

Operators are special symbols that perform specific operations on one or more operands. C++ has a rich set of operators with many different capabilities.

The most basic operators are the **arithmetic operators**, which perform basic mathematical operations like addition, subtraction, multiplication, and division. These operators are typically used on two operands, but the division operator can also be used on a single operand to calculate the reciprocal.

There are also **assignment operators**, which assign a value to a variable. The most common assignment operator is the `=` operator, which assigns the value of the right operand to the left operand. Other assignment operators include the `+=`, `-=`, `*=`, and `/=` operators, which perform the addition, subtraction, multiplication, or division operation and then assign the result to the left operand.

The **comparison operators compare** two values and return a boolean result indicating whether the comparison is true or false. The most common comparison operator is the `==` operator, which checks if two values are equal. Other comparison operators include the `!=` operator (not equal), the `<` and `>` operators (less than and greater than, respectively), and the `<=` and `>=` operators (less than or equal to and greater than or equal to, respectively).

There are also some operators that perform conditional evaluations. The most common of these is the `?:` operator, which is often called the **ternary operator** because it takes three operands. This operator evaluates the first operand and, based on the Boolean value of that operand, returns either the value of the second operand or the value of the third operand.

Finally, there are the **increment and decrement operators**. The `++` operator increments its operand by 1, and the `--` operator decrements its operand by 1. These operators can be used as prefix or postfix operators. When used as a prefix operator, the increment or decrement operation is performed before the value of the operand is used in the rest of the expression. When used as a postfix operator, the increment or decrement operation is performed after the value of the operand is used in the rest of the expression.

