)
            else if (score >= 75 && score <= 85)
            {
                Console.WriteLine("Very Good");
            }
            // this boolean expression satisfy the condition below 75
            else if (score > 75 && score <= 0 )
            {
                Console.WriteLine("Very Good");
            }
            else // when score is not between 0-100
            {
                Console.WriteLine("Wrong Output");
            }
            Console.ReadKey();
        }
    }
}
```

Output:



```
file:///E:/SchoolFiles/OnlineClasses/METE280
30
Very Good
```

## Nested-if

- **Syntax:**

```
if (condition)
{
    if (condition)
        statement;
    else
        statement;
}
```

Example:

5. XYZ Telegram Company charges P18.50 for a telegram that does not exceed 12 words and P1.50 for every succeeding word, plus P5 service charge if type of delivery is special. Create a program that accepts the number of words in a telegram and the type of delivery 'S' for special if the customer desires for a special delivery. Compute and display the customer's charge.

Program:

```
namespace IfExample5
{
    class Program
    {
        static void Main(string[] args)
        {
            double charge = 0;
            char deltype;
            int numOfWorks;

            Console.WriteLine("Input number of words and delivery type:");
            numOfWorks = Convert.ToInt32(Console.ReadLine());
            deltype = Convert.ToChar(Console.ReadLine());

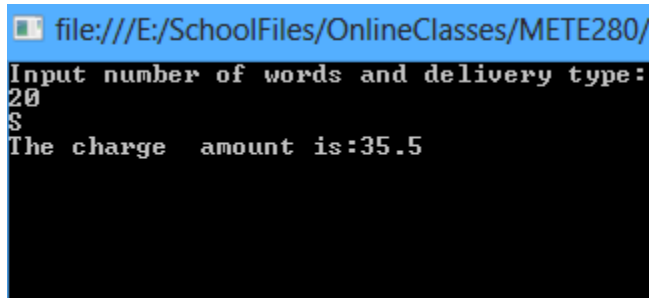
            // this Boolean expression test if number of words is from 0 - 12
            if (numOfWorks > 0 && numOfWorks <= 12) {
                if (deltype == 'S' || deltype == 's')
                {
                    charge = 18.50 + 5;
                }
                else {
                    charge = 18.50;
                }
            }
            //this Boolean expression test if number of words is from 13 and more
            else if (numOfWorks > 12)
            {
                int exceedword = numOfWorks - 12;
                if (deltype == 'S' || deltype == 's')
                {
                    charge = 18.50 + (exceedword * 1.50) + 5;
                }
                else
                {
                    charge = 18.50 + (exceedword * 1.50);
                }
            }
            //this is the option if input from the user is wrong like a negative value
            else {
                Console.WriteLine("Input Error!");
            }
        }
    }
}
```

```
// '+' operator concatenate the string and the value of charge w/c automatically
converted to string
    Console.WriteLine("The charge amount is:" + charge);

    Console.ReadKey();

}
}
```

Output:

A screenshot of a console window with a black background and white text. The window title bar shows the file path: file:///E:/SchoolFiles/OnlineClasses/METE280/. The text in the console reads: "Input number of words and delivery type:", followed by the user input "20", then "S", and finally the program output "The charge amount is:35.5".

```
file:///E:/SchoolFiles/OnlineClasses/METE280/
Input number of words and delivery type:
20
S
The charge amount is:35.5
```

#### Good Programming Practice

- A nested if...else statement can perform much faster than a series of single-selection if statements because of the possibility of early exit after one of the conditions is satisfied.
- In a nested if...else statement, test the conditions that are more likely to be true at the beginning of the nested if...else statement. This will enable the nested if...else statement to run faster and exit earlier than testing infrequently occurring cases first.
- Always putting the braces in an if...else statement (or any control statement) helps prevent their accidental omission, especially when adding statements to an if or else clause at a later time. To avoid omitting one or both of the braces, some programmers prefer to type the beginning and ending braces of blocks even before typing the individual statements within the braces.
- Forgetting one or both of the braces that delimit a block can lead to syntax errors or logic errors in a program.
- Placing a semicolon after the condition in an if statement leads to a logic error in single-selection if statements and a syntax error in double-selection if...else statements (when the if part contains an actual body statement).

## Switch-case statement

- The controlling expression, an expression with a value of type **int** or type **char**, is evaluated and compared to each of the case labels in the case constant until a match is found.
- A case constant is made of one or more labels of the form case followed by a constant value and a colon.
- When a match between the value of the controlling expression and a case label value is found, the statement following the case label are executed until a break statement is encountered. Then the rest of the switch statement is skipped.

- **Syntax:**

```
switch (controlling expression)
{
    case constant: statement;
                    break;
    case constant: statement;
                    break;
    . . .
    default:      statement;
                    break;
}
```

### Break

- exits from a certain block of code

Example:

6. Count Dracula buys blood from the public. But since he prefers certain types to others, he pays as follows:

Type	Rate/liter
A	P 3000.00
B	P 1800.75
C (for AB)	P 2880.25
O	P 1500.00

Write a program to input a client's name, his/her blood type, and the volume (liter) of blood extracted. Output how much each customer will be paid.

Program:

```
namespace switchExample6
{
    class Program
    {
        static void Main(string[] args)
        {
            char type;
            double volume, bill = 0;

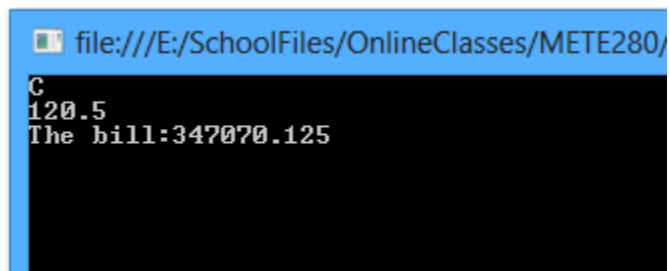
            type = Convert.ToChar(Console.ReadLine());
            volume = Convert.ToDouble(Console.ReadLine());
            //variable to test
            switch(type){

                case 'A': bill = volume * 3000;
                    break;
                case 'B': bill = volume * 1800.75;
                    break;
                case 'C': bill = volume * 2880.25;
                    break;
                case 'O': bill = volume * 1500;
                    break;
                default: Console.WriteLine("Wrong Input");
                    break;

            }

            Console.WriteLine("The bill:" + bill);
            Console.ReadKey();
        }
    }
}
```

Output:



A screenshot of a console window with a blue title bar. The title bar text is partially visible as "file:///E:/SchoolFiles/OnlineClasses/METE280/". The console output shows the character 'C' on the first line, the number '120.5' on the second line, and the text "The bill:347070.125" on the third line.

## Practice Exercises

1. Write a program that will allow the user to input a number. Print the word NEGATIVE if the number is a negative value.
2. Write a program that will input an integer and determine if it is an even number.
3. In order to discourage excess electric consumption, an electrical company charges its customers a lower rate of P75 for the first 250 kilowatt-hours and a higher rate of P85 for each additional kilowatt-hour. In addition, a 10% surtax is added to the final bill. Write a program that calculates the electrical bill given the number of kilowatt-hours consumed as input. At the end, print the number of kilowatt-hours consumed and the computed bill.
4. The amount of miscellaneous fees a student pays during enrollment is determined by his nationality, year level and sex. A flat fee of P200 is paid by all students. Male seniors pay an additional P100 for CMT. Female freshmen and sophomores pay an additional P75 for girl scouting. Foreigners pay an additional P200 as alien fees. Make a program that would input NATIONALITY(1 for Filipino, 2 for Foreigner), SEX (1 for male, 2 for female) and YEAR level (from 1-4) and output miscellaneous FEES.

## Exercises (Graded)

Create one folder to save all the following exercise below.

1. A company selling household appliances gives commissions to its salesman determined by the kind of product sold as well as the sales amount.
  - Type 1: 7% of sale or 400, whichever is more.
  - Type 2: 10% of sale or 900, whichever is less.
  - Type 3: 12% of sale.
  - Type 4: P250, regardless of sale price.Make a program that would input the KIND of appliance sold (between 1-4) and the sale PRICE (a positive floating-point value), and output the COMMISSION that the salesman will receive.
- 2.

## Procedure Proper:

1. Make a C program that computes the total charge of a certain registered caller that uses the gateway of the telephone company (DEEPCOMMUNICATIONS) that caters continental and cross continental calls. The company has the following tabulated data of charges for the continental and cross continental calls.

<i>Region Code</i>	<i>Region</i>	<i>Day Time Rate (time code = 1)</i>	<i>Night Time Rate (time code = 2)</i>
1	Asia	18.74/min	14.35/min
2	Australia	21.36/min	19.41/min
3	Antarctica	68.42/3 mins	65.25/3 mins
4	North America	20.61/min	18.23/min
5	South America	21.22/min	20.13/min
6	Africa	59.65/3 mins	57.32/3 mins
7	Europe	87.38/4 mins	84.72/4 mins

### Note:

- ✓ Display on the screen the regions and each corresponding region code, time and each corresponding time code prior to inputting the necessary values.
- ✓ Follow the screen output formats (see screenshots below).
- ✓ Do not include header files if none of its functions is catered for use.

