Operators

C operators are symbols that are used to perform mathematical or logical manipulations. The C programming language is rich with built-in operators. Operators take part in a program for manipulating data and variables and form a part of the mathematical or logical expressions.

C programming language offers various types of operators having different functioning capabilities.

1. Arithmetic Operators

Arithmetic Operators are used to performing mathematical calculations like addition (+), subtraction (-), multiplication (\*), division (/) and modulus (%).

1. Relational Operators

|  |  |
| --- | --- |
| == | Is equal to |
| != | Is not equal to |
| > | Greater than |
| < | Less than |
| >= | Greater than or equal to |
| <= | Less than or equal to |

1. Logical Operators

|  |  |
| --- | --- |
| && | And operator. It performs logical conjunction of two expressions. (if both expressions evaluate to True, result is True. If either expression evaluates to False, the result is False) |
| || | Or operator. It performs a logical disjunction on two expressions. (if either or both expressions evaluate to True, the result is True) |
| ! | Not operator. It performs logical negation on an expression. |

1. Assignment Operators

|  |  |
| --- | --- |
| = | Assign |
| += | Increments then assign |
| -= | Decrements then assign |
| \*= | Multiplies then assign |
| /= | Divides then assign |
| %= | Modulus then assign |

1. Increment and Decrement Operators

Increment and Decrement Operators are useful operators generally used to minimize the calculation, i.e. ++x and x++ means x=x+1 or -x and x−−means x=x-1. But there is a slight difference between ++ or −− written before or after the operand. Applying the pre-increment first add one to the operand and then the result is assigned to the variable on the left whereas post-increment first assigns the value to the variable on the left and then increment the operand.

1. Conditional Operator

|  |  |
| --- | --- |
| ? : | Conditional Expression |

1. Bitwise Operators

|  |  |
| --- | --- |
| << | Binary Left Shift Operator |
| >> | Binary Right Shift Operator |
| ~ | Binary Ones Complement Operator |
| & | Binary AND Operator |
| ^ | Binary XOR Operator |
| | | Binary OR Operator |

1. Special Operators

|  |  |
| --- | --- |
| sizeof() | Returns the size of a memory location. |
| & | Returns the address of a memory location. |
| \* | Pointer to a variable. |