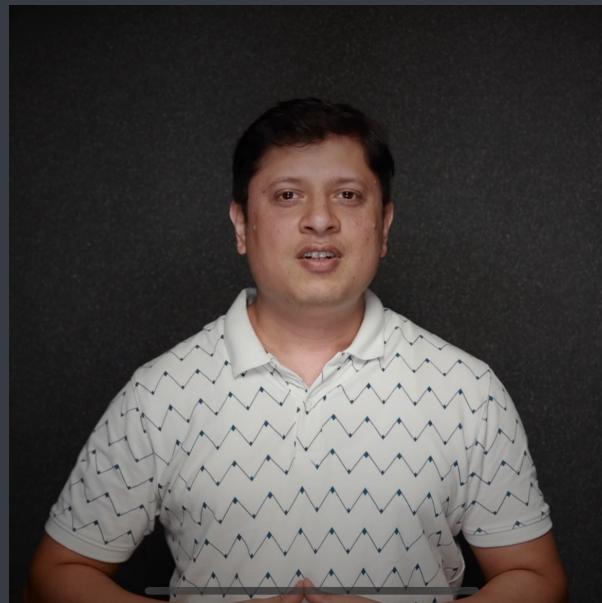


C Language

Operators in C Language



Saurabh Shukla (MySirG)

Agenda

- ① Arithmetic Instruction
- ② Classification of operators
- ③ Unary operators
- ④ Arithmetic operators
- ⑤ Bitwise operators
- ⑥ Relational operators
- ⑦ Logical operators
- ⑧ Assignment operators

Arithmetic Instruction

An instruction which is used to manipulate data using operators, is known as Arithmetic Instruction.

The diagram illustrates the components of an arithmetic expression. At the top right, the text "Data/operand" is written in blue. Two blue arrows point downwards from this text to the numbers "3" and "4" in a green "3 + 4" expression. Below this, the number "9" is shown in pink, with a pink arrow pointing upwards to it from the word "Operator" at the bottom. To the right of the "3 + 4" expression, the expression "3 + 4 * 5" is written in pink, followed by an equals sign and the result "23" in orange.

$$3 + 4$$

9

Operator

Data/operand

$$3 + 4 * 5$$
$$= 23$$

Classification of Operators

- ① Unary operators +, -, ++, --, sizeof()
- ② Arithmetic operators *, /, %, +, -
- ③ Bitwise Operators &, |, ^, ~, >>, <<
- ④ Relational Operators <, >, <=, >=, ==, !=
- ⑤ Logical Operators !, &&, ||
- ⑥ Conditional Operator ?:
- ⑦ Assignment Operators =, +=, -=, *=, /=, %=

Operands

- ① unary 1
- ② Binary 2
- ③ Ternary 3

Unary Operators

+ , - , ++ , --

+3 , -5

Increment operator $x++$

int $x = 5;$

printf("%d", x); 5

5 6 7

post increment

$x++;$ $x = x + 1$

printf("%d", x); 6

pre increment

$++x;$ $x = x + 1$

printf("%d", x); 7

Decrement operator

--

$x--;$ post decrement

$--x;$ pre decrement

$x = x - 1$

Find Output of the following program?

```
#include<stdio.h>
int main()
{
    int x=5, y;
    y=x++;
    printf("%d %d", x, y);
}
```

x y

6

5

Find Output of the following program?

```
#include<stdio.h>
int main()
{
    int x=5, y;
    y=++x;
    printf("%d %d", x, y);
}
```



6 6

Unary Operators

sizeof()

- ① Data type
- ② Variable
- ③ Constant

int x;

x = sizeof(float);

printf("%d", x); 4

x = sizeof(double);

printf("%d", x); 8

x = sizeof(char);

printf("%d", x); 1

```
int x, y;  
float m;  
char ch;  
double d1;  
8 x = sizeof(d1);  
1 x = sizeof(ch);  
4 x = sizeof(y);  
4 x = sizeof(m);
```

x = sizeof(35);	4
x = sizeof(4.7);	8
x = sizeof('A');	4

Arithmetic Operators (L to R)

$*$ / %

$15 \% 4$

3

$$\begin{array}{r} 4 \sqrt{15} \\ \underline{-12} \\ 3 \end{array}$$

$a * b / c$

*

+ -

$3 + 4$ 7

$3 - 4$ -1

$3 * 4$ 12

$3 / 4$ 0

0.75 X

$a / b * c$

/

$a + b * c$

*

$$\begin{array}{r} 20 \% 3 \quad 2 \\ 71 \% 10 \quad 1 \\ 7 \% 9 \quad 7 \\ \hline 0 \end{array}$$

$$9 \sqrt{70}$$

Integer operator Integer \Rightarrow Integer

$3.0 / 4$ 0.75

$3 / 4.0$ 0.75

$3.0 / 4.0$ 0.75

2.5 % 2 error

$a / 10 \rightarrow$ a without last digit

$$213 / 10 \rightarrow 21$$

$$1048 / 10 \rightarrow 104$$

$$25 / 10 \rightarrow 2$$

$a \% 10 \rightarrow$ last digit of a

$$\begin{array}{l} x/5 \rightarrow 3 \\ 15/5 \rightarrow 3 \\ 16/5 \rightarrow 3 \\ 17/5 \rightarrow 3 \\ 18/5 \rightarrow 3 \\ 19/5 \rightarrow 3 \end{array} \quad \left. \right\} x \% 5 \rightarrow 0$$

Bitwise Operators

8 | ^ ~ , >>, <<
 AND 'OR' XOR NOT Right Shift Left Shift

Binary digit is called
 a bit.
 0 & 1 are bits

0 & 0 → 0	0 0 → 0	0 ^ 0 → 0	~ 0 → 1
0 & 1 → 0	0 1 → 1	0 ^ 1 → 1	~ 1 → 0
1 & 0 → 0	1 0 → 1	1 ^ 0 → 1	
1 & 1 → 1	1 1 → 1	1 ^ 1 → 0	

int x = 25 & 100;
 128 64 32 16 8 4 2 1
 25 = 0 0 0 1 1 0 0 1

100 = 0 1 1 0 0 1 0 0
 0 = 0 0 0 0 0 0 0 0

$x = 35 \& 12;$

$$35 = 00100011$$

$$12 = 00001100$$

$$\sigma = \overline{0000\ 0000}$$

$x = 23 | 47;$

$$23 = 00010111$$

$$47 = 00101111$$

$$63 = \overline{00111111}$$

$x = 25 \gg 2;$

$25 = 00000000\ 00000000\ 00000000\ 00011001$

$6 = 00000000\ 00000000\ 00000000\ 000000110$

$x = 12 \ll 3;$

$12 = 00000000\ 00000000\ 00000000\ 00001100$

$96 = 00000000\ 00000000\ 00000000\ 01100000$

Relational Operators (L to R)

<, >, <=, >=. (Comparison operators)

==, !=

True → 1
False → 0

x = 3 > 4 ; 0

x = 4 != 3 ; 1

x = 4 <= 4 ; 1

x = 1 == 2 > 1 ;

x = 5 > 4 > 3 ; 0
| > 3

1 == 1
|

Logical Operators

! NOT(unary)

! True \rightarrow False

&& AND

! False \rightarrow True

|| OR

Every non-zero value
is True

zero is False

$x = !5 > -2 ; !0 > -2$

$x > 0 \& \& y > 0$

Exp1 || Exp2

Exp1 && Exp2

T	&&	T	\rightarrow	T
T	&&	F	\rightarrow	F
F	&&	X	\rightarrow	F

F		F	\rightarrow	F
F		T	\rightarrow	T
T		X	\rightarrow	T

age ≥ 18

nationality == 1

Assignment Operators R to L

=

= Assignment

== equal to

$x = 4;$

$4 = x;$ error

variable = anything

int $x=5;$

x
[5]

$x = x + 3;$

↑

content

Container

$y = x = 3;$

Compound Assignment Operators

$+ = , - = , * = , / = , \% =$

int $x = 5;$

$x += 4;$

$x = x + 4$

$x -= 3;$

$x = x - 3$

$x *= 2;$

$x = x * 2$

$x /= 6;$

$x = x / 6$

$x \% = 5;$

$x = x \% 5$

int $x = 5;$
 $x *= 3 + 4;$

35

int $x = 5;$
 $x = x * 3 + 4;$

19