C++ PROGRAMMING LANGUAGE

For Loop:

A **for** loop is a repetition control structure that allows you to efficiently write a loop that needs to execute a specific number of times.

Syntax

The syntax of a **for loop** in C++ is:

```
for (init; condition; increment or decrement)
{
      statement(s);
}
```

Here is the flow of control in a for loop:

1- The **init** step is executed first, and only once. This step allows you to declare and initialize any loop control variables. You are not required to put a statement here, as long as a semicolon appears.

2- Next, the **condition** is evaluated. If it is true, the body of the loop is executed. If it is false, the body of the loop does not execute and flow of control jumps to the next statement just after the for loop.

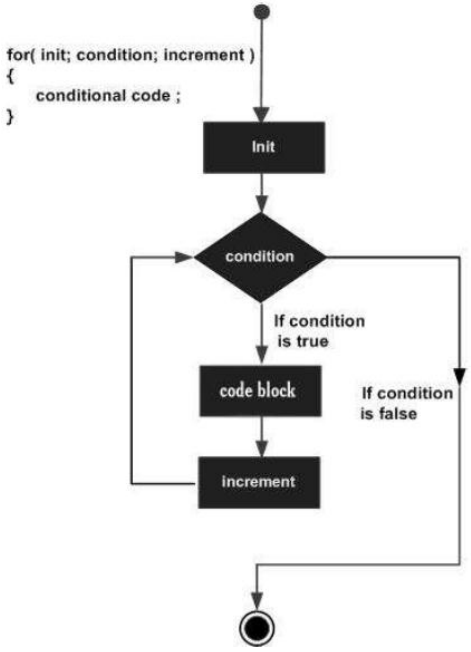
3- After the body of the for loop executes, the flow of control jumps backup to the **increment** statement. This statement allows you to update any loop control variables. This statement can be left blank, as long as a semicolon appears after the condition.

4- The condition is now evaluated again. If it is true, the loop executes and the process repeats itself (body of loop, then increment step, and then again condition). After the condition becomes false, the for loop terminates.

Note: There is no a semicolon (;) after while loop.

C++ PROGRAMMING LANGUAGE

Flow Diagram



Example: Write a program to print the numbers from 1 to 100 (Ascending).

```
#include <iostream>
#include <cstdlib>
using namespace std;
int main()
{
    int i;
    for(i=1;i<=100;i++)
    {
        cout<<i<<endl;
    }
    cout << "GO!" << endl;
    system("pause");
    return 0;
}
```

⟹ `for(int i=1;i<=100;i++)`

Example: Write a program to print the numbers from 100 to 1 (descending).

```
#include <iostream>
#include <cstdlib>
using namespace std;
int main()
{
    for(int i=100;i>=1;i--)
    {
        cout<<i<<endl;
    }
    cout << "GO!" << endl;
    system("pause");
    return 0;
}
```

C++ PROGRAMMING LANGUAGE

Example: Write a program to print the even numbers from 0 to 10.

```
#include <iostream>
#include <cstdlib>
using namespace std;
int main()
{
    for(int i=0;i<=10;i+=2)
    {
       cout<<i<<endl;
    }
    cout << "GO!" << endl;
    system("pause");
    return 0;
}
```

Notice the difference

- Using infinite for loop)

```
#include <iostream>
#include <cstdlib>
using namespace std;
int main()
{
    for(int i=0; ;i+=2)
    {
       cout<<i<<endl;
       if(i>=10)
        break;
    }
    cout << "GO!" << endl;
    system("pause");
    return 0;
}
```

⟹ To stop the loop

Example: Write a program to find the summation of numbers from 1 to 10.

```
#include <iostream>
#include <cstdlib>
using namespace std;
int main()
{
    int sum,i;
    for(sum=0,i=0;i<=10;i++)
    {
       sum=sum+i;
    }
    cout << "sum=" << sum << endl;
    system("pause");
    return 0;
}
```

Note: `for(;condition;)` ⟺ `while(condition)`

C++ PROGRAMMING LANGUAGE

Nested Loops

A loop can be nested inside of another loop. C++ allows at least 256 levels of nesting.

Syntax

The syntax for a **nested for loop** statement in C++ is as follows:

```
for (init; condition; increment)
{
      for (init; condition; increment)
      {
            statement(s);
      }
      statement(s); // you can put more statements.
}
```

The syntax for a **nested while loop** statement in C++ is as follows:

```
while(condition)
{
      while(condition)
      {
            statement(s);
      }
      statement(s); // you can put more statements.
}
```

The syntax for a **nested do...while loop** statement in C++ is as follows:

```
do
{
      statement(s); // you can put more statements.
      do
      {
            statement(s);
      }while(condition);
}while(condition);
```

C++ PROGRAMMING LANGUAGE

Example: Write a program to find the prime number from 2 to 100

```
#include <iostream>
#include <cstdlib>
using namespace std;
int main ()
{
    int i, j;
    for(i=2;i<100;i++)
        {
            for(j=2;j<=(i/2);j++)
                if(i%j==0)
                    break; // if factor found, not prime
            if(j>(i/2))
                cout << i << " is prime\n";
        }
    system("pause");
    return 0;
}
```

Example: Write a program to test a number weather prime or not.

```
#include <iostream>
#include <cstdlib>
using namespace std;
int main ()
{
    int i, n;
    cout << "Enter a number: ";
    cin >> n;
    for(i=2;i<=(n/2);i++)
        if(n%i==0)
        {
            cout<<n<<" : is not prime!"<<endl;
            system("pause");
            exit(1);
        }
    cout<<n<<"is prime!"<<endl;
    system("pause");
    return 0;
}
```

C++ PROGRAMMING LANGUAGE

Jump statements (break, continue, go to)

1- Break statement:

The **break** statement has the following two usages in C++:

- When the **break** statement is encountered inside a loop, the loop is immediately terminated and program control resumes at the next statement following the loop.
- It can be used to terminate a case in the **switch** statement (covered in the next chapter).

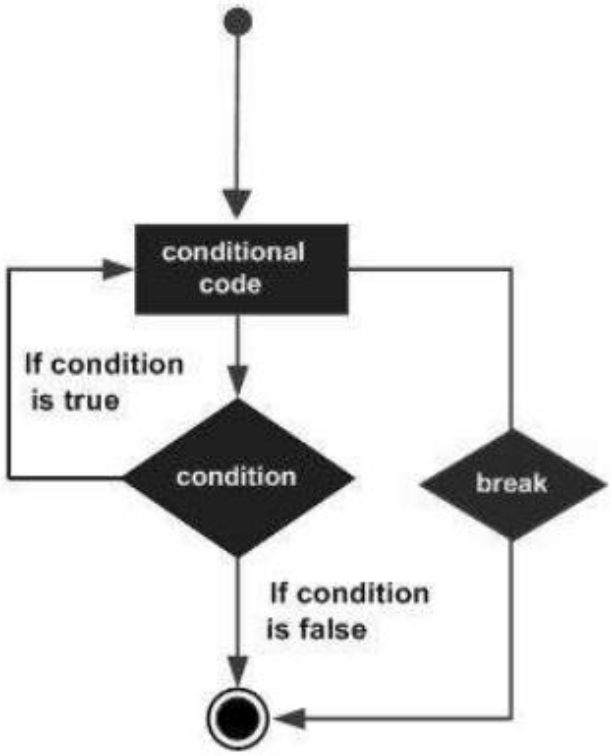
If you are using nested loops (i.e., one loop inside another loop), the break statement will stop the execution of the innermost loop and start executing the next line of code after the block.

Syntax

The syntax for a **break** statement in C++ is as follows:

```
break;
```

Flow Diagram



C++ PROGRAMMING LANGUAGE

Example:

```
#include <iostream>
#include <cstdlib>
using namespace std;
int main ()
{
    int a = 1;
    for(;;)
    {
        if(a > 15)
            break;
        cout << "value of a: " << a << endl;
        a++;
    }
    system("pause");
    return 0;
}
```

Example:

```
#include <iostream>
#include <cstdlib>
using namespace std;
int main ()
{
    int a = 1;
    do
    {
        cout << "value of a: " << a << endl;
        a = a + 1;
        if(a > 15)
        {
           break;
        }
    }while(a<20);
    system("pause");
    return 0;
}
```

C++ PROGRAMMING LANGUAGE

2- continue statement:

The **continue** statement works somewhat like the break statement. Instead of forcing termination, however, continue forces the next iteration of the loop to take place, skipping any code in between.

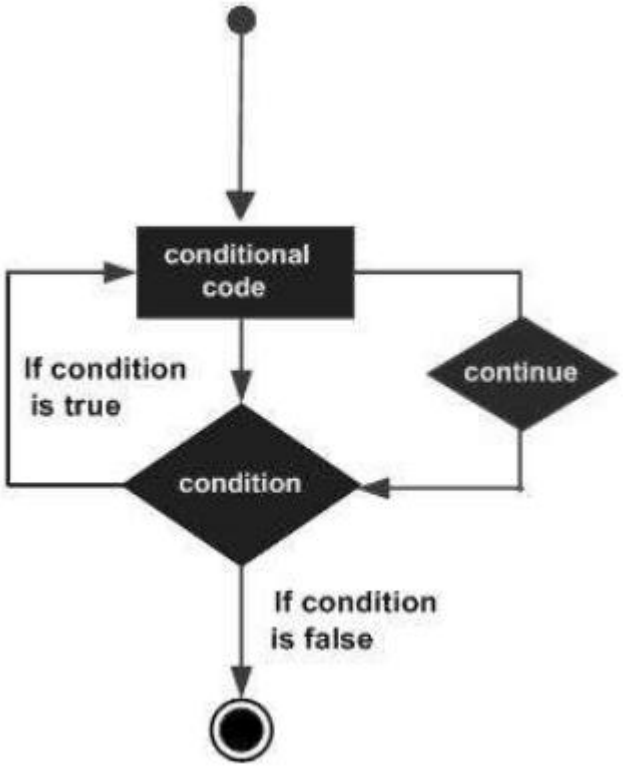
For the **for** loop, continue causes the conditional test and increment portions of the loop to execute. For the **while** and **do...while** loops, program control passes to the conditional tests.

Syntax

The syntax of a **continue** statement in C++ is:

```
continue;
```

Flow Diagram



C++ PROGRAMMING LANGUAGE

Example:

```
#include <iostream>
#include <cstdlib>
using namespace std;
int main ()
{
    int a = 1;
    while(1)
    {
        if(a <= 15)
        {
            cout << "value of a: " << a << endl;
            a++;
        }
        cout<<"GO!"<<endl;
        break;
    }
    system("pause");
    return 0;
}
```

Output:

```
value of a: 1
GO!
Press any key to continue . . .
```

In the example above, we have seen that the output is only one value.
To correct this program, we will use (**continue statement**) as follows:

```
#include <iostream>
#include <cstdlib>
using namespace std;
int main ()
{
    int a = 1;
    while(1)
    {
        if(a <= 15)
        {
            cout << "value of a: " << a << endl;
            a++;
            continue;
        }
        cout<<"GO!"<<endl;
        break;
    }
    system("pause");
    return 0;
}
```

C++ PROGRAMMING LANGUAGE

3- Goto Statement

A **goto** statement provides an unconditional jump from the goto to a labeled statement in the same function.

NOTE: Use of **goto** statement is highly discouraged because it makes difficult to trace the control flow of a program, making the program hard to understand and hard to modify. Any program that uses a goto can be rewritten so that it doesn't need the goto.

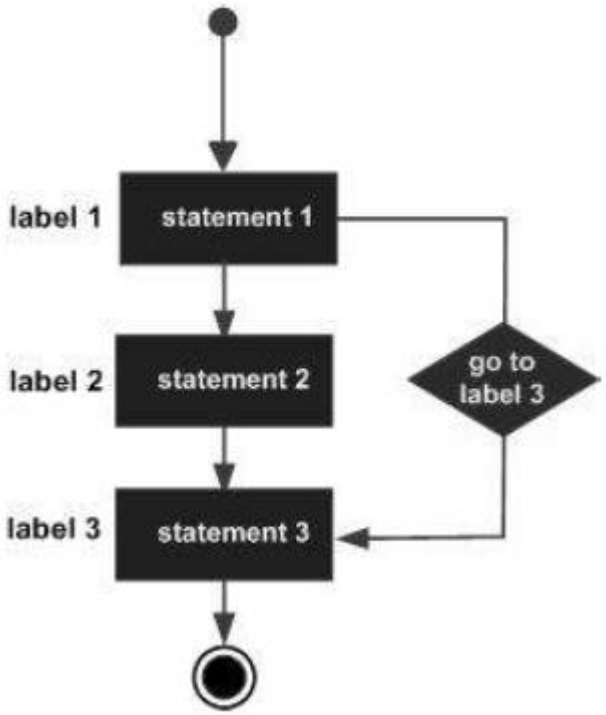
Syntax

The syntax of a **goto** statement in C++ is:

```
goto label;
..
.
label: statement;
```

Where **label** is an identifier that identifies a labeled statement. A labeled statement is any statement that is preceded by an identifier followed by a colon (:).

Flow Diagram



C++ PROGRAMMING LANGUAGE

Example:

```
#include <iostream>
#include <cstdlib>
using namespace std;
int main ()
{
    int a = 1;
    Loop:
    while(1)
    {
        if(a <= 15)
        {
            cout << "value of a: " << a << endl;
            a++;
            goto Loop;
        }
        cout<<"GO!"<<endl;
        break;
    }
    system("pause");
    return 0;
}
```

C++ PROGRAMMING LANGUAGE

Examples:

1- Write a program to calculate the factorial of a number:

n!=n*(n-1) *(n-2) *(n-3)*…..*1

```
#include <iostream>
#include <cstdlib>
using namespace std;
int main ()
{
    int n,k,fact;
    cout << "\nEnter an integer number :";
    cin >> n;
    if(n==0)
        cout << "\nThe factorial of 0 is 1";
    else if(n>0)
    {
        for(fact=1,k=n;k>=1;k--)
            fact*=k;
        cout << "\nThe factorial of" << n << "is: "<< fact
<<endl;
    }
    else
        cout << "\nError Data Entry";
    system("pause");
    return 0;
}
```

(H.W)

2- Rewrite the previous program by using another method.

3- Write a program to print 5 stars as follows:

a- ***** **(H.W) b-**

```
*
*
*
*
*
```

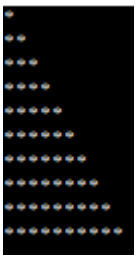
```
#include <iostream>
#include <cstdlib>
using namespace std;
int main ()
{
    int i;
    for(i=1;i<=5;i++)
        cout << "*";
    cout<<endl;
    system("pause");
    return 0;
}
```

C++ PROGRAMMING LANGUAGE

4- Write a program to print the following form:

```
#include <iostream>
#include <cstdlib>
using namespace std;
int main ()
{
    int a,i;
    for(a=1;a<=10;a++)
    {
        for(i=1;i<=5;i++)
            cout << "*";
        cout << endl;
    }
    system("pause");
    return 0;
}
```

5- Write a program to print the following form:



```
#include <iostream>
#include <cstdlib>
using namespace std;
int main ()
{
    int a,i;
    for(a=1;a<=10;a++)
    {
        for(i=1;i<=a;i++)
            cout << "*";
        cout << endl;
    }
    system("pause");
    return 0;
}
```

6- Write a program to print the following form:

```
#include <iostream>
#include <cstdlib>
using namespace std;
int main ()
{
    int a,i;
    for(a=1;a<=10;a++)
    {
        for(i=1;i<=2*a;i++)
            cout << "*";
        cout << endl;
    }
    system("pause");
    return 0;}
```

C++ PROGRAMMING LANGUAGE

(H.W)

7- Write a program to print the following form:

8- Write a program to print the following form:

```
#include <iostream>
#include <cstdlib>
using namespace std;
int main ()
{
    int a,i;
    for(a=1;a<=5;a++)
    {
        for(i=1;i<=6-a;i++)
            cout << "*";
        cout << endl;
    }
    system("pause");
    return 0;
}
```

9- Write a program to print the following form:

```
#include <iostream>
#include <cstdlib>
using namespace std;
int main ()
{
    int a,i,j;
    for(a=1;a<=5;a++)
    {
        for(j=1;j<=5-a;j++)
            cout << " ";
        for(i=1;i<=2*a-1;i++)
            cout << "*";
        cout << '\n';
    }
    system("pause");
    return 0;
}
```

(H.W)10- Write a program to print the following form: