Programming Fundamentals:

Operators

Operators

Different operations on variables and values are performed using operators. Operators are symbols that can be used in code to guide computers on what type of operation should be performed. In C++, there are five types of operators

- Arithmetic Operators
- Relational Operators
- Logical Operators
- Bitwise Operators
- Assignment Operators
- Ternary or Conditional Operators

Arithmetic Operators: These operators are used to compute basic mathematical operations like addition or division.

Operator	Name	Detail	Example
+	Addition	Adds together two values	x + y
-	Subtraction	Subtracts one value from another	x - y
*	Multiplication	Multiplies two values	x * y
/	Division	Divides one value by another	x / y
%	Modulus	Returns the division remainder	x % y

Assignment Operators: These operators are used to assign value.

Operator	Detail	Example	Same As
=	To Assign the value from RHS to LHS variable	x = 5	x = 5
+=	To add more in same variable	x += 3	x = x + 3
-=	To minus something in same variable	x -= 3	x = x - 3
*=	To multiply something with same variable	x *= 3	x = x * 3
/=	To divide something from same variable	x /= 3	x = x / 3
%=	To get the modules of variable and assign to same	x %= 3	x = x % 3

Unary Operators: These operators are used one single variable

Operator	Name	Details	Example
++	Increment	Increases the integer value of the variable by one	a++, ++a
--	Decrement	Decreases the integer value of the variable by one	b--, --b

Prefix notation: It assigns the value after performing operation, i.e., ++var
Postfix notation: It assigns the value before performing operation, i.e., var--

Comparison Operators: These operators are used to compare the values of two variables

Operator	Name	Detail	Example
==	Equal to	It check if two variables have same values or not	x == y
!=	Not equal	It check if two variables have different values or not	x != y
>	Greater than	It checks if the variable at LHS has a greater value as compared to the variable at RHS	x > y
<	Less than	It checks if the variable at LHS has a smaller value as compared to the variable at RHS	x < y
>=	Greater than or equal to	It checks if the variable at LHS has a greater or same value as compared to the variable at RHS	x >= y
<=	Less than or equal to	It checks if the variable at LHS has a smaller or same value as compared to the variable at RHS	x <= y

Logic Table

Logic Table of AND

a	b	R
0	0	0
0	1	0
1	0	0
1	1	1

Logic Table of OR

a	b	R
0	0	0
0	1	1
1	0	1
1	1	1

Logic Table of XOR

a	b	R
0	0	0
0	1	1
1	0	1
1	1	0

Logic Table of NOT

a	R
0	1
1	0

Logical Operators: These operators use for logical operations.

Operator	Name	Description	Example
&&	Logical and	Returns true if both statements are true	x < 5 && x < 10
||	Logical or	Returns true if one of the statements is true	x < 5 || x < 4
!	Logical not	Reverse the result, returns false if the result is true	!(x < 5 && x < 10)

Bitwise Operators: These operators use to perform logical operations at bit level

Operator	Name	Description	Example
&	Bitwise and	Returns true if both bits are true	x & y
|	Bitwise or	Returns true if any of the bit is true	x | y
~	Bitwise inverse	Reverse the bit, returns 0 if the bit is 1 and vice versa	x ~ y
<<	Bitwise Left Shift	Shift the bit values toward left	x << y
>>	Bitwise Right Shift	Shift the bit values toward Right	x >> y