# 墨烯C语言技术栈-C语言基础-013

操作符

C语言是非常灵活的

C语言提供了非常丰富的操作符 使用起来就比较灵活

算术操作符

+

加

-

减

\*

乘

/

除 得到是商 两端都是整数 执行整数除法 如果两端有一个是浮点数 执行浮点数除法 .1f输出的是一位小数 .2f输出两位小数

%

模 得到是余数 怎么说呢 3/2 余 1 得到的余就是1

移位操作符 涉及二进制的运算

>>

右移

<<

左移

位操作符

&

与

用于对两个整数的二进制表示进行逐位与运算

只有两个操作数的对应位都为1时 结果的对应位才为1 否则为0

简单来说就比如 59 二进制为 0011 1011

14二进制为 0000 1110

59 & 14就是 0000 1010 也就是 10

^

或

如果两个操作数的对应位相同(都是0或者都是1),则结果对应位为0 如果两个操作数对应位不同(一个是0 一个是1)则对应位为1

59 ^ 14 也就是 00110101 十进制也就是53

|

和

两个操作数对应位至少有一个是1则对应该位为1否则为对应位为0

59 | 14 0011 1111 十进制为63

赋值操作符

=

int num = 0;// 初始化

num = 20; //赋值

+=

num += 5;等价为num = num + 5;

-=

num -= 4;等价为num = num - 4;

\*=

num \*= 3;等价为num = num \* 3;

/=

num /= 2;等价为num = num / 2;

%=

num %= 1;等价为num = num % 1;

以下跟前者相同

&=

^=

|=

>>=

<<=

单目操作符

! 逻辑反操作

- 负值

+ 正值

& 取地址

sizeof 操作数的类型长度(以字节为单位)

~ 对一个数的二进制按位取反

-- 前置,后置--

++ 前置,后置++

\* 间接访问操作符(解引用操作符)

(类型) 强制类型转换