How i++ differs from ++i

Authors:

Vadeev Dmitrii

Zdorov Iliya

```cpp
#include<iostream>
using namespace std;
int main() {
    int i = 0;

    cout << "Preincrement i" << endl;
    while (i < 10)
        cout << ++i << " ";
    cout << endl;

    i = 0;
    cout << "Preincrement i" << endl;
    while (i < 10)
        cout << i++ << " ";
    cout << endl;
    return 0;
}
```

Консоль отладки Microsoft Visual Studio

```
Preincrement i
1 2 3 4 5 6 7 8 9 10
Preincrement i
0 1 2 3 4 5 6 7 8 9

C:\Users\Илья\source\repos\Project20\Debug\Project20.exe (процесс 7960) завершил работу с кодом 0.
Нажмите любую клавишу, чтобы закрыть это окно...
```

Often a novice programmer gets lost in these i ++ and ++ i. In this short post, we will try to clarify the material so that there is no doubt that you will never get confused and will definitely correctly decide which option should be used in a particular case. Perhaps you need to start with a definition. The ++ operator is shorthand for increment. In other words, this operator is used to add 1 to a variable.

```
i = 0;
i++; // то же самое что и i=i+1
++i; // то же самое что и i=i+1
```

But why, and most importantly, in what cases i ++! = ++ i? The fact is that the increment operation, like other operations, returns the result of the same operation. What is important is the result! Having understood this, there will be no more questions.
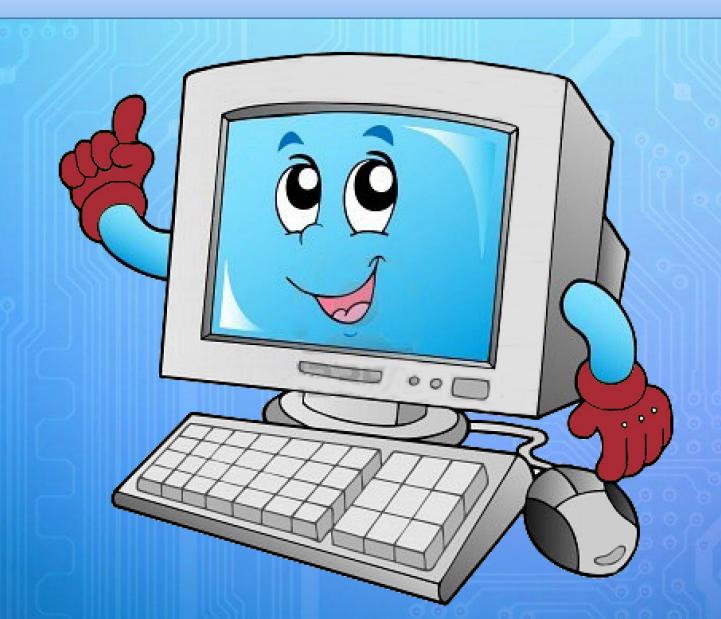
Example 1: while loop

```cpp
int i = 0;
cout << "Preincrement i" << endl;
while (i < 10)
    cout << ++i << " ";
cout << endl;

i = 0;
cout << "Postincrement i" << endl;
while (i < 10)
    cout << i++ << " ";
cout << endl;
```

```
Preiterated i
1 2 3 4 5 6 7 8 9 10
Postiterated i
0 1 2 3 4 5 6 7 8 9
```

As we can see in the first case i value is displayed after the addition operation and in the second case: before.

```
int i = 0;
for (i = 0; i < 10; i++)
    cout << i << " ";
cout << endl;

for (i = 0; i < 10; ++i)
    cout << i << " ";
cout << endl;
```

```
0 1 2 3 4 5 6 7 8 9
0 1 2 3 4 5 6 7 8 9
```

In this case, in fact, nothing changes. Those. whether you use ++ i or i ++, in the body of the loop we get i = i + 1. Therefore, you should not rack your brains over this at all. It all depends on the preferences of the programmer and does not affect the course of the program in any way.

The following example is identical to example 1 with a while loop:

```
int i = 0;
for (i = 0; i < 10;) {
    cout << i++ << " ";
}
cout << endl;

for (i = 0; i < 10;) {
    cout << ++i << " ";
}
cout << endl;
```

```
0 1 2 3 4 5 6 7 8 9
1 2 3 4 5 6 7 8 9 10
```

Summing up:

The increment operation is used quite often in all programming languages. And the fact that I gave examples in C ++, in fact, does not change absolutely anything. Throughout your career as a programmer, you will put this knowledge into practice many times. Therefore, I really hope that I succeeded, after all, to explain in detail the difference between the incremental operations i ++ and ++ i.