

**学生实验实习报告册**

|  |  |
| --- | --- |
| 学年学期： | 2016-2017 学年 🗹春🞏秋学期 |
| 课程名称： | 面向对象的程序设计 |
| 学生学院： | 经济管理学院 |
| 专业班级： | 信息管理与信息系统 |
| 学生学号： | 2016211002 |
| 学生姓名： | 钟建华 |
| 联系电话： | 17784458710 |

**重庆邮电大学教务处制**

|  |  |  |  |
| --- | --- | --- | --- |
| **课程名称** | **面向对象的程序设计** | **课程编号** |  |
| **实验地点** | **电子商务与网络开发实验室509** | **实验时间** | **2017/5/26** |
| **校外指导教师** | **无** | **校内指导教师** | **胡大权** |
| **实验名称** | **实验四 派生类与继承** | | |
| **评阅人签字** |  | **成绩** |  |

**一、实验目的**

1． 掌握运算符重载的基本方法。

2． 学习虚函数的定义与使用方法。

3． 理解静态多态性和动态多态性。

4． 学习使用虚函数和继承实现动态多态性。

**二、实验环境**

Visual C++ 6.0

**三、实验内容及步骤**

1． 重载运算符“\*”进行矩阵相乘，相乘的算法代码请参考实验 3 中的 Mul() 函数。这里给出一个示例，在矩阵相乘之前要检查两个矩阵是否可以相乘，之后根 据矩阵相乘的定义返回一个新的矩阵。 //---------------------------------------------------------------------------------------------------

Matrix

Matrix::operator\*(Matrix &m) { if(iMaxy!=m.iMaxx)//check the validity

{ cout<<"\nCannot multipiy Matrix a with b!\n";

return \*this; }

int ix,iy,iz;

Matrix c(iMaxx,m.iMaxy);

for(ix=0;ix<iMaxx;ix++)

for(iy=0;iy<m.iMaxy;iy++)

{ c.pMatrix[ix][iy]=0;

for(iz=0;iz<iMaxy;iz++) {

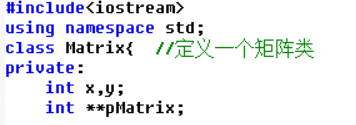
c.pMatrix[ix][iy]+=pMatrix[ix][iz]\*m.pMatrix[iz][iy]; }

}

return c;

}

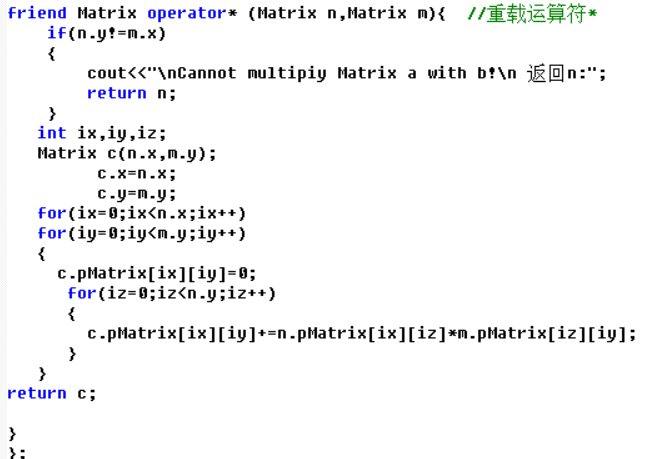
1）先定义一个矩阵，定义其私有部分的变量x,y和二级指针\*\*pMatrix,其中，x和y用于构造函数为数组赋值。



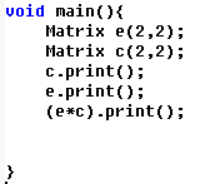
2）在公有部分定义其构造函数和析构函数，用构造函数为二级数组赋值，析构函数清理指针，后定义print函数输出数组。



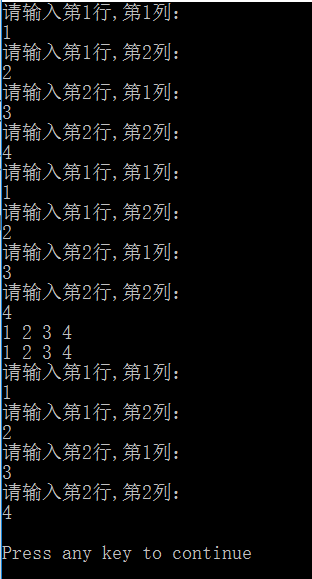
3）定义友元函数重载运算符\*，使其能够完成两个类的相乘。此处，在函数中用了三个for循环完成对两个矩阵的相乘。



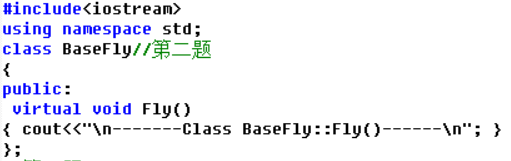
4）最后完成主函数。



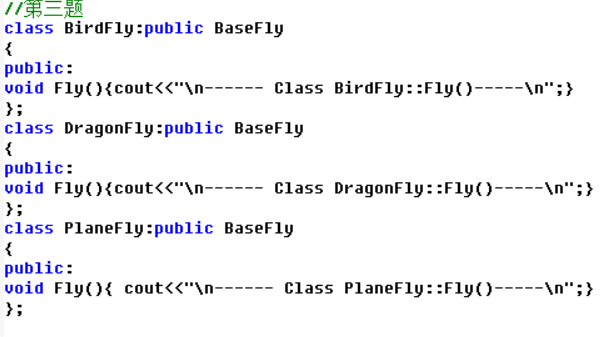
实验结果:



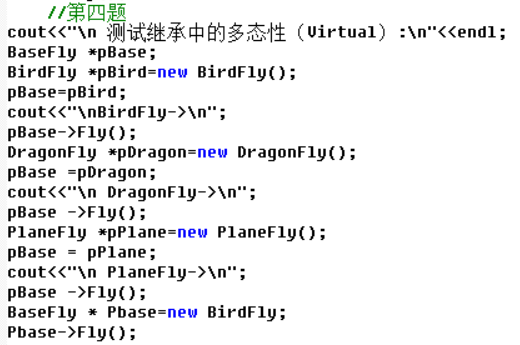
2． 下面提供一个体会继承中的多态性和虚函数在多态性中的作用的题目。请 根据提示进行实验。定义类 BaseFly()函数输出特定内容。



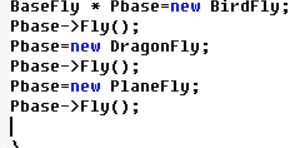
3． 定义类 BirdFly、DragonFly 和 PlaneFly，都继承自 BaseFly，重载 Fly()函 数使得各类中的 Fly()函数分别输出不同的内容。



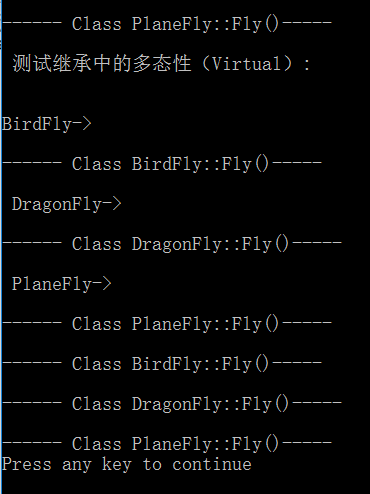
4． 在 main 函数中，用“new”关键字分配出以上四个类的实例，调用各个例 的 Fly()函数测试多态性。



5． 将 BaseFly::Fly()声明为 virtual，在 main（）中定义 BaseFly 的指针：\* Pbase， 依次分别指向 BirdFly、DragonFly 和 PlaneFly，并调用各类的 Fly()函数，体会虚函 数作用。



实验结果：



**四、实验心得及体会**

通过这次实验，使我明白了运算符重载的基本方法，虚函数定义方法，只有类的非静态成员函数才能定义为虚基类，同时学会了学习使用虚函数和继承实现动态多态性。虽然第一个实验没有完全成功，但是也是我对运算符重载有了更深刻的理解。