Gán liên tiếp : a=b=c=10;

4 kiểu toán tử : toán tử số học, toán tử quan hệ và luận lý, toán tử kết hợp,toán tử luận lý nhị phân

+; toán tử số học : ++I % 7, 5 + (c = 3 +8), a\*(b + c/d) -22

+; toán tử quan hệ và luận lý : >, >=, <, <=,==,!= ví dụ : if(a>b) true; else false;

1 là true, 0 là false

+; toán tử kết hợp : &&(and),II(or),!(not) . VÍ dụ : if(a>10) && (a<20)

! tác độc trên các giá trị riêng lẻ, chuyển đổi true thành false và ngược lại.

XOR ( 1x1=1, còn lại bằng 0 )

+; toán tử luận lý nhị phân :

Chuyển đổi giá trị đầu vào thành giá trị nhị phân

And ( 1x1=1, còn lại bằng 0 ) ví dụ 5 && 1 = 1

OR( 0x0 = 0, còn lại bằng 1) ví du a II b =15

NOT(tác độc trên các giá trị riêng lẻ, chuyển đổi true thành false và ngược lại. )

XOR ( khác nhau bằng 1, giống nhau bằng 0 ) ví dụ x ^ y

Toán dịch bit:

X >> 5 ; dịch phải 5 bit ( 1 số có 8 bit thôi dịch ra ngoài là thành 0 )

Y << 2; dịch trái 2 lần

Ví dụ : a = a | (1<<9); 1 dịch trái 9 lần rồi or vs a được a mới.

2 phép toán cơ bản :

1, SET : and với 111111111111111111111

2, CLEAR: ???

Chuyển đổi kiểu ( ép tự động )

Char ch;

Int i;//nếu vượt quá thì sẽ tự chuyển qua float

Float f;// 4 byte, 6 số sau số phảy

Double d; // 8 byte, độ chính sác sau số phảy là 10 số

Result = (ch/i) + (f\*d) + (f+i); kết quả cuối cùng sẽ là double

Ép kiểu ( tự ép )

Ví dụ :

float x,f;

f=3.14159

x = (int) f;

// kết quả x sẽ là 3

Độ ưu tiên của toán tử :

* Độ ưu tiên tạo lên cấu trúc phân cấp
* Thứ tự thực thi
* Độ ưu tiên có thể thay đổi

Một ngôi : --, ++, ….

Hai ngôi : \*,%,….

Độ ưu tiên của các toán tử so sánh : cái nào trước thực hiện trước, sau thực hiện sau.

Độ ưu tiên của các toán tử luận lý : thực hiện từ phải sang trái cái cùng 1 loại. not>> and >>….???

1 số học>> 2 so sánh >> 3 luận lý