

# C++ if-else

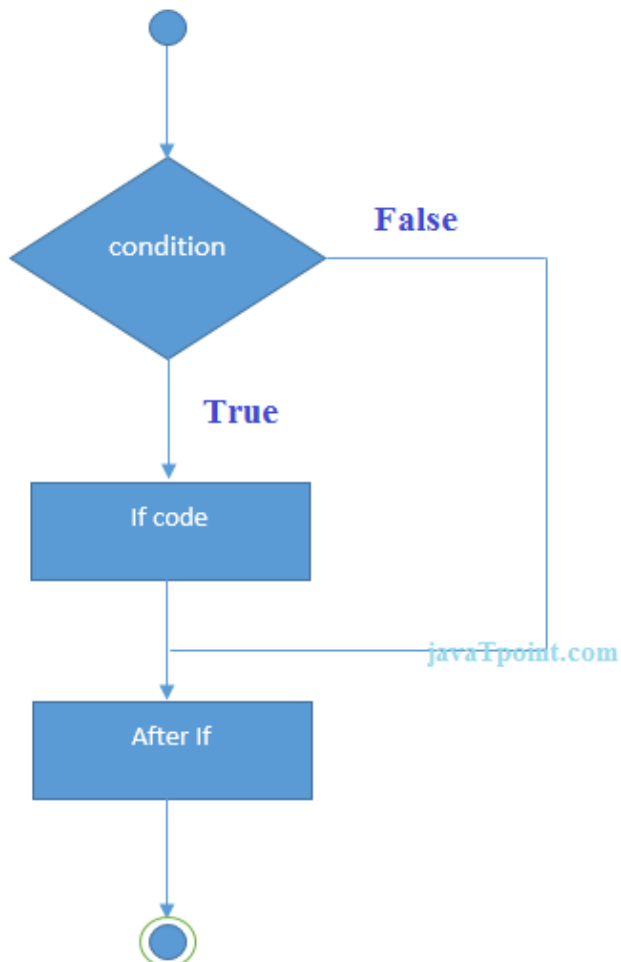
In C++ programming, if statement is used to test the condition. There are various types of if statements in C++.

- if statement
- if-else statement
- nested if statement
- if-else-if ladder

## C++ IF Statement

The C++ if statement tests the condition. It is executed if condition is true.

```
if(condition){  
  //code to be executed  
}
```



## C++ If Example

```
#include <iostream>

using namespace std;

int main () {
    int num = 10;
    if (num % 2 == 0)
    {
        cout<<"It is even number";
    }
    return 0;
}
```

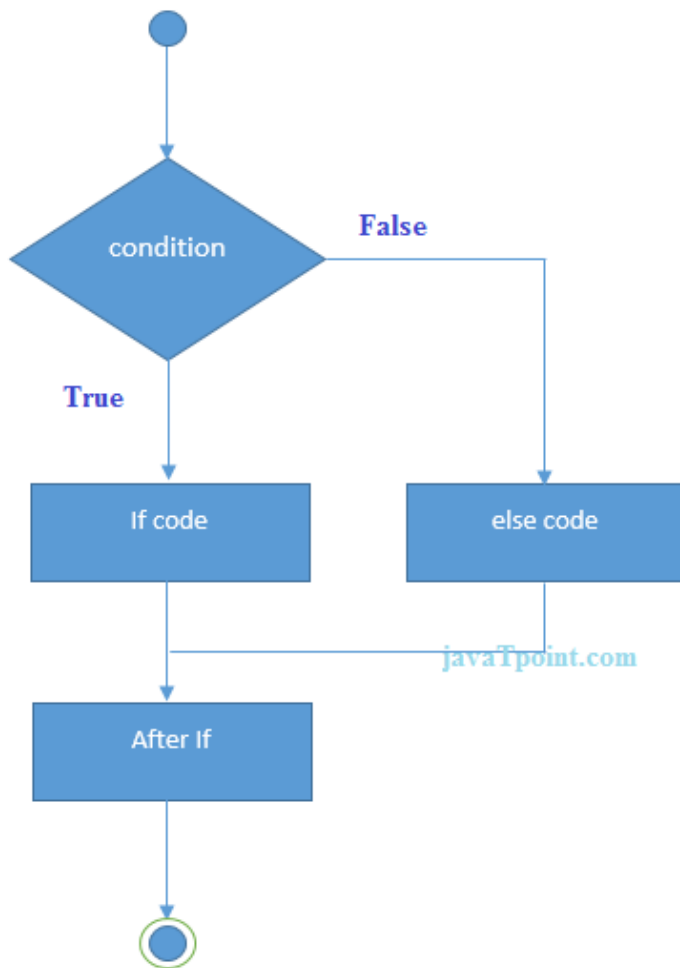
Output:/p>

```
It is even number
```

## C++ IF-else Statement

The C++ if-else statement also tests the condition. It executes if block if condition is true otherwise else block is executed.

```
if(condition){
    //code if condition is true
}else{
    //code if condition is false
}
```



AD



VIDEO STREAMING FOR THE AWAKENED MIND

GET STARTED NOW

## C++ If-else Example

```
#include <iostream>
using namespace std;
int main () {
    int num = 11;
    if (num % 2 == 0)
    {
        cout<<"It is even number";
    }
    else
    {
        cout<<"It is odd number";
    }
    return 0;
}
```

```
}
```

**Output:**

```
It is odd number
```

## C++ If-else Example: with input from user

```
#include <iostream>
using namespace std;
int main () {
    int num;
    cout<<"Enter a Number: ";
    cin>>num;
    if (num % 2 == 0)
    {
        cout<<"It is even number"<<endl;
    }
    else
    {
        cout<<"It is odd number"<<endl;
    }
    return 0;
}
```

**Output:**

```
Enter a number:11
It is odd number
```

**Output:**

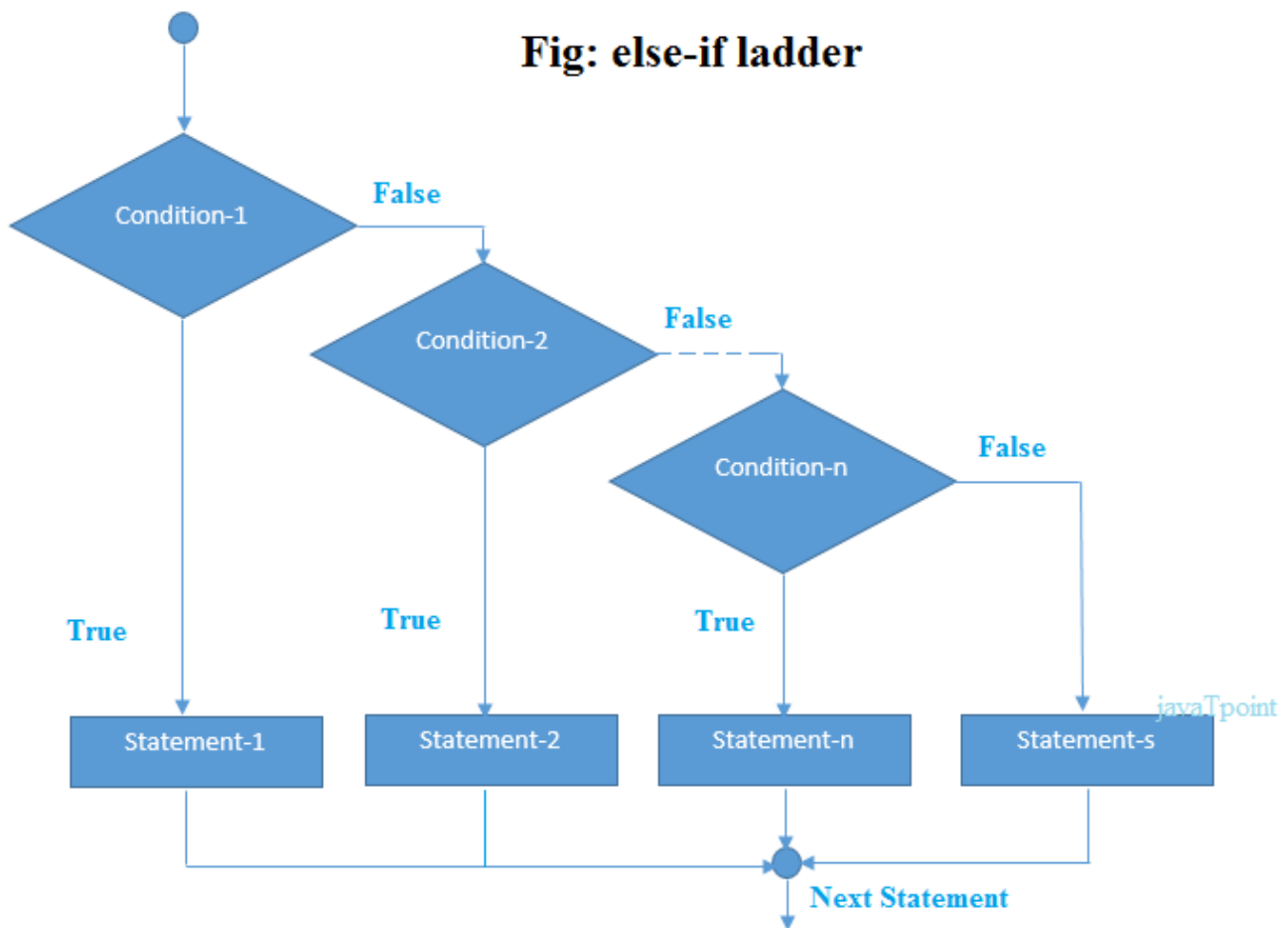
```
Enter a number:12
It is even number
```

## C++ IF-else-if ladder Statement

The C++ if-else-if ladder statement executes one condition from multiple statements.

```
if(condition1){  
    //code to be executed if condition1 is true  
}  
else if(condition2){  
    //code to be executed if condition2 is true  
}  
else if(condition3){  
    //code to be executed if condition3 is true  
}  
...  
else{  
    //code to be executed if all the conditions are false  
}
```

**Fig: else-if ladder**



## C++ If else-if Example

```
#include <iostream>

using namespace std;

int main () {
    int num;
    cout<<"Enter a number to check grade:";
    cin>>num;

    if (num <0 || num >100)
    {
        cout<<"wrong number";
    }

    else if(num >= 0 && num < 50){
        cout<<"Fail";
    }

    else if (num >= 50 && num < 60)
    {
        cout<<"D Grade";
    }

    else if (num >= 60 && num < 70)
    {
        cout<<"C Grade";
    }

    else if (num >= 70 && num < 80)
    {
        cout<<"B Grade";
    }

    else if (num >= 80 && num < 90)
    {
        cout<<"A Grade";
    }

    else if (num >= 90 && num <= 100)
    {
        cout<<"A+ Grade";
    }

}
```

**Output:**

```
Enter a number to check grade:66  
C Grade
```

**Output:**

```
Enter a number to check grade:-2  
wrong number
```

[< Prev](#)[Next >](#)

AD

 [For Videos Join Our Youtube Channel: Join Now](#)

## Feedback

- Send your Feedback to [feedback@javatpoint.com](mailto:feedback@javatpoint.com)

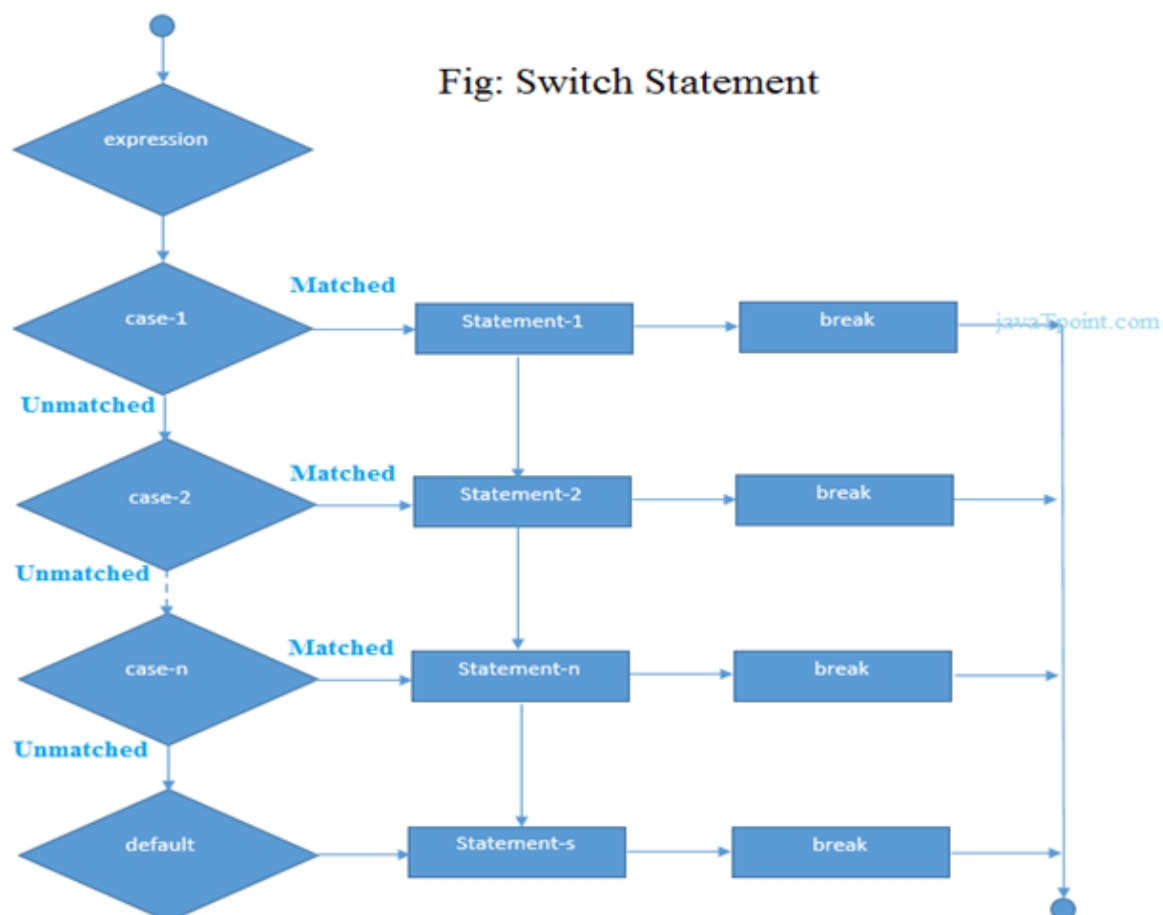
## Help Others, Please Share



# C++ switch

The C++ switch statement executes one statement from multiple conditions. It is like if-else-if ladder statement in C++.

```
switch(expression){  
  case value1:  
    //code to be executed;  
    break;  
  case value2:  
    //code to be executed;  
    break;  
  .....  
  default:  
    //code to be executed if all cases are not matched;  
    break;  
}
```





## C++ Switch Example

```
#include <iostream>
using namespace std;
int main () {
    int num;
    cout<<"Enter a number to check grade:";
    cin>>num;
    switch (num)
    {
        case 10: cout<<"It is 10"; break;
        case 20: cout<<"It is 20"; break;
        case 30: cout<<"It is 30"; break;
        default: cout<<"Not 10, 20 or 30"; break;
    }
}
```

Output:

```
Enter a number:
10
It is 10
```

Output:

```
Enter a number:
55
Not 10, 20 or 30
```

← Prev

Next →

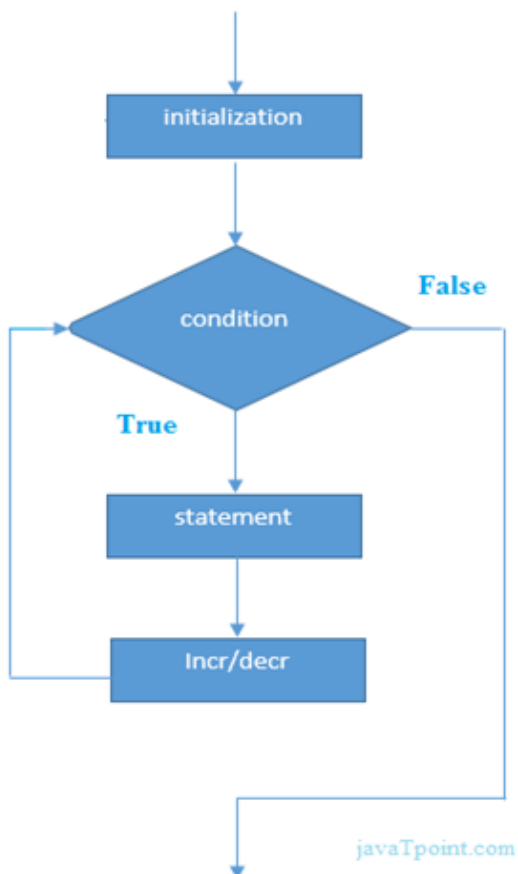
# C++ For Loop

The C++ for loop is used to iterate a part of the program several times. If the number of iteration is fixed, it is recommended to use for loop than while or do-while loops.

The C++ for loop is same as C/C#. We can initialize variable, check condition and increment/decrement value.

```
for(initialization; condition; incr/decr){  
    //code to be executed  
}
```

## Flowchart:



## C++ For Loop Example

```
#include <iostream>  
using namespace std;  
int main() {
```

```
for(int i=1;i<=10;i++){  
    cout<<i <<"\n";  
}  
}
```

Output:

```
1  
2  
3  
4  
5  
6  
7  
8  
9  
10
```

## C++ Nested For Loop

In C++, we can use for loop inside another for loop, it is known as nested for loop. The inner loop is executed fully when outer loop is executed one time. So if outer loop and inner loop are executed 4 times, inner loop will be executed 4 times for each outer loop i.e. total 16 times.

## C++ Nested For Loop Example

Let's see a simple example of nested for loop in C++.

```
#include <iostream>  
  
using namespace std;  
  
int main () {  
    for(int i=1;i<=3;i++){  
        for(int j=1;j<=3;j++){  
            cout<<i<<" "<j<<"\n";  
        }  
    }  
}
```

```
}
```

Output:

```
1 1
1 2
1 3
2 1
2 2
2 3
3 1
3 2
3 3
```

AD

## C++ Infinite For Loop

If we use double semicolon in for loop, it will be executed infinite times. Let's see a simple example of infinite for loop in C++.

```
#include <iostream>
using namespace std;

int main () {
    for (;)
    {
        cout<<"Infinitive For Loop";
    }
}
```

Output:

```
Infinitive For Loop
Infinitive For Loop
```

```
Infinitive For Loop  
Infinitive For Loop  
Infinitive For Loop  
ctrl+c
```

[← Prev](#)[Next →](#)

AD

 [For Videos Join Our Youtube Channel: Join Now](#)


## Feedback


- Send your Feedback to [feedback@javatpoint.com](mailto:feedback@javatpoint.com)

## Help Others, Please Share




## Learn Latest Tutorials

 [Splunk tutorial](#)  
Splunk

 [SPSS tutorial](#)  
SPSS

 [Swagger tutorial](#)  
Swagger

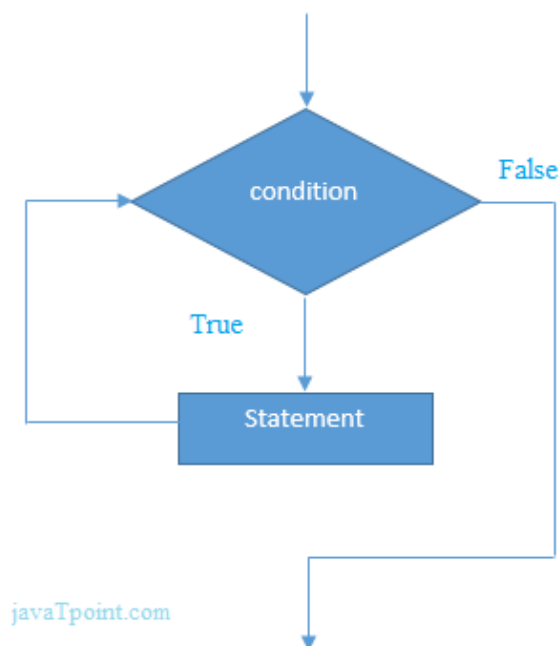
 [T-SQL tutorial](#)  
Transact-SQL

# C++ While loop

In C++, while loop is used to iterate a part of the program several times. If the number of iteration is not fixed, it is recommended to use while loop than for loop.

```
while(condition){  
    //code to be executed  
}
```

## Flowchart:



## C++ While Loop Example

Let's see a simple example of while loop to print table of 1.

```
#include <iostream>  
  
using namespace std;  
  
int main() {  
    int i=1;  
    while(i<=10)  
    {  
        cout<<i <<"\n";  
        i++;  
    }  
}
```

```
}  
}
```

Output:

```
1  
2  
3  
4  
5  
6  
7  
8  
9  
10
```

## C++ Nested While Loop Example

In C++, we can use while loop inside another while loop, it is known as nested while loop. The nested while loop is executed fully when outer loop is executed once.

Let's see a simple example of nested while loop in C++ programming language.

```
#include <iostream>  
using namespace std;  
int main () {  
    int i=1;  
    while(i<=3)  
    {  
        int j = 1;  
        while (j <= 3)  
        {  
            cout<<i<<" "<<j<<"\n";  
            j++;  
        }  
        i++;  
    }  
}
```

```
}
```

Output:

```
1 1
1 2
1 3
2 1
2 2
2 3
3 1
3 2
3 3
```

## C++ Infinitive While Loop Example:

We can also create infinite while loop by passing true as the test condition.

```
#include <iostream>
using namespace std;
int main () {
    while(true)
    {
        cout<<"Infinitive While Loop";
    }
}
```

Output:

```
Infinitive While Loop
Infinitive While Loop
Infinitive While Loop
Infinitive While Loop
Infinitive While Loop
ctrl+c
```



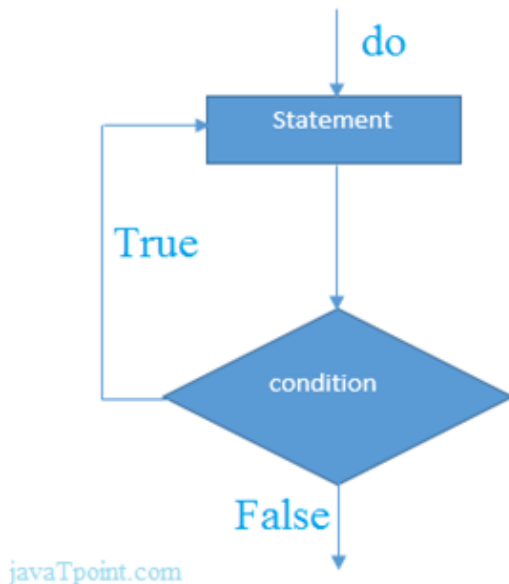
# C++ Do-While Loop

The C++ do-while loop is used to iterate a part of the program several times. If the number of iteration is not fixed and you must have to execute the loop at least once, it is recommended to use do-while loop.

The C++ do-while loop is executed at least once because condition is checked after loop body.

```
do{  
    //code to be executed  
}while(condition);
```

## Flowchart:



## C++ do-while Loop Example

Let's see a simple example of C++ do-while loop to print the table of 1.

```
#include <iostream>  
using namespace std;  
int main() {  
    int i = 1;  
    do{  
        cout<<i<<"\n";  
        i++;  
    }
```

```
    } while (i <= 10);  
}
```

Output:

```
1  
2  
3  
4  
5  
6  
7  
8  
9  
10
```

## C++ Nested do-while Loop

In C++, if you use do-while loop inside another do-while loop, it is known as nested do-while loop. The nested do-while loop is executed fully for each outer do-while loop.

Let's see a simple example of nested do-while loop in C++.

```
#include <iostream>  
using namespace std;  
int main() {  
    int i = 1;  
    do{  
        int j = 1;  
        do{  
            cout<<i<<"\n";  
            j++;  
        } while (j <= 3);  
        i++;  
    } while (i <= 3);  
}
```

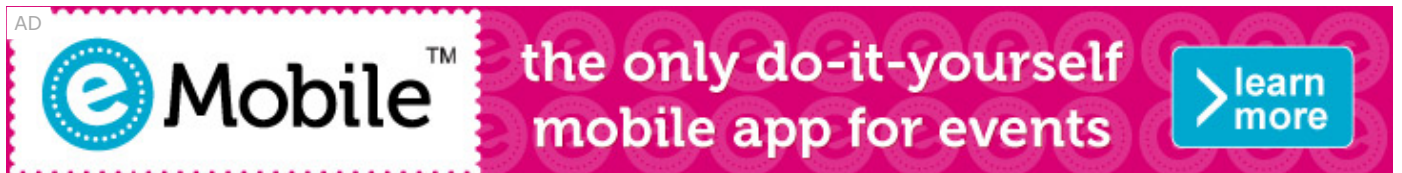
Output:

```
1 1
1 2
1 3
2 1
2 2
2 3
3 1
3 2
3 3
```

## C++ Infinitive do-while Loop

In C++, if you pass **true** in the do-while loop, it will be infinitive do-while loop.

```
do{
    //code to be executed
}while(true);
```



## C++ Infinitive do-while Loop Example

```
#include <iostream>
using namespace std;
int main() {
    do{
        cout<<"Infinitive do-while Loop";
    } while(true);
}
```

Output:

```
Infinitive do-while Loop
Infinitive do-while Loop
```

```
Infinitive do-while Loop
Infinitive do-while Loop
Infinitive do-while Loop
ctrl+c
```

[< Prev](#)[Next >](#)

AD

 [For Videos Join Our Youtube Channel: Join Now](#)


## Feedback


- Send your Feedback to [feedback@javatpoint.com](mailto:feedback@javatpoint.com)

## Help Others, Please Share




## Learn Latest Tutorials

 [Splunk tutorial](#)  
Splunk

 [SPSS tutorial](#)  
SPSS

 [Swagger tutorial](#)  
Swagger

 [T-SQL tutorial](#)  
Transact-SQL

# C++ Break Statement

The C++ break is used to break loop or switch statement. It breaks the current flow of the program at the given condition. In case of inner loop, it breaks only inner loop.

jump-statement;

**break;**

## Flowchart:

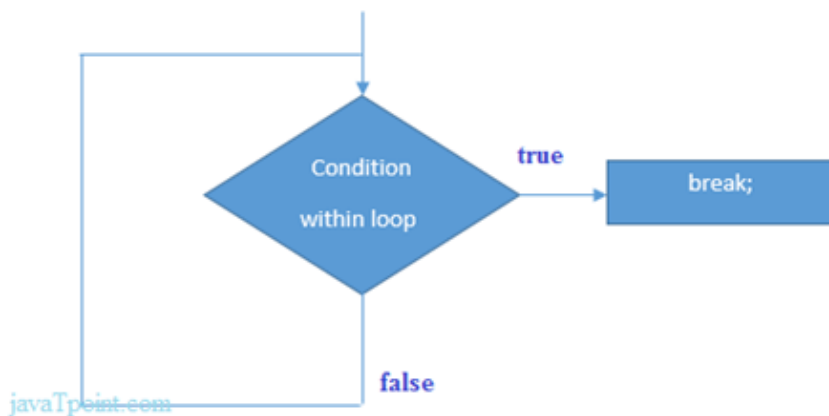


Figure: Flowchart of break statement

## C++ Break Statement Example

Let's see a simple example of C++ break statement which is used inside the loop.

```
#include <iostream>
using namespace std;
int main() {
    for (int i = 1; i <= 10; i++)
    {
        if (i == 5)
        {
            break;
        }
        cout<<i<<"\n";
    }
}
```

```
}
```

Output:

```
1
2
3
4
```

## C++ Break Statement with Inner Loop

The C++ break statement breaks inner loop only if you use break statement inside the inner loop.

Let's see the example code:

```
#include <iostream>
using namespace std;
int main()
{
    for(int i=1;i<=3;i++){
        for(int j=1;j<=3;j++){
            if(i==2&&j==2){
                break;
            }
            cout<<i<<" "<<j<<"\n";
        }
    }
}
```

Output:

```
1 1
1 2
1 3
2 1
3 1
```

```
3 2
```

```
3 3
```

[< Prev](#)[Next >](#)

AD

 [For Videos Join Our Youtube Channel: Join Now](#)


## Feedback


- Send your Feedback to [feedback@javatpoint.com](mailto:feedback@javatpoint.com)

## Help Others, Please Share





## Learn Latest Tutorials


 [Splunk tutorial](#)  
Splunk

 [SPSS tutorial](#)  
SPSS


 [Swagger tutorial](#)  
Swagger

 [T-SQL tutorial](#)  
Transact-SQL

 [Tumblr tutorial](#)  
Tumblr

 [React tutorial](#)  
ReactJS

 [Regex tutorial](#)  
Regex

 [Reinforcement learning tutorial](#)

# C++ Continue Statement

The C++ continue statement is used to continue loop. It continues the current flow of the program and skips the remaining code at specified condition. In case of inner loop, it continues only inner loop.

```
jump-statement;  
continue;
```

## C++ Continue Statement Example

```
#include <iostream>  
using namespace std;  
int main()  
{  
    for(int i=1;i<=10;i++){  
        if(i==5){  
            continue;  
        }  
        cout<<i<<"\n";  
    }  
}
```

Output:

```
1  
2  
3  
4  
6  
7  
8  
9  
10
```



## C++ Continue Statement with Inner Loop

C++ Continue Statement continues inner loop only if you use continue statement inside the inner loop.

```
#include <iostream>
using namespace std;
int main()
{
    for(int i=1;i<=3;i++){
        for(int j=1;j<=3;j++){
            if(i==2&&j==2){
                continue;
            }
            cout<<i<<" "<<j<<"\n";
        }
    }
}
```

Output:

```
1 1
1 2
1 3
2 1
2 3
3 1
3 2
3 3
```

[< Prev](#)[Next >](#)

# C++ Goto Statement

The C++ goto statement is also known as jump statement. It is used to transfer control to the other part of the program. It unconditionally jumps to the specified label.

It can be used to transfer control from deeply nested loop or switch case label.

## C++ Goto Statement Example

Let's see the simple example of goto statement in C++.

```
#include <iostream>
using namespace std;
int main()
{
ineligible:
    cout<<"You are not eligible to vote!\n";
    cout<<"Enter your age:\n";
    int age;
    cin>>age;
    if (age < 18){
        goto ineligible;
    }
    else
    {
        cout<<"You are eligible to vote!";
    }
}
```

Output:

```
You are not eligible to vote!
Enter your age:
16
You are not eligible to vote!
Enter your age:
7
```

```
You are not eligible to vote!  
Enter your age:  
22  
You are eligible to vote!
```

[< Prev](#)[Next >](#)

AD

 [For Videos Join Our Youtube Channel: Join Now](#)


## Feedback


- Send your Feedback to [feedback@javatpoint.com](mailto:feedback@javatpoint.com)

## Help Others, Please Share




## Learn Latest Tutorials

 [Splunk tutorial](#)  
Splunk

 [SPSS tutorial](#)  
SPSS

 [Swagger tutorial](#)  
Swagger

 [T-SQL tutorial](#)  
Transact-SQL

# C++ Comments

The C++ comments are statements that are not executed by the compiler. The comments in C++ programming can be used to provide explanation of the code, variable, method or class. By the help of comments, you can hide the program code also.

There are two types of comments in C++.

- Single Line comment
- Multi Line comment

## C++ Single Line Comment

The single line comment starts with // (double slash). Let's see an example of single line comment in C++.

```
#include <iostream>
using namespace std;
int main()
{
    int x = 11; // x is a variable
    cout<<x<<"\n";
}
```

Output:

```
11
```

## C++ Multi Line Comment

The C++ multi line comment is used to comment multiple lines of code. It is surrounded by slash and asterisk (/\* ..... \*/). Let's see an example of multi line comment in C++.

```
#include <ostream>
using namespace std;
int main()
```

```
{  
/* declare and  
print variable in C++. */  
int x = 35;  
cout<<x<<"\n";  
}
```

Output:

35

← Prev

Next →

AD

 Youtube For Videos Join Our Youtube Channel: [Join Now](#)

## Feedback

- Send your Feedback to [feedback@javatpoint.com](mailto:feedback@javatpoint.com)

## Help Others, Please Share

