Control Structure and Loop Statements

A C/C++ program executes in sequential order that is the way the instructions are written. There are situations when we have to skip certain code in the program and do something else. The control structure in C++ are used to transfer control of the execution sequence to another point in the program. There are various types of control structures:

- if-else
- switch
- break – unconditional and conditional
- goto
- continue

LOOP statement

When a statement or group of statements is required to be executed repeatedly, then loop statements are of great help to us. The loop statement in C++ are given below:

- For-loop
- While loop
- Do while loop

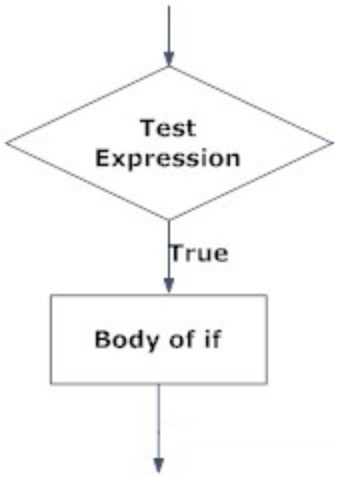
Control Statements

1. If statement

If statement is used when conditional execution is required. The statement test for a condition, if the condition matches (i.e evaluates to true) then the statements are executes else (if condition evaluates to false) the statements following the if are skipped. There are various form of writing the if statement, and these are:

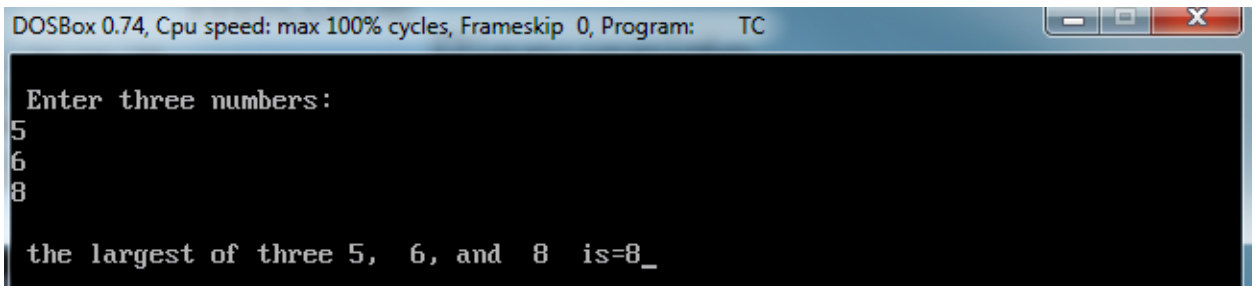
i. if statement
ii. if-else statement
iii. if-else-if statement
iv. nested if statements

i. if statement



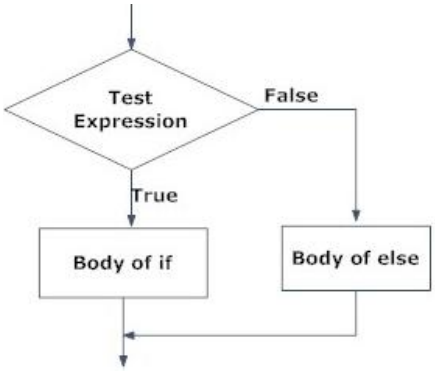

```
#include<iostream.h>
#include<conio.h>
void main()
{
        int a,b,c, largest;
        cout << "enter three numbers\n";
```

```
        cin>>a>>b>>c;
        largest = a;
        if(b>largest)
        {
                largest = b;
        }
        If (c > largest)
        {
                largest = c;
        }
        cout<<  "largest of "<<a<<",  "<<b<", and  "<<c <<"is = "<<largest;
        getch();
}
```



ii. The if-else statement

The if-else statement



WAP to display the result 1st, 2nd, 3rd division or fail if a student scores >60%, >50%, >40%, <40%

```
#include<iostream.h>
#include<conio.h>
void main()
{
```

```
        int percent, division;
        cout <<”enter your percentage of marks”<<endl;
        cin>>percent;
        if(percent > 60)
                cout<<”You scored 1st division\n”;
        else
        if(percent >50)
                cout<<”You scored 2nd division\n”;
        if(percent>40)
                cout<<”You scored 3rd division\n”;
        else
                cout<<”You Failed\n”;
        getch();
}
```

SWITCH CASE

//WAP to develop a calculator to perform Add, Sub, Mul, and Division Operation using switch case and do-while statement

```
#include<iostream.h>
#include<conio.h>
void main()
{
        clrscr();
        int a,b,c;
        char choice,ans;
        do
        {
                cout<<"enter A for Addition\n";
                cout<<"enter s for Sub"<<endl;
                cout<< "enter M for Mul"<<endl;
                cout<<"enter D for Div"<<endl;
                cout <<"What u want to do ?";
                cin>>choice;

        switch(choice)
                {
                case 'A':cout<<"enter two numbers";
                                cin>>a>>b;
                                c=a+b;
                                cout<<"the sum is"<<c;
                                break;
                case 'S':  cout<<"enter two numbers";
                                cin>>a>>b;
```

```
                    c=a-b;
                    cout<<"the diff is"<<c;
                    break;
            case 'M'   : cout<<"enter two numbers";
                    cin>>a>>b;
                    c=a*b;
                    cout<<"the mul is"<<c;
                    break;
            case 'D':    cout<<"enter two numbers";
                    cin>>a>>b;
                    c=a/b;
                    cout<<"the div is"<<c;
                    break;
            default:     cout<<"invalid choice"; break;
            }
            cout <<"Want to do another operation (Y or y for yes)"<<endl;
            cin>>ans;
        }while(ans=='y' || ans=='Y');
}
```

LOOP Statements

FOR-LOOP

```
// program to generate factorial of a number

#include<iostream.h>
#include<conio.h>
void main()
{
  int n, fact=1;
  cout <<"enter a number whose factorial is required:"<<endl;
  cin>>n;
  for (int i=1; i<=n;i++)
  {
   fact=fact*i;
  }
  cout<<"factorial of : "<<n <<" is : "<<fact<<endl;
  getch();
}
```

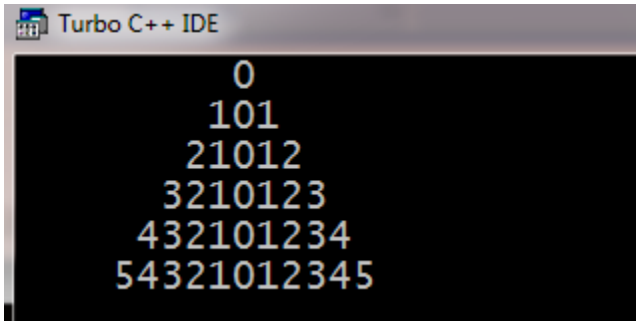
```
Turbo C++ IDE
enter a number whose factorial is required:
5
factorial of : 5 is : 120
```

Nested For LOOP

```
#include<iostream.h>

#include<conio.>
void main()
{
 int i,j;
 clrscr();
 for (i=0 ; i<=5 ;i++)
 {
   for (j=0;j<=i ; j++)
   {
     cout <<j;
   }
 cout<<"\n";
 }
getch();
}
```

```
Turbo C++ IDE
0
01
012
0123
01234
012345
```

```
#include<iostream.h>
#include<iomanip.h>
#include<conio.h>
void main()
{
        int i,j,k,m=10;
        clrscr();
                for (i=0 ; i<=5 ;i++)
                {
                        cout<<setw(m);
                        for (j=i;  j>=0 ; j--)
                        {
                                cout <<j;
                        }

                        for (k=1;k<=i;k++)
                        {
                                cout <<k;
                        }
                m--;
                cout<<"\n";
                }
        getch();
}
```



BREAK STATEMENT

```
while (test expression) {
    statement/s
    if (test expression) {
        break;
    }
    statement/s
}
```

```
do {
    statement/s
    if (test expression) {
        break;
    }
    statement/s
}
while (test expression);
```

```
for (intial expression; test expression; update expression) {
    statement/s
    if (test expression) {
        break;
    }
    statements/
}
```

NOTE: The break statment may also be used inside body of else statement.

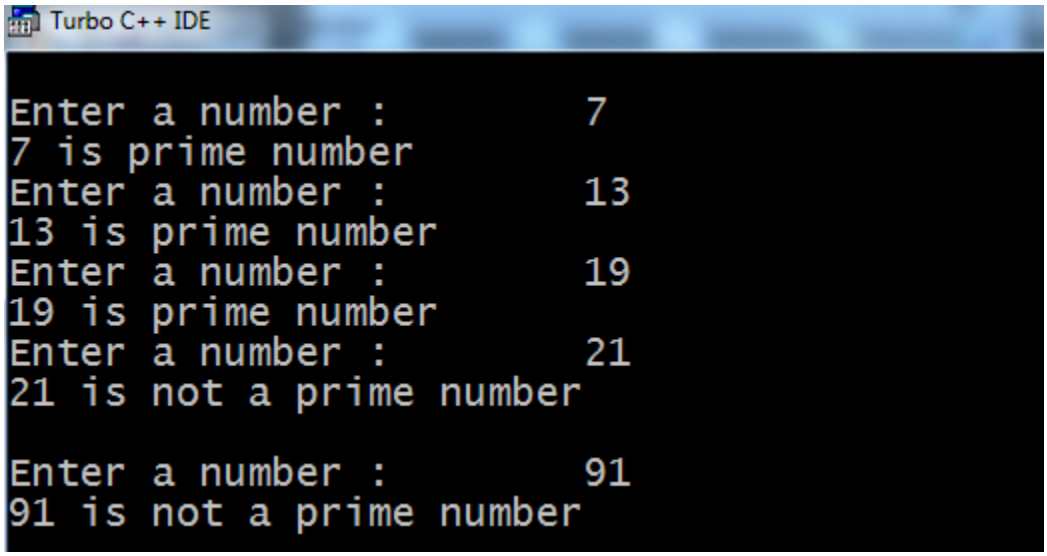
// Program to enter +ve number; Break if –ve number is entered

```
#include<iostream.h>
#include<conio.h>
void main()
{
        int x,i;
        for (i=0; i<=10;i++)
        {
        cout <<"enter a number:\n";
        cin>>x;
        if(x< 0)
        break;
        }
        cout<<"\n You entered a -ve number:";
        getch();
}
```

//Program to find if the number is prime or not

```
#include<iostream.h>
#include<conio.h>
void main()
{
```

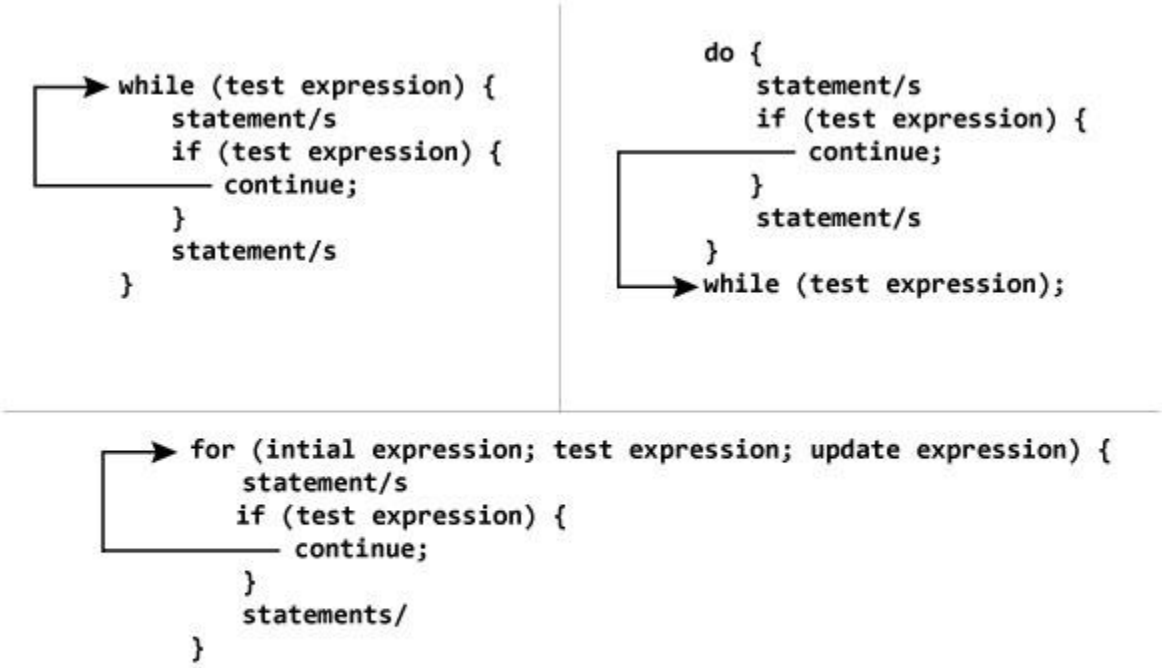
```
        int num, d=2;
        cout <<"\nEnter a number :\t";
        cin>>num;
        while(d <num)
        {
                if(num%d==0)
                {
                        cout<<num<<" is not a prime number\n";
                        break;
                }
        }
        d++;

        }

        if (d==num)
        cout<< num <<" is prime number";

        getch();
}
```



CONTINUE STATEMENT

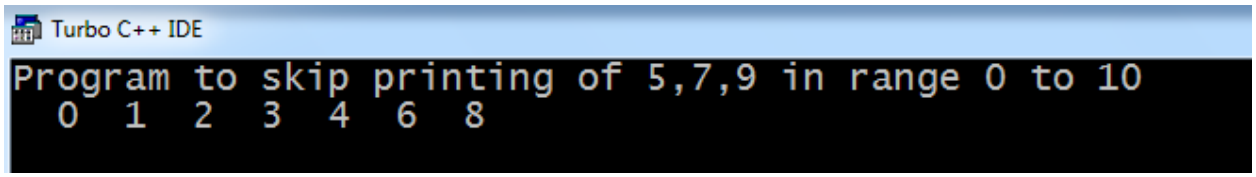
The continue statement causes control of the loop body to go at the beginning of the loop bypassing the remaining statements in the loop body. The continue statement is usually associated with the if statement. On meeting certain condition the program will enter the next iteration.

- In the case of the for loop as soon as after the execution of continue statement, increment/decrement statement of the loop gets executed. After the execution of increment statement, condition will be checked.
- In case of the while loop, continue statement will take control to the condition statement.
- In case of the do-while loop, continue statement will take control to the condition statement specified in the while loop.

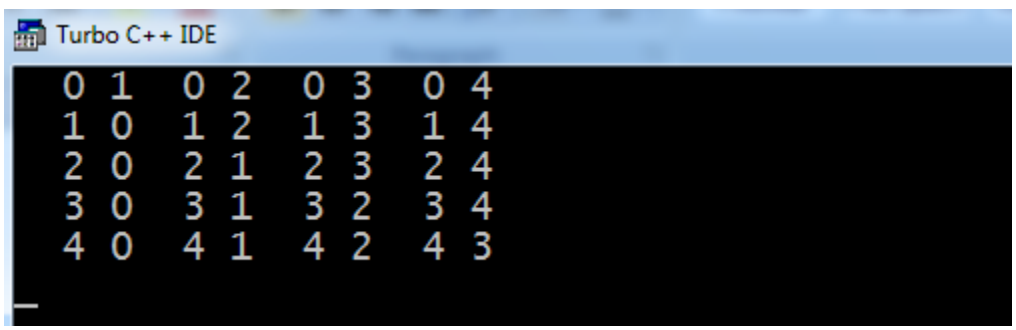


NOTE: The continue statment may also be used inside body of else statement.

```
// C++ Program to demonstrate working of continue statement
#include<iostream.h>
#include<conio.h>
void main()
{
      int x,i;
      cout<<"Program to enter +ve number:\n";
      for (i=0; i<=10;i++)
      {
       if(i ==5 || i==7 || i==9)
            continue;
            cout<<"\n i=:\t"<<i;
       }
   getch()
}
```

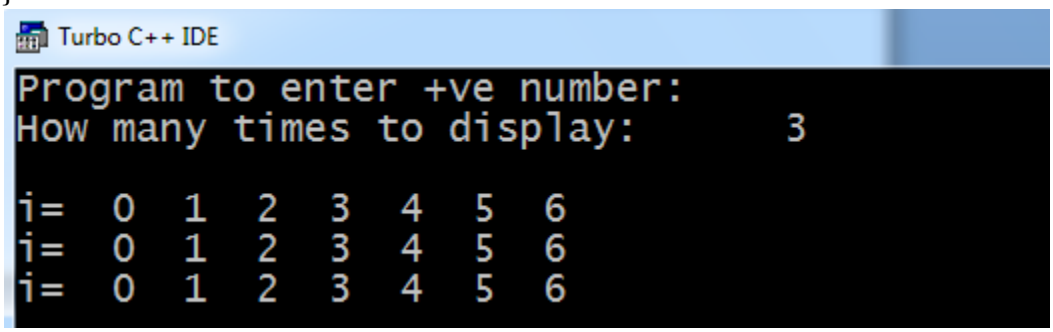



```
//The program skips the next statement in the for loop when i==j and prints
//the remaining
#include<iostream.h>
#include<conio.h>
void main()
{
   int j,i;
   clrscr();
   for(i=0; i<5;i++)
   {
    for(j=0;j<5;j++)
    {
       if (i==j)
       continue;
       cout <<"\n"<<i<<"  "<<j;
      }
    }

   getch();
}
```




```
// C++ Program to demonstrate working of continue statement
#include<iostream.h>
#include<conio.h>
void main()
{
        int x,i;
        cout<<"Program to enter +ve number:\n";
        cout<<"How many times to display:\t";
        cin>>x;
        do{
                for (i=0; i<=10;i++)
                {
```

```
                if (i >6)
                  continue;
                  cout<<"\n i=:\t"<<i;
            }
             x--;
      }while(x>=0);
      getch();
}
```



Exercise: From Chapter-3 Kanitkar (Let us C)

1. WAP to Print all prime numbers between 1 and 100 (use break and continue statement)
2. WAP to fill the entire screen with smily face (ASCII value=1)
3. WAP to add first seven terms of the following series using for loop 1/1!+2/2!+3/3!.....
4. WAP to produce the following outputs:

```
5 4 3 2 1 0
  4 3 2 1 0
    3 2 1 0
      2 1 0
        1 0
          0
```

```
5 4 3 2 1 0 1 2 3 4 5
  4 3 2 1 0 1 2 3 4
    3 2 1 0 1 2 3
      2 1 0 1 2
        1 0 1
          0
```

```
* * * * *
  * * * *
    * * *
      * *
        *
```

5. WAP to generate the table of 2 to 9
6. WAP to display the printable character when a user input any ASCII code from 0 to 255
7. WAP to find the roots of a quadratic equation