7.7-运算符优先级

运算符优先级决定了运算符用于值的顺序，在以下列出的优先级顺序中，第一组优先级最高，第二组次之，以此类推。

当两个运算符被应用于同一个操作数，则首先应用优先级高的运算符；若两个运算符的优先级相同，则按照**C++结合性**的规则来决定优先使用哪个运算符。

C++结合性：L-R表示同优先级下首先应用最左边的运算符；R-L...

C++运算符的优先级和结合性

第一组：

：： 作用域解析运算符

using namespace std; //包含命名空间std

std::cout<<...

第二组：L-R

() 运算中的（）、函数调用、值构造

[] 数组下标

-> 间接成员运算符

. 直接成员运算符（结构体和类访问成员变量或函数时用）

++ 后缀递增

-- 后缀递减 //优先级靠前，但运算法则使其靠后计算

第三组：R-L，都是一元运算符

！ 逻辑非

~ 按位取反

+ 正号

- 负号

++ 前缀递增

-- 后缀递增

& 取地址符

\* 解除引用（取内容）

() 强制类型转换

sizeof 计算字节长度

new 动态分配内存

delete 动态释放内存

第四组：L-R

.\* 成员解除引用

->\* 间接成员解除引用

第五组：L-R，都是二元运算符

\* 乘法

/ 除法

% 取模

第六组：L-R，都是二元运算符

+ 加法

- 减法

第七组：L-R

<< 左移

>> 右移

第八组：L-R

>

<

>=

<=

第九组：L-R

==

!=

第十组：L-R

& 按位与

第十一组：L-R

^ 按位异或

第十二组：L-R

| 按位或

第十三组：L-R

&& 逻辑与

第十四组：L-R

|| 逻辑或

第十五组：R-L

？ ： （三元）条件运算符

第十六组：R-L

= 赋值

\*= 乘并赋值

/= 除并赋值

%= 取模并赋值

+= 加并赋值

-= 减并赋值

&= 按位与并赋值

^= 按位异或并赋值

|= 按位或并赋值

<<= 左移并赋值

>>= 右移并赋值

第十七组：L-R

throw 异常处理

第十八组：L-R

, 表达式的值为逗号后的表达式的值

示例代码：

char ch1='A';

char ch2='b';

int num1=ch2-ch1;

char ch3=ch2-ch1;

cout<<num1<<endl;

cout<<ch3<<endl;

/\*

'A'的ASCII码为65，‘b’为98，98-65=33->这个ASCII码对应的字符是!

\*/

char ch4=ch1+ch3/10;

cout<<ch4<<endl;

/\*

33/10+65=68（小数部分被忽略），对应字符‘D’

\*/

double d1=0.5;

d1\*=num1-3;

cout<<d1<<endl;

/\*

减法优先级高于乘并赋值，所以先算33-3=30，然后0.5\*30=15赋值给d1

\*/

int a=1,b=2,c=3;

cout<<(b!=c && a>c || b==1)<<endl;

/\*

按照优先级和结合性计算：

a>c==0;

(b!=c)==1

(b==1)==0

0 || 0==0

1 && 0==0

\*/

a=++b+=c++;

cout<<a<<" "<<b<<" "<<c<<endl;

/\*

b=b+1; //b=3

b=b+c; //b=6;

c=c+1; //c=4

a=b; //a=6

\*/

总结：别这么写代码！别给自己找麻烦噢......也防止收作业的助教和老师心里一万个mmp

至此，运算符部分我们就已经全部讲完了，希望大家多多回顾巩固，这一段是非常基础的东西，大家一定要牢记

最后要三连