Lab6阅读并总结11-12章

第11章

1.运算符重载

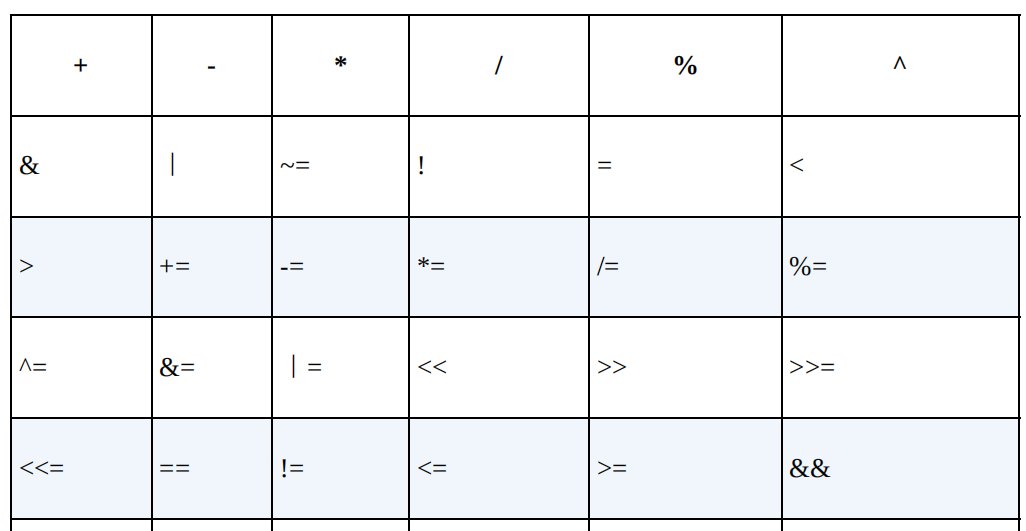
要重载运算符，需使用被称为运算符函数的特殊函数形式。格式为：

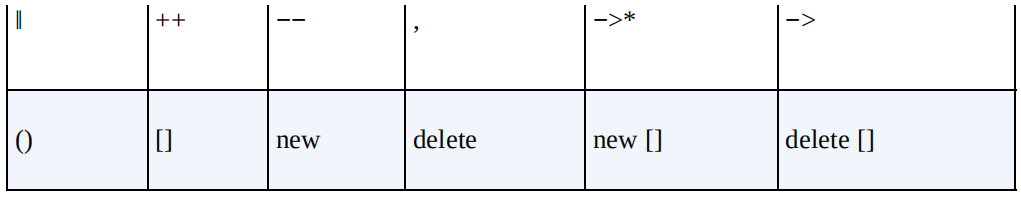
Operatorop(argument\_list) *argument-list表示该运算符的操作数。*

可用这种方法把district2=sid+sara;替换为：

District2=sid.operator+(sara)

1. 限制：重载后的运算符必须至少有一个操作数是用户定义的类型；使用运算符时不能违反运算符原来的句法规则（如：不能改变操作数的个数）；不能创建新运算符；可重载的运算符：





2.友元

友元是除公有类方法外访问私有对象的方法，有友元函数、友元类、友元成员函数三种。

1. 大多数运算符都可以通过成员或非成员函数重载，例如A=2.75\*B;与下面的非成员函数调用匹配：

A=operator\*(2.75,B);

1. 创建友元

友元函数是可以访问类的私有成员的非成员函数。

创建友元函数的第一步是将其原型放在类声明中，并在原型声明前加上关键字friend，例如：

friend Time operator\*(double m,const Time & t);

第二步是编写函数定义。因为它不是成员函数，所以不能使用限定符（如Time::）。

3.类的自动转换和强制类型转换

（1）知识引入

将一个标准类型变量的值赋给另一种标准类型的变量时，如果这两种类型兼容，则这个值转换为接收变量的类型，但这些转换可能会降低精度。

而不兼容的类型不会自动转换。此时可使用强制类型转换：

例如int \*p=(int \*)10;

（2）使用构造函数将double类型的值转换为Stonewt类型：

Stonewt（double lbs）;

注意只有接受一个参数的构造函数才能作为转换函数。

4.类转换函数。

（1）转换函数用于从类类型到某种类型的转换。转换函数是用户定义的强制类型转换，可以像使用强制类型转换那样使用它们。

要转换为typeName类型，需要使用这种形式的转换函数：operator typeName();

注意：转换函数必须是类方法、不能指定返回类型、不能有参数，必须返回转换后的值。

需要将函数原型添加到类声明中。

1. 类和动态的内存分配

C++使用new和delete运算符来动态控制内存。

1.动态内存和类

1. 与StringBad相应的构造函数原型

如：StringBad(const StringBad &)

1. 默认成员函数

在没有提供任何构造函数的情况下，C++默认构造函数，如：Klunk::Klunk(){ }

并且构造函数只能有一个。

1. 复制构造函数

用于将一个对象复制到新创建的对象中。也就是说， 它用于初始化过程中（包括按值传递参数），而不是常规的赋值过程中。类的复制构造函数原型：Class\_name(const Class\_\_name &);

新建一个对象并将其初始化为同类现有对象时，复制构造函数都将被调用。每当程序生成了对象副本时，编译器都将使用复制构造函数。

默认的复制构造函数逐个复制非静态成员（成员复制也称为浅复制），复制的是成员的值，例如：

语句StringBad sailor=sports;

等效于StringBad sailor; sailor.str=sports.str; sailor.len=sports.len;

2.在构造函数中使用new时的注意事项

（1）如果在构造函数中使用new，则应在析构函数中使用delete。

（2） new和delete必须相互兼容。new对应于delete，new[ ]对应于delete[ ]。 （3）如果有多个构造函数，则必须以相同的方式使用new，要么都带中括号，要么都不带。

（4）应定义一个复制构造函数，通过深度复制将一个对象初始化为另一个对象。

（5）应定义一个赋值运算符，通过深度复制将一个对象复制给另一个对象。

3.有关返回对象的说明

（1）返回指向const对象的引用

例如Vector Max(const Vector & v1,const Vecor & v2)

{

if(v1.magval()>v2.magval())

return v1;

else

return v2;

}

1. 返回对象

Vector Vector::operator+(const Vector & b) const

{

return Vector(x+b.x,y+b.y);

}

3.使用指向对象的指针

new：为创建的每一个对象的名称字符串分配存储空间，这是在构造函数中进行的；delete：析构函数使用它来释放这些内存。

（1）可以使用new来初始化指针：

String \*favorite=new String(sayings[choice]);

（2）使用->运算符通过指针访问类方法

if (sayings[i].length()<shortest->length())

（3）对对象指针应用解除引用运算符（\*）来获得对象：

if (sayings[i]<\*first)

first=&sayings[i];

4.复习

(1)转换函数

将单个值转换为类类型，需要创建的类构造函数原型如：

C\_name(type\_name value);

创建类成员函数如：

operator type\_name();