Lecture 4
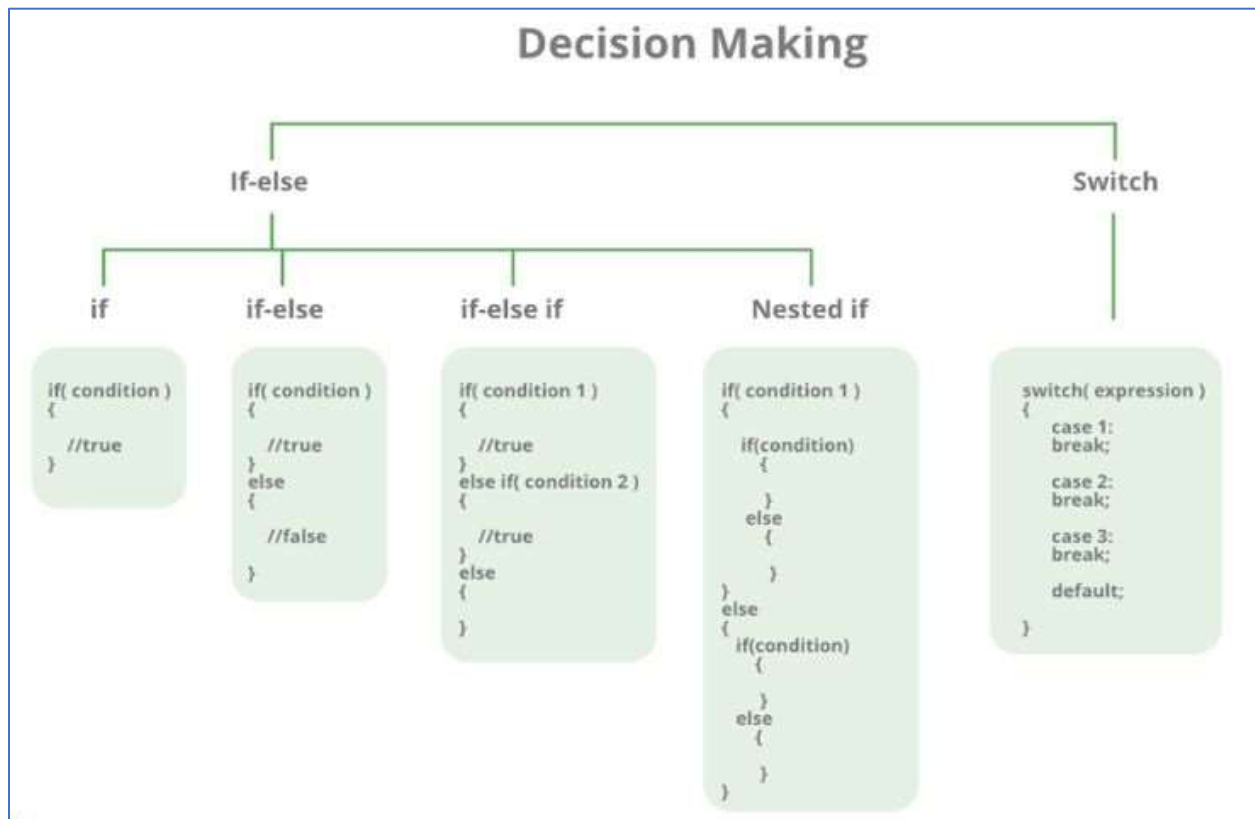
CONDITIONAL SELECTION STATEMENTS
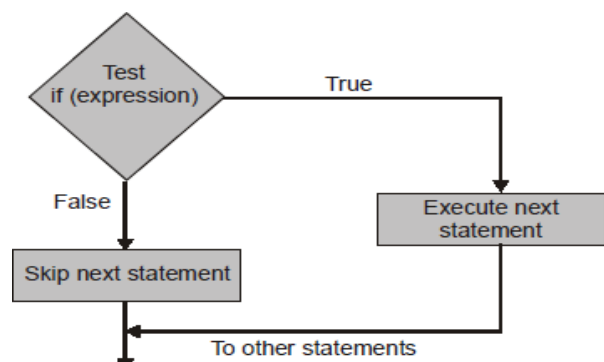
C++ programming language provides three conditional selection statements. They are:

1. **if statement.**

2. **if .... else statement.**

3. **switch statement.**

1. if STATEMENT:

The **if** statement allows to execute an instruction or block of instructions only if thespecified condition is true.

Syntax:

if (conditional expression) statement;

If there are more than one statement to be executed, they are enclosed in curly braces { }:

if (conditional expression)
{
    Block_of_statements;
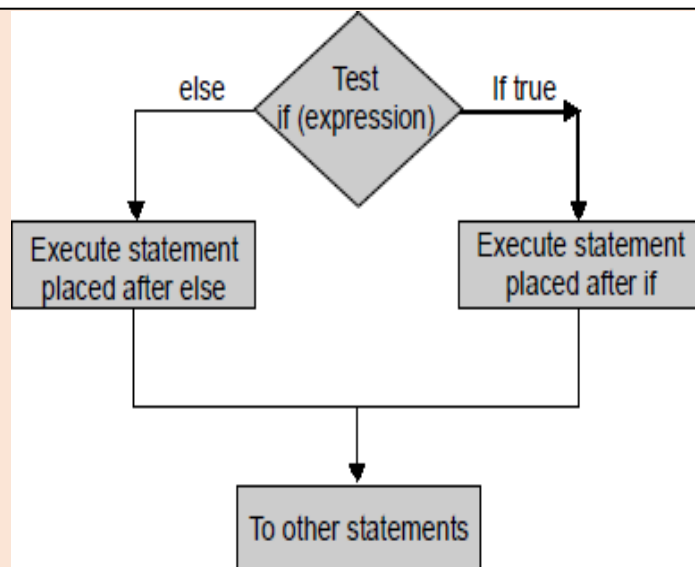**}**

| ```
if (x %2==0)
{
    cout << "x is even";

}
``` | ```
if (x %2==0)
{
    cout << "x is even";
    cout << x;
}
``` |
|---|---|

2. <u>if .... else</u> STATEMENT:

This statement is used when we have two choices, it is written as follows
Syntax:

if (condition)

{

Block1_of_statements;

}

else

{

Block2_of_statements;

Example1:

The following program uses the **if-else** to test for the divisibility of an integer.

```cpp
int main()
{
int n, m;
cout << "Enter two integer numbers:";
cin>>n >>m;
if (n % m==0)
cout<<n<< " is divisible by"<< m ;
else
cout<<n<<" is not divisible by "<< m ;
return (0) ;
}
```

A chain of **if. else** expressions are used if there are more than two choices.

Example2: The following program selects the name of the day of the week out of 7 choices.

```cpp
#include <iostream>
using namespace std;
int main()
{
Int day ;
cout<<"Enter the day number(1-7)" <<endl;
cin>> day ;
if (day == 1)
cout<<"It is Sunday "<<endl;
else if (day == 2)
cout<<"It is Monday "<<endl;
else if (day == 3)
cout<<"It is Tuesday "<<endl;
else if (day == 4)
cout<<"It is Wednesday "<<endl;
else if (day == 5)
cout<<"It is Thursday "<<endl;
else if (day == 6)
cout<<"It is Friday "<<endl;
else if(day==7)
cout<<"It is Saturday "<<endl;
else
cout<<"The number is not in range.";
}
```

### 3.   The switch statement (Multiple Choice Statement)

When multiple selections are required, we may use a **switch** statement which

Syntax is illustrated below:

```
        switch (expression or variable)
        { case value1 : statement1; break;
             case value2 :
             statement2; break;
             .
             .
             .
             case value n :
             statement n; break;
             default : statement; }
```

During execution of the program, the expression is evaluated and compared with the values mentioned in different cases of switch expression. If the value matches a value of a particular case, the statements in that case are executed. If no case-value matches with the value of the expression the program goes to the last statement which is a **default statement** as shown in figure below:
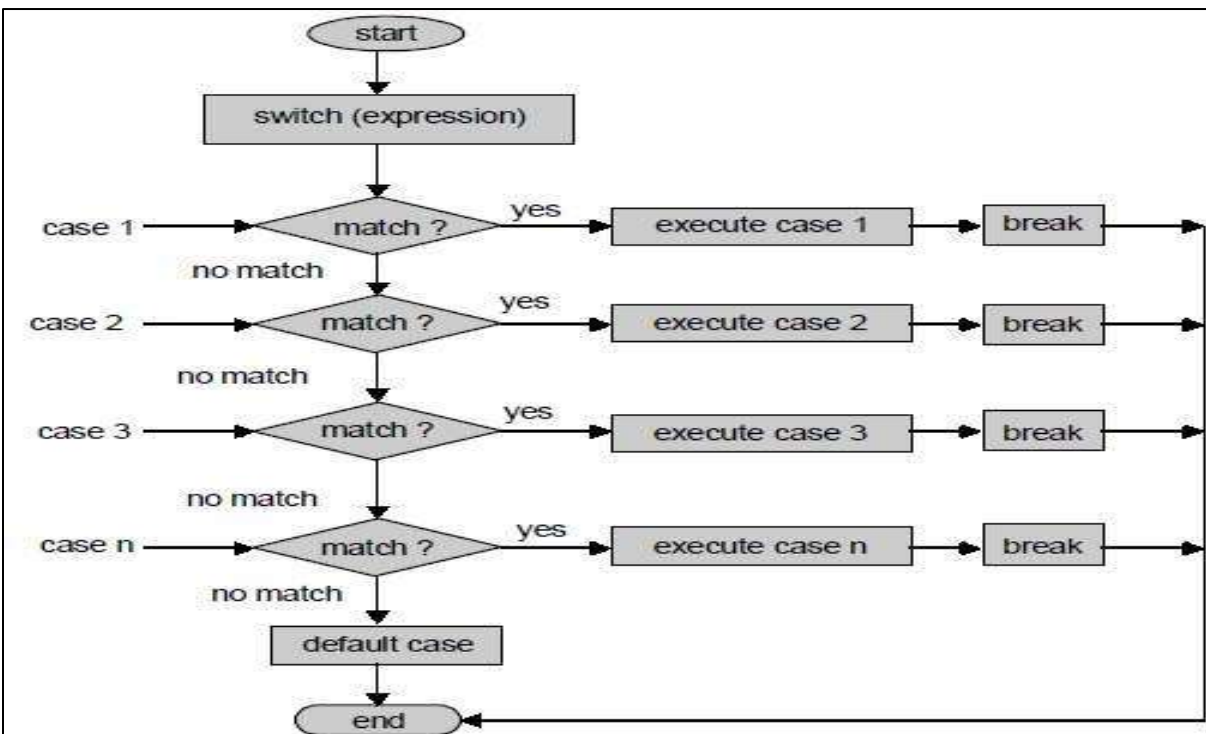
Note The word **break** means an exit from **switch** statement.

Example: The following program illustrates the **switch** statement.

```cpp
#include<iostream>
using namespace std;
 int main()
 {
 int day;
 cout<<" Enter the weekday (1-7)";
 cin>> day;
 switch (day)
 {
 case 1: cout<<"The day is Sunday" ; break;
 case 2: cout<<"The day is Monday" ; break ;
 case 3: cout<<"The day is Tuesday" ; break;
 case 4: cout<<"The day is Wednesday" ; break;
 case 5: cout<<"The day is Thursday" ; break;
 case 6: cout<<"The day is Friday" ; break;
 case 7: cout<<"The day is Saturday" ; break;
 default: cout<<"The number is not in range.";
 }
 }
```
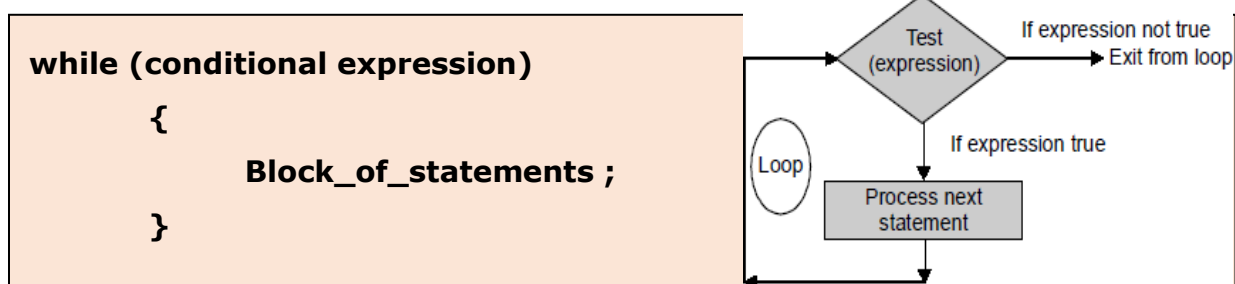
LOOP AND OTHER CONTROL STATEMENTS

In C++ programming language, there are three loop statements, they are:

1. **The while statement (loop).**
2. **The do…while statement (loop).**
3. **The for statement (loop).**

1. THE <u>while</u> STATEMENT.

The **while** statement or loop Syntax:


```
while (conditional expression)
    {
        Block_of_statements ;
    }
```

This means that as long as the condition is true, **Block_of_statements** will be executed.

Example: **Use while loop to find the sum of numbers from 0 to 10.**
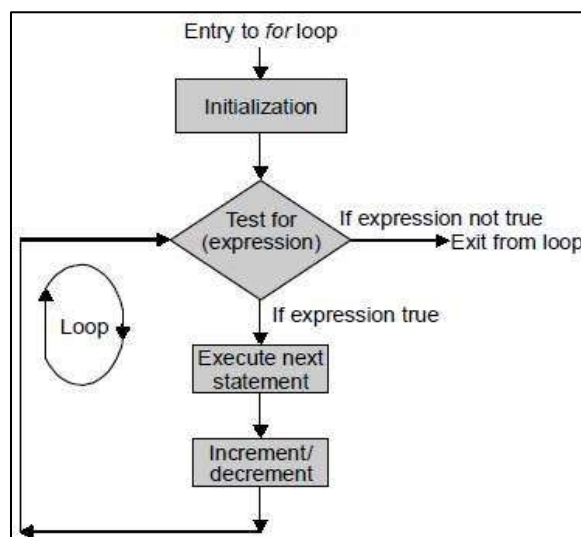
```
#include <iostream>
using namespace std;
int main ()
{
   int n = 10, i= 0, sum = 0;
    while (i <= n)
      sum = sum + i++;
cout<< "Sum = "<< sum;
}
```

1. THE for LOOP

The **for-loop** Syntax is:

for (initial value; condition; increment /decrement)

      {

            Block_of_statements ;

      }

The **for** loop is controlled by three expressions: an initialization, a condition, and update(increment/decrement).

Example: Write c++ program to add the numbers between 1 and 10.

```
#include <iostream>
using namespace std;
        int main()
        {
         int n=5, sum=0,i;
        for(i=1;i<=n;++i)
         {
        sum = sum+i;
        cout <<i;
         }
        cout <<"\n The sum of first 10 natural numbers:" << sum ;
        }
}
```