

南京邮电大学

实验报告

(2023 / 2024 学年 第 二 学期)

课程名称	面向对象程序设计及 C++
实验名称	实验二： 继承与派生实验
实验时间	2024 年 4 月 30 日
指导单位	计算机学院、软件学院、网络空间安全学院
指导教师	吴家皋

学生姓名	于明宏	班级学号	B23041011
学院(系)	计软网安学院	专 业	信息安全

实 验 报 告

实验名称	继承与派生实验		
实验类型	验证	实验学时	2
一、 实验目的和要求 <p>(1) 掌握单继承和多重继承下派生类的定义方法，理解基类成员在不同的继承方式下不同的访问属性。</p> <p>(2) 正确定义派生类的构造函数与析构函数，理解定义一个派生类对象时各个构造函数、析构函数被调用的顺序。</p> <p>(3) 正确定义虚基类，消除在多层次多重继承方式下顶层基类中成员访问的二义性问题，关注此时各构造函数、析构函数的调用顺序。</p> <p>(4) 通过基类与公有派生类的定义，及基类对象、指针、引用与派生类的对象、地址间相互赋值的方法，正确理解赋值兼容的 4 种情况，通过程序理解其不可逆性。</p>			
二、实验环境(实验设备) <p>硬件： 微型计算机</p> <p>软件： Windows 操作系统、Microsoft Visual Studio 2010</p>			
三、实验原理及内容 <p>(中文五号宋体，英文五号 Consolas 字体，单倍行距)</p> <p>实验题目 1: 定义一个车基类，派生出自行车类和汽车类，又以自行车类和汽车类为基类共同派生出摩托车类，每个类都要定义带有参数的构造函数。对自行车类继承车基类的方式分别用 private、protected、public，观察基类成员在派生类中的访问属性；观察自行车类、汽车类和摩托车类对象定义时构造、析构函数的调用顺序。最后将车基类定义为虚基类再观察程序运行结果。</p> <p>实验解答:</p> <p>① 根据提示进行填写完整实验指导 223 页代码对应位置内容如下：</p> <pre>(1)MaxSpeed = m; (2)Weight = w; (3)Vehicle(m, w) (4)Height = h; (5)Vehicle::Show();_</pre>			

实 验 报 告

```
(6)cout << "It\'s height is:" << Height << endl;
```

```
(7)b(15, 75, 60);
```

此时程序的运行结果是:

Constructing Vehicle...

Constructing Bicycle...

The vehicle is running!

Please stop running!

It's maxspeed is:15

It's weight is:75

It's height is:60

Destructing Bicycle...

Destructing Vehicle...

② 将继承方式改为 private 或 protected, 观察并分析程序的编译结果。

继承方式改为 protected:

函数 "Vehicle::Run"(已声明)不可访问

函数 "Vehicle::Stop"(已声明)不可访问

"Vehicle::Run"不可访问, 因为"Bicycle"使用"protected"从"Vehicle"继承

"Vehicle::Stop"不可访问, 因为"Bicycle"使用"protected"从"Vehicle"继承

继承方式改为 private:

函数 "Vehicle::Run"(已声明)不可访问

函数 "Vehicle::Stop"(已声明)不可访问

"Vehicle::Run"不可访问, 因为"Bicycle"使用"private"从"Vehicle"继承

"Vehicle::Stop"不可访问, 因为"Bicycle"使用"private"从"Vehicle"继承

③ 在 Bicycle 类下面增加 Car 类的定义, 参考实验教材给出的代码, 划线部分自己完成。

```
(8)Vehicle(m, w)
```

```
(9)SeatNum = s;
```

```
(10)Vehicle::Show();
```

```
(11)cout << "It\'s SeatNum is:" << SeatNum << endl;
```

```
(12)b(15, 75, 60);
```

实 验 报 告

(13)c(120, 1500, 100);_

- ④ 增加的第3层类 MotorCycle 及修改以后的 main()函数, 代码参见实验教材。

(14)public Bicycle, public Car

(15)mc(60, 240, 80, 2);

- ⑤ 将 Vehicle 声明为虚基类以消除二义性, 具体要在上面的基础上修改3个地方。

- 将 class Bicycle: public Vehicle 修改为 class Bicycle: virtual public Vehicle。

- 将 class Car: public Vehicle 修改为 class Car: virtual public Vehicle。

- 在第3层类的构造函数 MotorCycle(int m,int w,int h,int s): (16) 的初始化列表中增加对虚基类构造函数的调用。

(16)Bicycle(m, w, h),Car(m, w, s),Vehicle(m, w)

实验题目 2: 定义 Base 类及它的公有派生类 Derived 类, 两个类中均定义带参数的构造函数, 基类中定义函数 Show(), 派生类中也定义一个同名的 Show(), 二者输出内容有所区别。主函数中定义基类的对象、指针、引用, 也定义派生类的对象。

① 对赋值兼容的4种情况作测试, 对每行的输出结果进行观察, 理解赋值兼容何时调用基类的成员函数, 什么情况下才会调用派生类的成员函数。

② 在主函数的 return 0;语句前增加4条语句, 观察并记下编译时的报错信息, 理解赋值兼容的不可逆性。

实验解答:

- ① 按提示将程序填写完整, 代码参见实验教材, 对应位置内容是:

(1)b1(2);

(2)d1(5);

(3)b1 = d1;

(4)& b2 = d1;

(5)* b3 = &d1;

(6)new Derived(6);

程序的运行结果是:

基类对象 b1.show():

i in Base is:2

Constructing derived!

基类 b1=d1, b1.show():

i in Base is:5

派生类对象 d1.show():

实 验 报 告

```
i in Derived is:5
引用 b2=d1, b2.show():
i in Base is:5
基类指针 b3=&d1,b3->show():
i in Base is:5
Constructing derived!
基类指针 b4 = d4,b4->show( ):
i in Base is:6
派生类指针 d4, d4->show():
i in Derived is:6
```

② 在主函数的 return 0;语句前增加 4 条语句:

```
Derived d5=b1;
Derived &d6=b1;
Derived *d7=&b1;
d7=b3;
```

观察并记下编译时的报错信息，理解赋值兼容的不可逆性。

“初始化”：无法从“Base”转换为“Derived”

“初始化”：无法从“Base”转换为“Derived &”

“初始化”：无法从“Base*”转换为“Derived*”

“=”：无法从“Base*”转换为“Derived*”

四、实验小结（包括问题和解决方法、心得体会、意见与建议等）

（中文五号宋体，英文五号 Times new roman 字体，1.25 倍行距）

（一）实验中遇到的主要问题及解决方法

1.在题目（1）中将 Bicycle 继承 Vehicle 类的方式分别修改为 protected 和 private，再重新编译，请在小结中记录报错信息，解释原因。记录采取何种修改方式使程序正确运行？

继承方式改为 protected:

函数 "Vehicle::Run"(已声明)不可访问

函数 "Vehicle::Stop"(已声明)不可访问

"Vehicle::Run"不可访问，因为"Bicycle"使用"protected"从"Vehicle"继承

"Vehicle::Stop"不可访问，因为"Bicycle"使用"protected"从"Vehicle"继承

继承方式改为 private:

函数 "Vehicle::Run"(已声明)不可访问

函数 "Vehicle::Stop"(已声明)不可访问

"Vehicle::Run"不可访问，因为"Bicycle"使用"private"从"Vehicle"继承

"Vehicle::Stop"不可访问，因为"Bicycle"使用"private"从"Vehicle"继承

实 验 报 告

原因：采用“private”和“protected”继承，使得基类中可被继承的成员访问属性变为“private”或“protected”，而无法被访问。

修改方式：采取“public”继承方式可以使程序正确运行。

2. 在题目（2）中观察运行结果，总结在有赋值兼容的情况下，何时调用基类的成员函数，何时才会调用派生类的成员函数。

调用基类的成员函数：基类调用虚基类函数且对基类对象赋值；调用非虚基类函数。

调用派生类的成员函数：调用虚基类函数且对基类的对象引用和基类的对象指针进行赋值。

3. 其它问题及解决方法：

问题：定义派生类对象并生成新对象，忘记关键字 new 的用法，即给指针申请新空间。

解决办法：查找书籍，复习回顾 new 的用法。

（二）实验心得

在本次实验中，我深入学习了单继承和多重继承下派生类的定义方法，并且理解了不同继承方式下基类成员的访问属性。这让我对于继承的概念有了更清晰的认识。通过编写代码并观察不同继承方式下的访问权限，我进一步巩固了这些概念，并且理解了在派生类中如何正确使用基类的成员。

其次，我学习了如何正确定义派生类的构造函数与析构函数，并且理解了在创建派生类对象时各个构造函数、析构函数被调用的顺序。这使我能够更好地控制对象的生命周期，并且有效地管理资源。通过深入理解构造函数与析构函数的调用顺序，我们、能够编写出更加健壮、可靠的程序。

在学习虚基类的定义过程中，我解决了多层次多重继承方式下顶层基类中成员访问的二义性问题，并且关注了此时各构造函数、析构函数的调用顺序。这让我更深入地理解了多重继承的复杂性，并且学会了如何使用虚基类消除二义性问题，确保程序的正确性和可维护性。

综上所述，通过本次实验，我不仅掌握了继承的基本概念和语法，还深入理解了继承在面向对象编程中的重要性。同时，通过编写代码并观察程序运行结果，我对于继承和派生类的概念有了更深入的理解，并且掌握了正确使用这些概念的方法。这将对我今后的编程工作和学习都有很大的帮助。

（三）意见与建议（没有可省略）

可以提供更多的时间上机操作，以确保更多程序设计思路得以实现，提升面向对象语言的掌握程度和编程能力。

实 验 报 告

五、支撑毕业要求指标点

信息安全：

1.2-M 掌握计算机软硬件相关工程基础知识，能将其用于分析信息安全领域的相关工程问题。

3.1-H 掌握信息安全领域所涉及的软硬件系统，从数字电路、计算机系统、到各类系统软件的基本理论与设计结构。

六、指导教师评语

评 分 细 则	评分项	优秀	良好	中等	合格	不合格
	遵守实验室规章制度					
	学习态度					
	算法思想准备情况					
	程序设计能力					
	解决问题能力					
	算法设计合理性					
	算法效能评价					
	报告书写认真程度					
	内容详实程度					
	文字表达熟练程度					
	其它评价意见					
	本次实验能力达成评价 (总成绩)		批阅人		日期	