

Logical Operators in C++

Logical operators are used to perform logical operations on the given expressions, mostly to test the relationship between different variables or values. They return a boolean value i.e., either true (1) or false (0) based on the result of the evaluation.

C++ provides the following logical operators:

AND Operator (&&) The AND operator checks if both the operands/conditions are true, then the expression is true. If any one of the conditions is false, the whole expression will be false.

```
(expression1 && expression2)
```

Example:

```
int a = 5, b = 10;
if (a > 0 && b > 0) {
    std::cout << "Both values are positive.\n";
}
```

•

OR Operator (||) The OR operator checks if either of the operands/conditions are true, then the expression is true. If both the conditions are false, it will be false.

```
(expression1 || expression2)
```

Example:

```
int a = 5, b = -10;
if (a > 0 || b > 0) {
    std::cout << "At least one value is positive.\n";
}
```

•

NOT Operator (!) The NOT operator reverses the result of the condition/expression it is applied on. If the condition is true, the NOT operator will make it false and vice versa.

```
!(expression)
```

Example:

```
int a = 5;

if (!(a < 0)) {
    std::cout << "The value is not negative.\n";
}
```



Using these operators, you can create more complex logical expressions, for example:

```
int a = 5, b = -10, c = 15;

if (a > 0 && (b > 0 || c > 0)) {
    std::cout << "At least two values are positive.\n";
}
```

This covers the essential information about logical operators in C++.