Logical Operators in C++ (&&, ||, !) — Complete Professional Guide

The Three Logical Operators

AND (&&) — True only if both expressions are true. Uses short-circuiting: if the left side is false, the right side is not evaluated.

OR (||) — True if at least one expression is true. Short-circuits when the left side is true.

NOT (!) — Reverses the truth value of the expression.

Understanding Short-Circuiting

Short-circuiting means C++ stops evaluating a condition once the final result is already known.

Example 1 — Preventing Crashes

This ensures ptr->value is accessed only if ptr is valid.

```
if (ptr && ptr->value > 10) {
    cout << ptr->value;
}
```

Example 2 — OR Operator Optimization

If marks > 90 is true, the second check isn't needed.

```
if (marks > 90 || hasSportsCertificate()) {
    cout << "Eligible";
}
```

Example 3 — Short-Circuit Demonstration

bar() will not run if foo() returns false.

```
bool foo() { cout << "foo "; return false; }
bool bar() { cout << "bar "; return true; }

if (foo() && bar()) {
    cout << "done";
```

```
}
```

Why Short-Circuiting Matters

- Prevents invalid memory access
- Avoids expensive unnecessary operations
- Ensures safe evaluation order

Logical vs Bitwise Operators

Logical operators (&&, ||, !) perform boolean logic and short-circuit.

Bitwise operators (&, |) operate on bits and **do not** short-circuit.

```
// Logical AND (safe)
if (ptr && ptr->value > 10) {}

// Bitwise AND (dangerous)
if (ptr & ptr->value > 10) { /* Wrong */ }
```

NOT Operator Usage

- **!flag** reverses a boolean
- **!!x** converts any number into true/false
- **if (!ptr)** checks for null pointers

Real-World Usage Patterns

These are common idioms in real C++ codebases:

```
// Safe pointer usage
if (user && user->active) {}

// Avoid division by zero
if (den != 0 && num/den > 2) {}

// Safe array access
if (i >= 0 && i < arr.size()) {
    process(arr[i]);
}
```

How && and || Work Internally

AND (&&): If left side is false, result is false immediately.

OR (||): If left side is true, result is true immediately.