int n1 = 10 ;

int n3 = n1++ ; // *tuong duong 2 lenh :* n3 = n1; n1 = n1+1

// => n3 = 10 va n1 = 11

int n3 = ++n1 ; // *tuong duong 2 lenh :* n1 = n1+1; n3 = n1;

// => n3 = 11 va 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  
la phep toan luan ly tren he so nhi phan (2 ky hieu so : 0 va 1, co so la 2)

|  |  |  |
| --- | --- | --- |
| AND & | 1 | 0 |
| 1 | 1 | 0 |
| 0 | 0 | 0 |

Vd : a = 14 , b = 29

**a = 0000 1110  
b = 0001 1101  
 0000 1100** => 1\*2^3 + 1\*2^2 = 8+4 = 12(d)  
  
=> 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** => 2^4 + \*2^3 + \*2^2 + 2^1 + 2^0 = 16+8+4+2+1 = 31(d)  
  
=> a | b = 31

|  |  |  |
| --- | --- | --- |
| XOR ^ | 1 | 0 |
| 1 | 0 | 1 |
| 0 | 1 | 0 |

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

He thap phan (10 ky hieu so : 0 1 2 3 4 5 6 7 8 9, co so 10)

20 123 134 = 2\*10^7 + 1\*10^5 + 2\*10^4 + 3\*10^3 + 1\*10^2 + 3\*10^1 + 4\*10^0

%d -> int

%f 🡪 float, double

%c 🡪 char, vi du ‘a’

%s 🡪 string, vi du “HELLO”