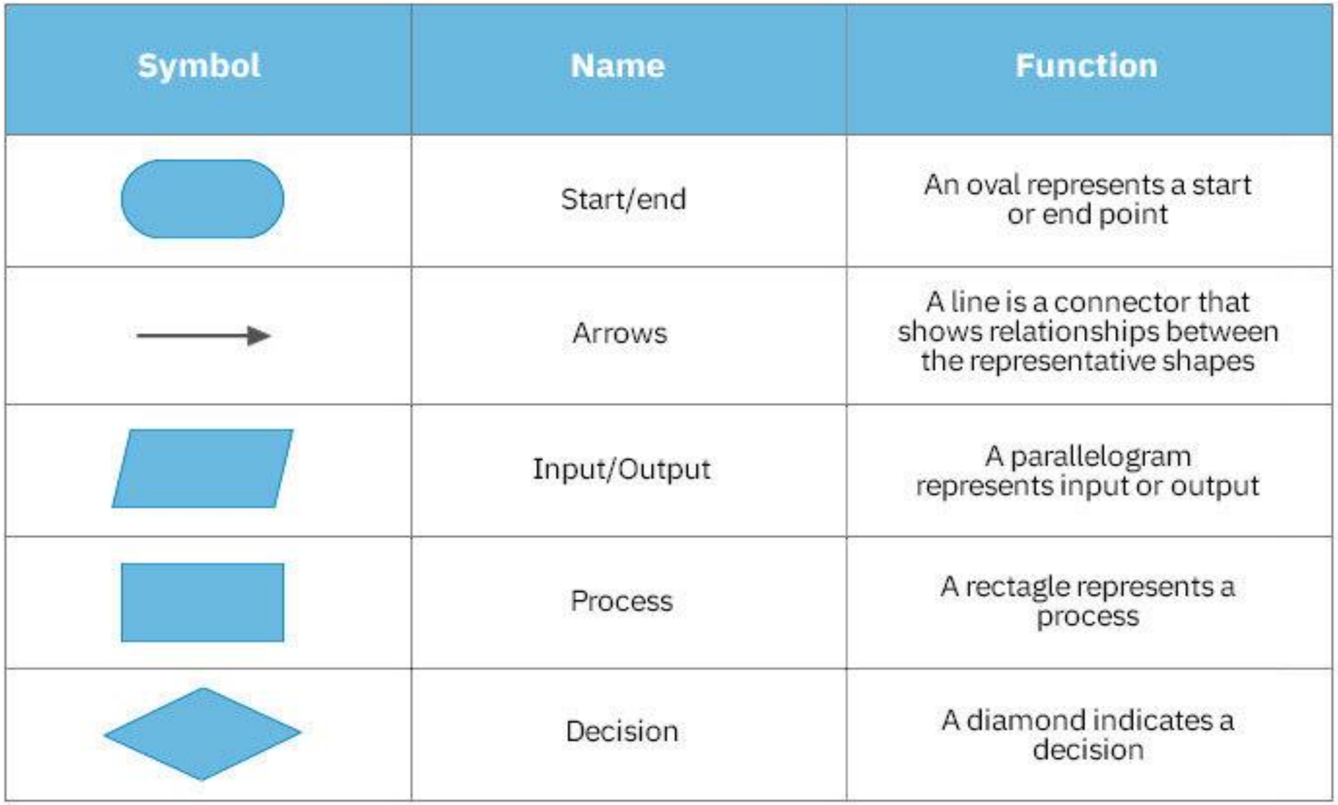
Control Structures

Ms. Kanwal Lodhi

Flowchart



Symbol	Name	Function
⬭	Start/end	An oval represents a start or end point
→	Arrows	A line is a connector that shows relationships between the representative shapes
▱	Input/Output	A parallelogram represents input or output
▭	Process	A rectagle represents a process
◆	Decision	A diamond indicates a decision

Control structures

- Control structure control the flow of program execution.
- The statements of a computer program are always executed one after another in the order of its sequence. However, the order of execution in the program can be change. This is done with the help of control structures .
- A control structure is like a block of programming that analyses variables and chooses a direction in which to go based on given parameters. The term flow control details the direction the program takes. Hence it is the basic decision-making process in computing.
- There are three type of control structure: Sequence, Selection and Repitition.

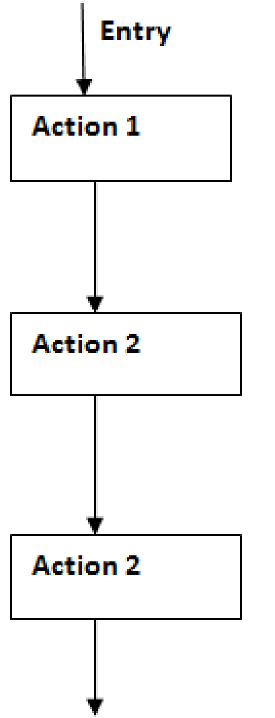
Sequence structures

- Sequence :
  - A sequence control structure uses a compound statements or blocks to specify sequential flow.
  - {

    Statement 1;

    Statement 2;

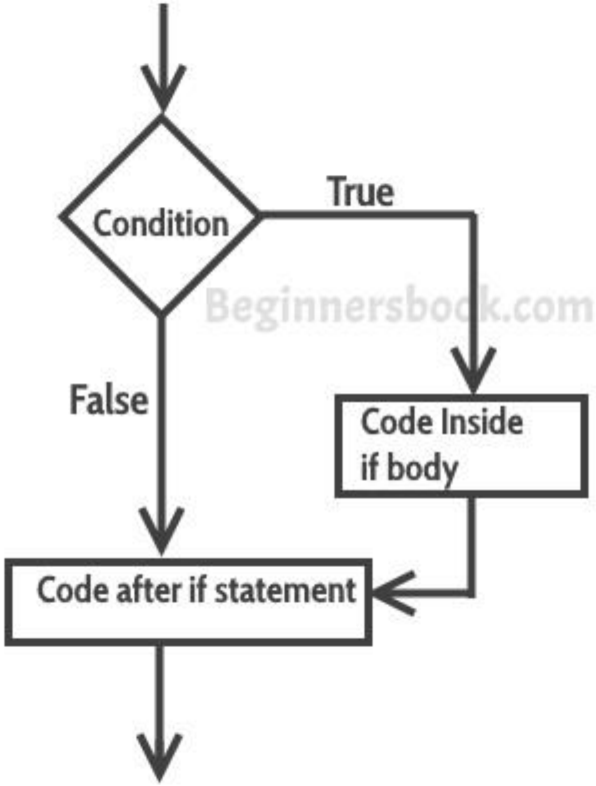
    ……… }
  - A set of instruction is executed in sequence from top to bottom.
  - Statement executed one after the other in order of their sequential execution.



Sequence Structure

The “if” statement

- The “if” statement
  - The if statement is used to execute or ignore a set of statements after testing a condition.
  - The if statement evaluates the condition. If the given condition is true the statement or set of statements under the ‘if statement’ is executed or if its false the statement or set of statements will be ignored.
  - Syntax
    - if(condition)
      statement-1 ;   \\ for single statement
    - if(condition)
      {statement-1;
      Statement-2; ……}   \\ for set of statements



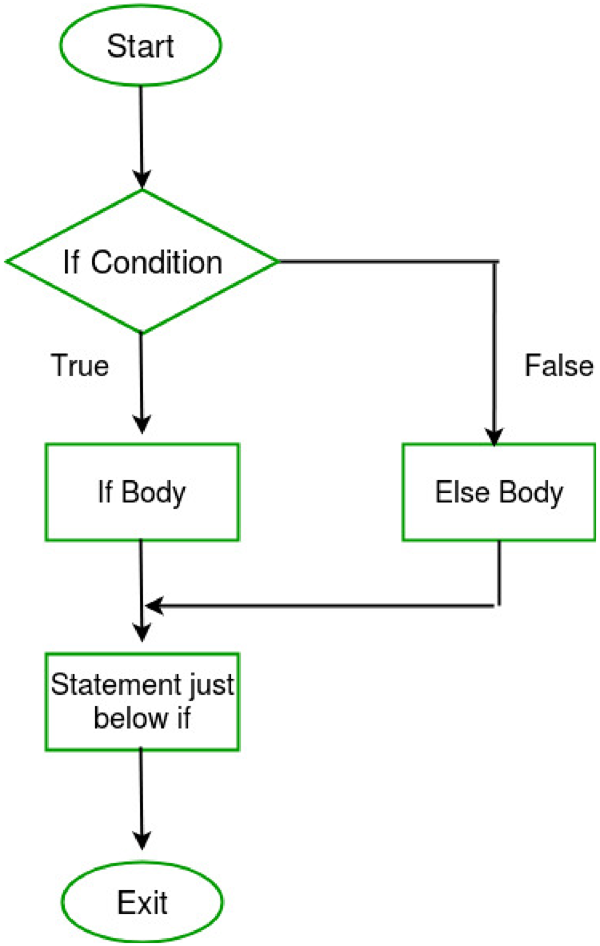
if-else statement

- if-else statement
  - It is used for making two way decision. In computer programming, we use the if...else statement to run one block of code under certain conditions and another block of code under different conditions.
  - In this statement, one condition and two block of statements are given and any one block is executed after evaluating the condition.
  - Syntax:
    - if(condition)

```
if(condition)
     statement-1;
else
     statement-2;
```

  - Syntax:

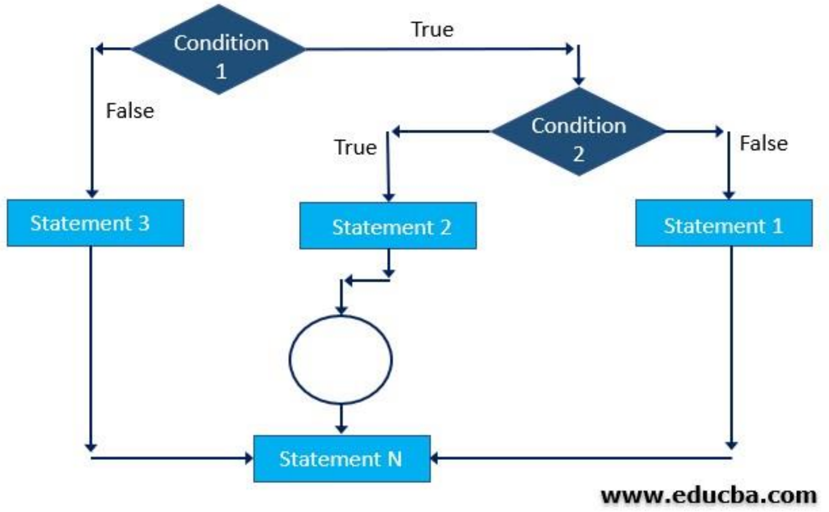
```
if(condition)
        {statement-1;
        Statement-2; ……}      \\ first block of
            statements
    else
        {statement-1;
        Statement-2; ……}      \\ second block of
            statements
```



Nested if statement

- When “if statement” is used within another “if statement”, it is called nested if statement. It is used for multi way decision making.
- Syntax:

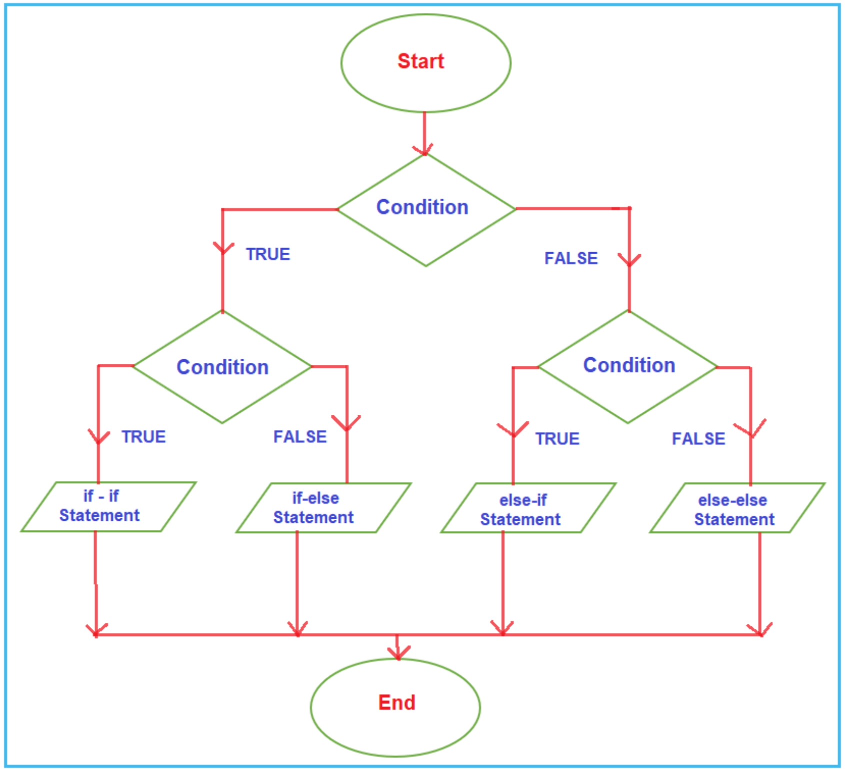
```
if(condition-1)
{      if(condition-2)
          {   statement-2;
            }
   statement-3;
}
```



nested if-else

- When if-else structure is placed in another ‘if-else’ structure, it is called “nested if-else” structure. It is used for multiple selection.
- Syntax:

```
If(condition-1)
      statement-1;
else if(condition-2)
      statements-2;
else if (condition-3)
       statement-3;
-----------------------
---------------
else
Statement-n;
```



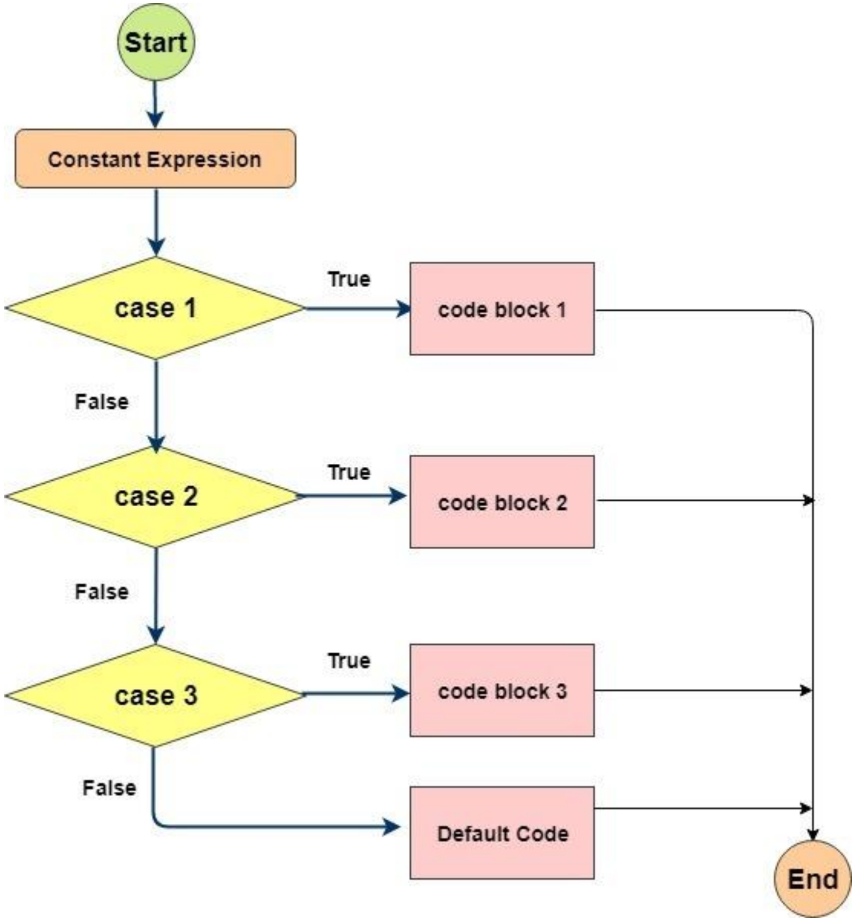
Switch statements

- The “switch ” statement

The switch statement is used as a substitute of “nested if-else” statements. It is used when multiple choices are given and one choice is to be selected . The “nested if-else” structure becomes complicated in multiple choices. Only one condition is given in the “switch statement” and multiple choices are given inside the main body as cases.

Syntax :

```
switch(expression)
{
case const-1:
   statements ;
   break;
case const-2:
  statement ;
  break;
……till n;
Default:
 statements;
}
```



The break Statement

- The break statement is used to exit from the body of the switch structure.
- In the switch statement, the break statement is normally used at the end of statements in each case. It exits the control from the body of the switch structure. If it is not used then the statements of other cases that come after the matching case will also be executed.

Difference between nested if-else and switch statements

Nested if-else statements

1. Complicated for multiple selection.
2. It uses independent expression for each case.
3. The test condition can be given in a specified range of values. If the given condition matches then the statements under it will be executed.

Switch statement

1. Easy to understand for multiple selections.
2. It uses a single expression for all cases, but each case must have a constant value of integer type or character type.
3. Only a single expression in given is this statement that returns a single value.
4. The test condition cant be given in a specified range , which is its major drawback.