Agenda

1. Operators
2. Arithmetic operators
3. Precedence of operators
4. Boolean operators
5. Comparison operators
6. Assignment operators
7. Logical Operators

Arithmetic Operators

{ +, -, *, /, //, **, % }

- + : Add
- - : Subtract
- * : Multiply
- / : Float Division
- // : Integer Division
- ** : Power / Exponentiation
- % : Modulus / Remainder

12 + 13 = 25

20 - 12 = 8

20 - 23 = -3

2 * 5 = 10

5/2 = 2.5

10/3 = 3.33...

10 = Q * 3 + R

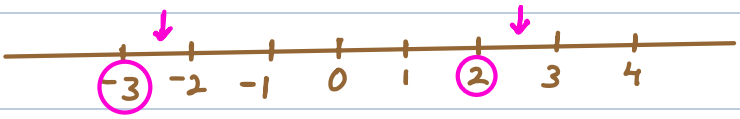
- 10 : dividend
- 3 : divisor
- Q : Quotient
- R : Remainder
- R <= 10

10 // 3 = 3

20 // 8 = 2

50 // 11 = 4

50 = 4 * 11 + 6



-5 // 2 = -3

-5 = -3 * 2 + 1

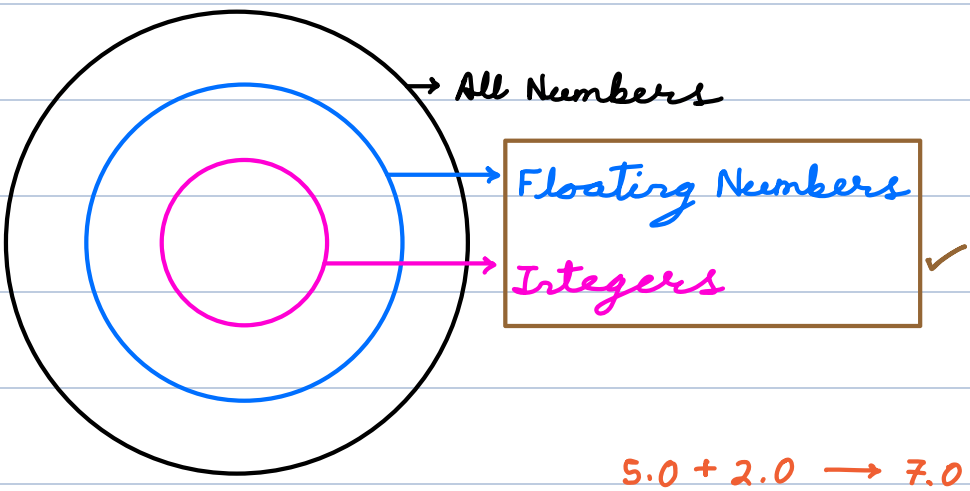
-6

-9 // 4 = -3

-3 * 4 = -12

-9 = Q * 4 + R

max value <= -9



5.0 + 2.0 → 7.0 ✓

5.0 + 2 → 7.0 ✓

10.0 // 3 → 3.0 (resultant int converted to float)

Modulus

$x \,\%\, y = r \Rightarrow x = q * y + r$ → $[0 \quad (y-1)]$

$q * y$: max value <= x

$20 \,\%\, 3 \rightarrow \quad 20 = 6 * 3 + 2$

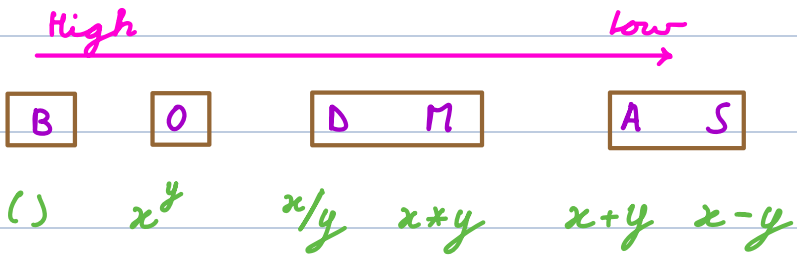
$= 2$

$-10 \,\%\, 3 \rightarrow \quad -10 = -4 * 3 + 2$

$= 2$

(-1 in Java/C++/C# etc.)

Precedence of Operators



For same precedence → go from left to Right.

$10 - 4 * 2 + 5 - 6 / 3$

$= 10 - 8 + 5 - 2.0$

$= 2 + 5 - 2.0 = 7 - 2.0 = \underline{5.0}$

Boolean

`bool()`

- Every number except 0 is True.
- Every non-empty string is True.

`bool(20)` → True

`bool(-5)` → True

Comparison Operators

{ `>` `<` `>=` `<=` `==` `!=` }

- `>` greater
- `<` lesser
- `>=` greater or equal
- `<=` lesser or equal
- `==` equal
- `!=` not equal

`7 > 5` → True

`-7 < -5` → True

`-7 == -6` → False

`-7 > -5` → False

`-7 >= -7` → True

`-7 == -7` → True

`-7 != -6` → True

Assignment Operator { = }

a = 5 a == 5 // checks if a is equal to 5.

Logical Operators

{ and , or , not }

Check if x is greater than y & z.

(x > y) and (x > z)

Check if x is greater than y or less than z.

(x > y) or (x < z)

Truth Table

0 → False 1 → True

x	y	x and y	x or y
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	1

not (6 < 10) → False

(6 < 10) → True