OPERATORS IN C LANGUAGE

Friday, 1. July 2022 11:26

ACTION

① I/O instructions

② Arithmetic instruction

③ Control instruction

ARITHMETIC INSTRUCTION

An instruction which is used to manipulate data using operators, is known as arithmetic

i.e 3+4 → operands | data

↓

operator

(i) Unary operators

(ii) Binary operators

(iii) Ternary operators

OPERATORS

① Unary operators = +, -, ++, --, sizeof()

② Arithmetic operators

③ Bitwise operators

④ Relational operators

⑤ Logical operators

⑥ conditional operators

⑦ Assignment operators

n++ → post increment (less priority)

++n → pre increment (high priority)

n-- → post decrement (less)

--n → pre decrement (High)

② Arithmetic operator (Left to right)

* / %

+ -

a*b/c *

a/b*c /

a%b*c *

In C language

3/4 = 0 because when we divide an integer by an integer then result will be alway integer.

3.0/4 = 0.75

3/4.0 = 0.75

3.0/4.0 = 0.75

③ Bitwise operators

& | ^ ~ >> <<

and or XOR NOT Right shift Left shift

0 & 0 = 0	0 ^ 0 = 0
0 & 1 = 0	0 ^ 1 = 1
1 & 0 = 0	1 ^ 0 = 1
1 & 1 = 1	1 ^ 1 = 0
0 | 0 = 0	~0 = 1
0 | 1 = 1	~1 = 0
1 | 0 = 1	
1 | 1 = 1	

29 = 0 0 1 1 1 0 1

2^5 2^4 2^3 2^2 2^1 2^0

32 16 8 4 2 1

1 1 1 0 0 1

47 = 0 0 1 0 1 1 1 1

2^7 2^6 2^5 2^4 2^3 2^2 2^1 2^0

128 64 32 16 8 4 2 1

0 0 1 0 1 1 1 1

```
int n;
n = 47 & 29;
```

```
  47 = 0 0 1 0 1 1 1 1
& 29 = 0 0 0 1 1 1 0 1
-----------------------
       0 0 0 0 1 1 0 1
       8 4 1 = 13
```

∴ 47 & 29 = 13

n = 106 >> 2

binary of 106

0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 1 0

So we have to do right shifting

④ Relational operators

< > <= >=

True = 1

False = 0

3 < 2 1

5 >= 5 1

2 == 3 0

5 > 4 > 3 0

(5 > 4 → 1)

! NOT (unary)

&& AND

|| OR

!T ⇒ F

!F ⇒ T

⑤ Assignment operator

= += -= *= /= %=