break statement in C++

A C++ breaking statement in a program used to terminate control statement or execute or skip a particular statement.

The break statement in C++ is divided into three parts which are as follows,

Type of breaking statement in C++

- C++ break statement
- C++ continue statement
- C++ goto statement
- C++ exit statement

break statement in C++

The break statement is used with the conditional statement to terminate the other control statements for examples if-else, switch, for, while and do-while.

In the program, we have to define under which condition we want to terminate the statement, so it will be used with the conditional statement.

However, the switch statement has a `break` by default.

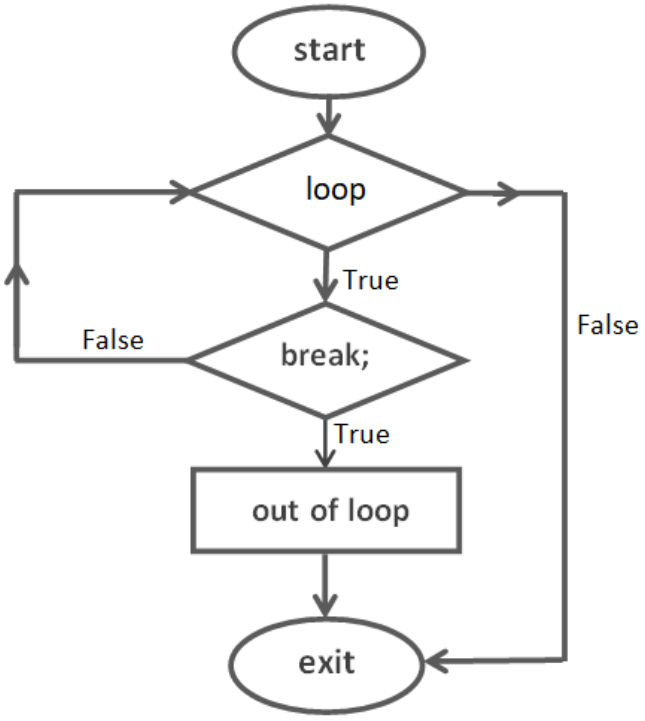
here is a syntax to use the break statement in the program,

syntax

```
{

   ……….;

   break;

 }
```

flow-diagram of break in C++

as you can see the following diagram break-statement is breaking the loop. After execution of break-statement loop will be terminated and control will be transfer to the next statement out of the loop (if available).



Here you can say that loop is terminated in two ways, first when loop condition becomes False and second when break statement condition becomes True.

Here is the program,

Example of break statement in C++

In the program below, the loop was to be executed 9th time(**count<=9**) but we have used the break keyword inside the conditional statement and defined the condition that the loop terminates as soon as the value of variable **count** is equal to variable **x**.

```
#include<iostream.h>


using namespace std;
```

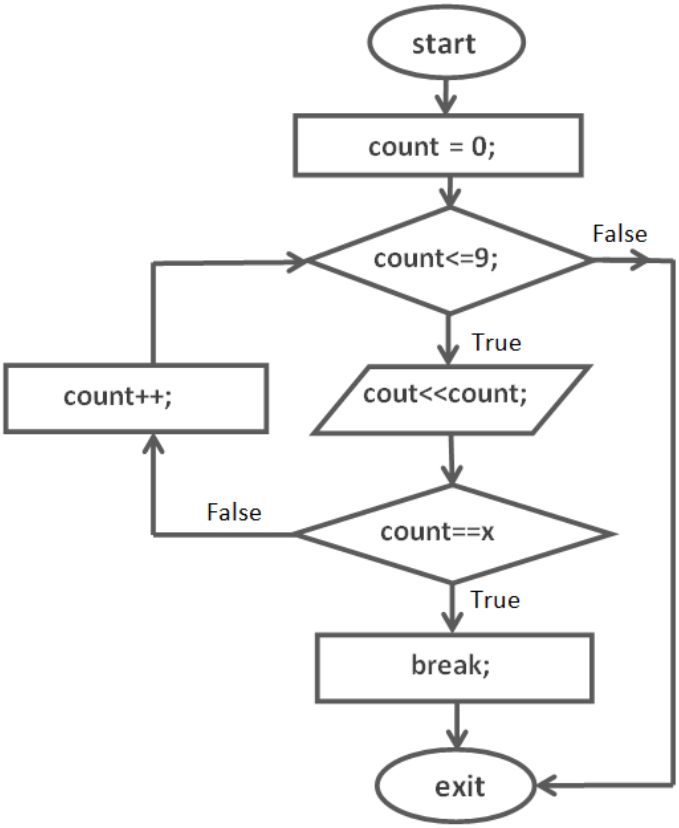
```
int main()
 {
   int x=5;
    for(int count = 0; count <= 9; count++)
      {
        cout<<count<<"\t";
        if(count==x)
           break;
      }

   cout<<"\nThis is next statement";
return 0;
 }
```

OUTPUT

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

```
This is next statement
```

Flow diagram will be as follows of the above program,



break with loop in C++

In the below program, we used break statement with do-while loop,

as we know normally a loop terminates when their condition becomes false but here we use break statement so there is no need to define a condition.

although up-dation is required to repeating the statement.

```
#include<iostream.h>

using namespace std;

 int main()

  {

    int count,input, pn = 4321;

    count = 0;

    do {

           clrscr();

           cout<<"Enter Pin: ";

           cin>>num;

            if(input==4321)

              break;

           count++;

       }while(count);//no condition is given

     cout<<"successfully matched...";
```

```
return 0;

 }
```

OUTPUT

```
Enter Pin: 4321

successfully matched...
```

Explanation

In the program, as soon as the value of a variable **pn** is equal to the variable **input**, loop will be break and execution will be go to the next statement.

Remember here, only terminate the loop, not the entire program, means the program will remain in the execution. what will be doing if we want to terminate the entire program? use exit statement in C++.

terminate an infinite loop using break statement

In the below program an infinite loop is terminated by the break statement.

```
#include<iostream>

using namespace std;

 int main()
```

```
 {

     int num1,num2;

      for(int i=0; i>=0; i++) //infinite loop

       {

          cout<<"\nEnter Two number: ";

          cin>>num1>>num2;

           if(!num1||!num2)

               break;

           else

               cout<<"sum: "<<num1+num2;

       }

      cout<<"loop break successfully..";

      getch();

 }
```

OUTPUT

```
Enter Two number: 2 3
```

```
sum: 5

Enter Two number: 4 6

sum: 10

Enter Two number: 4 6

sum: 10

Enter Two number: 4 0

sum: 10

loop break successfully..
```

Explanation:

The program will ask the user to enter the input until the user enters. 0. As soon as the value of either of the two variables is 0, the loop will break and the execution will move to the next statement outside the loop.

As mentioned above, the break keyword is by default in the switch-statement.

Try in switch-statement

without break keyword switch statement