Operators

Operators

- An operator is a symbol, which helps the user to command the computer to do a certain mathematical or logical manipulations.
- Operators are used in programming language program to operate on data and variables.

- Operators can be classified as:
  - Arithmetic operators
  - Relational Operators
  - Logical Operators
  - Increments and Decrement Operators

Arithmetic Operators

Operator	Name	Description
+	Addition	to add two numbers together
-	Subtraction	to subtract one number from another
*	Multiplication	to multiply one number by another.
/	Division	to divide one number by another.
%	Modulus (Remainder)	to find the remainder from dividing one number by another.

Arithmetic Operators

C operation	Arithmetic operator	Algebraic expression	C expression
Addition	+	$f + 7$	`f + 7`
Subtraction	–	$p - c$	`p - c`
Multiplication	*	bm	`b * m`
Division	/	x / y or $\frac{x}{y}$ or $x \div y$	`x / y`
Remainder	%	$r \bmod s$	`r % s`

Arithmetic Operators

- Example:
  1. 5+3=8
  2. 5−3=2
  3. 5*3=15
  4. 5/3=1
  5. 5%3=2

Precedence of arithmetic operators

Operator(s)	Operation(s)	Order of evaluation (precedence)
( )	Parentheses	Evaluated first. If the parentheses are nested, the expression in the innermost pair is evaluated first. If there are several pairs of parentheses "on the same level" (i.e., not nested), they're evaluated left to right.
*	Multiplication	Evaluated second. If there are several, they're evaluated left to right.
/	Division	
%	Remainder	
+	Addition	Evaluated last. If there are several, they're evaluated left to right.
-	Subtraction	

- Example

1. $2 + 5 * 4 - 3 = ?$
2. $(2 + 5) * (4 - 3) = ?$

Integer Arithmetic

- Let x=27 and y=5 be two integer numbers. Then the integer operation leads to the following results:
  - x + y = 32
  - x –y = 22
  - x * y = 115
  - x % y = 2
  - x / y = 5

Floating-point Arithmetic

- Let x = 14.0 and y = 4.0 then
  - x + y = 18.0
  - x –y = 10.0
  - x * y = 56.0
  - x / y = 3.50

Relational Operators

- An operator that compares two values.

  x < 5

- This expression will have a value of TRUE if the variable x is less than 5; otherwise the value of the expression will be FALSE.

- Relational operators are some times called comparison operators.

Equality and relational operators

Algebraic equality or relational operator	C equality or relational operator	Example of C condition	Meaning of C condition
Equality operators			
=	==	x == y	x is equal to y
≠	!=	x != y	x is not equal to y
Relational operators			
>	>	x > y	x is greater than y
<	<	x < y	x is less than y
≥	>=	x >= y	x is greater than or equal to y
≤	<=	x <= y	x is less than or equal to y

Relational Operators

- Let x=2 and y=5 then
  - x < y
  - (x + 2) > (y * 2)
  - (x + 3) <= y
  - x != y
  - y > (3 + x)

Relational Operators

- Let x=2 and y=5 then
  - x < y — True
  - (x + 2) > (y * 2) — False
  - (x + 3) <= y — True
  - x != y — True
  - y > (3 + x) — False

Precedence and associatively of the operators

Operators	Associativity
()	left to right
* / %	left to right
+ -	left to right
< <= > >=	left to right
== !=	left to right
=	right to left

Logical Operators

- An operator that compare or evaluate logical and relational expressions.

Logical Operators

Operator	Description
&&	Called Logical AND operator. If both the operands are non-zero, then the condition becomes true.
||	Called Logical OR Operator. If any of the two operands is non-zero, then the condition becomes true.
!	Called Logical NOT Operator. It is used to reverse the logical state of its operand. If a condition is true, then Logical NOT operator will make it false.

- **Logical AND**
  - (a > b) && (x == 10)
  - The expression to the left is a>b and that on the right is x==10, the whole expression is true only if both expressions are true i.e.,if a is greater than b and x is equal to 10.

Logical AND

- Given a=2, b=3 and c=5, evaluate the following logical expressions:

- (a > b) && (c != 5)
- (a < b) && (c < b)
- (a > b) && (c == 5)
- (a < b) && (b < c)

Logical AND

- Given a=2, b=3 and c=5, evaluate the following logical expressions:

- (a > b) && (c != 5) False
- (a < b) && (c < b) False
- (a > b) && (c == 5) False
- (a < b) && (b < c) True

Logical OR

- (a < m) || (a < n)
- The expression evaluates to true if any one of them is true or if both of them are true.

- Given a=2, b=3 and c=5, evaluate the following logical expressions:
  - (a > b) || (c != 5) False
  - (a < b) || (c < b) True
  - (a > b) || (c == 5) True
  - (a < b) || (b < c) True

Logical NOT

- ! (x >= y)

Increment and Decrement Operators

- The increment and decrement operators are one of the unary operators which are very useful in programming language.
- They are extensively used in loops.
- The syntax of the operators is given below:
  - ++ variable name
  - variable name++
  - ––variable name
  - variable name––

Increment and Decrement Operators

- The increment operator ++ adds the value 1 to the current value of operand.
- The decrement operator ––subtracts the value 1 from the current value of operand

Increment and Decrement Operators

- m = 5;
- y = ++m;

```
//(prefix) In this case the value of y
  and m would be 6.
```

- m = 5;
- y = m++;

```
//(postfix)  Then the value of y will be
  5 and that of m will be 6.
```

Increment and Decrement Operators

- A prefix operator first adds 1 to the operand and then the result is assigned to the variable on the left.
- On the other hand, a postfix operator first assigns the value to the variable on the left and then increments the operand.

- Example 1:

```
x = 4
y = ++x
PRINT x
PRINT y
```

- Example 1:

```
x = 4
y = ++x
PRINT x
PRINT y
```

- What is the output?

  4

  5

- Example 2:

  x = 3

  y = x++

  PRINT x

  PRINT y

- What is the output?

  3

  3