

# C++ Increment

## C++ Increment

---

C++ Increment operation can be done using Arithmetic Increment Operator `++` .

Increment operator takes only one operand. Based on the side of operator at which the operand is given, there are two forms for Increment Operator. They are:

- Prefix
- Postfix

In terms of execution, the main difference between prefix and postfix increments is that: Prefix increments the variable value and then participates in the expression evaluation or statement execution. But postfix first lets the expression evaluation or statement execution complete and then increments the value of the operand.

## C++ Prefix Increment

---

For Prefix Increment, the operand comes to the right of the operator. Following is an example for prefix increment.

```
++operand
```

## Example

---

In the following example, we take an integer and increment it using prefix form.

### C++ Program

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

int main() {
    int a = 12;
    ++a;
    cout << a << endl;
}
```

### Output

## C++ Postfix Increment

For Postfix Increment, the operand comes to the left of the operator.

```
operand++
```

### Example

In the following example, we take an integer and increment it using postfix form.

#### C++ Program

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

int main() {
    int a = 12;
    a++;
    cout << a << endl;
}
```

#### Output

## C++ Postfix vs Prefix Increment

Let us check the difference of postfix and prefix using an example program.

### Example

In the following example, we shall perform postfix and prefix on different variables, and then look into how they act during execution.

#### C++ Program

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

int main() {
```

```

//postfix
int a = 12;
cout << a++ << endl; //12
cout << a << endl << endl; //13

//prefix
int b = 12;
cout << ++b << endl; //13
cout << b << endl; //13
}

```

## Output

```

12
13

13
13

```

`cout << a++ << endl;` takes the value of current `a`, executes the **cout** statement, and then increments `a`. So, for the next statement, `cout << a << endl << endl;`, value of `a` is `13`.

`cout << ++b << endl;` increments the value of `b` to 13, executes the **cout** statement. So, when this statement is executed, `b` is printed as `13`.

## Increment Float Value

Increment operator can take a float variable as operand and increase its values by 1.

## Example

In the following example, we shall initialize a float variable with some value and increment its value by 1 using `++` operator.

## C++ Program

```

#include <iostream>
using namespace std;

int main() {
    float a = 12.5;
    a++;
    cout << a << endl;
}

```

## Output

```

13.5

```

## Conclusion

In this [C++ Tutorial](#), we learned what increment operator does, how to use increment operator in prefix or postfix form, with the help of example C++ programs.

### C++ Tutorials

- ◆ [C++ Tutorial](#)
- ◆ [C++ Hello World Program](#)
- ◆ [C++ If Else](#)
- ◆ [C++ Switch](#)
- ◆ [C++ Ternary Operator](#)
- ◆ [C++ Logical Operations](#)
- ◆ [C++ Arithmetic Operations](#)
- ◆ [C++ While Loop](#)
- ◆ [C++ Do-While Loop](#)
- ◆ [C++ For Loop](#)
- ◆ [C++ ForEach](#)
- ◆ [C++ Continue](#)
- ◆ [C++ Break](#)
- ◆ [C++ Comments](#)
- ◆ [C++ Recursion](#)
- ◆ [C++ Try Catch](#)
- ◆ [C++ String Operations](#)
- ◆ [C++ Array Operations](#)
- ◆ [C++ Vector Operations](#)
- ◆ [C++ Input Output Operations](#)
- ◆ [C++ Class](#)
- ◆ [C++ Programs](#)