



Operator: Operators are those characters or group of characters from the character set of 'C' programming language that are used with data items (constants, variables or both together) to result some value. Such as: +, -, *, /, <, >, <=, >=, ==, !=, &&, ||, ++, --, !, ?: etc.

The data item used with operator are also known as "Operand".

The operators are classified into two categories, as

(1) **Unary Operator:** These operators require only one data items to work with. These operators are: -, +, !, ++ (increment operator) and -- (decrement operator).

- (Unary minus) operator is used to represent a numeric quantity in negative form.

ex: -5, a=-5, b=-a;

+ (Unary plus) operator is used to represent a numeric quantity in positive form.

ex: +5, a=+5;

Note: These operators can be used with either a constant or a variable.

! (Not) operator is used with a conditional expression to get the opposite result of the condition. The condition always evaluates to be either true or false.

!(a>b), !(num1>=num2&&num2>=num3)

Note: in 'C' programming language the false value is represented by numeric value 0 and any positive value represents 1.

++ (Unary increment operator) is used to increase the value of the variable (used with this operator in the expression) by 1.

-- (Unary decrement operator) is used to decrease the value of the variable (used with this operator in the expression) by 1.

Both the unary increment and decrement operators can be used in two ways as:

a. Prefix: operator (++ or --) appears before the data item (operand). Hence the task of operator is performed before using the value in the expression. (It has the highest precedence among all available operators in 'C')

b. Postfix: operator (++ or --) appears after the data item (operand). Hence the task of operator is performed after using the value in the expression. (It has the lowest precedence among all available operators in 'C')

Example: ++a, a++, -- a, a --;

(2) **Binary Operator:** These operators are always used with two data items (operands). These are classified into following types, as:

a. Arithmetic operator: +, -, *, /, % (% used with integers and not with float or double or long double)

example: a+b, a-b, a*b, a/b, a%b (if a and b both are of type integers)

b. Relational operator: <, >, <=, >=, ==, !=

example: a<b, a>b, a<=b, a>=b, a==b, a!=b;

c. Logical operator: &&, ||

example: (num1>=num2&&num1>=num3), (ch=='a' || ch=='A')

d. Assignment operator: =, +=, -=, *=, /=, %= (%= used with integers and not with float or double or long double)

example: a+=b; a-=5, a*=b, a/=10; a%=10; (if a is of type integer)

e. Conditional operator: ?: (It is also known as ternary operator)

example: int res=(num1>=num2?num1:num2);

Some other operators (membership operator, Address operator, indirection operator, de-reference operator, bitwise operator etc. are also available in 'C' that will be discussed later.