

```
int n1 = 10 ;
```

```
int n3 = n1++; // tương đương 2 lệnh : n3 = n1; n1 = n1+1
```

```
// => n3 = 10 và n1 = 11
```

```
int n3 = ++n1; // tương đương 2 lệnh : n1 = n1+1; n3 = n1;
```

```
// => n3 = 11 và n1 = 11
```

LOGICAL OPERATORS : AND , OR, NOT, XOR

AND &&	true	false
true	true	False
false	false	false

Vd : a = true, b = false, c = true

=> a && b = false, a&& c = true

OR	true	false
true	true	true
false	true	false

Vd : a = true, b = false, c = true , d = false

=> a || b = true, a|| c = true, b|| d = false

XOR ^	true	false
true	false	true
false	true	false

Vd : a = true, b = false, c = true , d = false

=> a ^ b = true, a^c = false, b^d = false

BITWISE LOGICAL OPERATORS : AND , OR, NOT, XOR

là phép toán luận lý trên hệ số nhị phân (2 ký hiệu số : 0 và 1, cơ số là 2)

AND &	1	0
1	1	0
0	0	0

Vd : a = 14 , b = 29

a = 0000 1110

b = 0001 1101

$$0000 \ 1100 \Rightarrow 1 \cdot 2^3 + 1 \cdot 2^2 = 8 + 4 = 12(d)$$

$\Rightarrow a \& b = 12$

OR	1	0
1	1	1
0	1	0

Vd : a = 14 , b = 29

a = 0000 1110

b = 0001 1101

$$0001 \ 1111 \Rightarrow 2^4 + 2^3 + 2^2 + 2^1 + 2^0 = 16 + 8 + 4 + 2 + 1 = 31(d)$$

$\Rightarrow a | b = 31$

XOR ^	1	0
1	0	1
0	1	0

Vd : a = true, b = false, c = true , d = false

$\Rightarrow a \wedge b = \text{true}, a \wedge c = \text{false}, b \wedge d = \text{false}$

Hệ thập phân (10 ký hiệu số : 0 1 2 3 4 5 6 7 8 9, cơ số 10)

$$20123134 = 2 \cdot 10^7 + 1 \cdot 10^5 + 2 \cdot 10^4 + 3 \cdot 10^3 + 1 \cdot 10^2 + 3 \cdot 10^1 + 4 \cdot 10^0$$

%d -> int

%f → float, double

%c → char, vi du 'a'

%s → string, vi du "HELLO"