C++(下) 实验 继承(二)和多态

**实验目的：**熟悉并掌握继承的技术，能够利用公有继承方式建立符合用户需求的类族。

**实验要求：**按照每个类两个文件的方式（一个头文件，一个源文件）组织工程内的代码。

**实验内容：**

1. **代码调试: 附件中的代码在建立类族的过程中，由于编程人员的疏忽，出现了一些小问题，请帮忙修改过来。**

**2. 设计交通工具类族: 开发一个名为Vehicle 的类的层次体系。创建两个类Taxi 和Truck，均以公有模式从类Vehicle 中继承而来。Taxi 类中应包含一个数据成员passenger说明其是否载客。Truck类应包含一个数据成员cargo说明其是否载货。根据题后附的测试程序输出为类Vehicle添加必要的数据成员,并为所有类添加必要的函数来控制和访问类的数据。编写一段测试程序，将Vehicle对象、Truck 对象和Taxi对象打印到屏幕。**

测试程序输出实例为：

Vehicle

Number of doors: 2

Number of cylinders: 6

Transmission type: 3

Color: blue

Fuel level: 14.6

Taxi

Number of doors: 4

Number of cylinders: 6

Transmission type: 5

Color: yellow

Fuel level: 3.3

The taxi has no passengers.

Truck

Number of doors: 2

Number of cylinders: 16

Transmission type: 8

Color: black

Fuel level: 7.54

The truck is carrying cargo.

**2.多态-虚函数和抽象类**

**实验内容: 重现ppt 中（详见附件）的继承层次: Point/Circle/Cylinder ,**

**建立下列主函数测试这个继承模型。**

**int main()**

**{**

**Point point(3.5,6.4), \*p;**

**Circle circle(4,5,6), \*cir;**

**Cylinder cyclinder(5,6,8), \*cyl;**

**cout<<point;**

**cout<<circle;**

**cout<<cyclinder;**

**p=&point;**

**cir=&circle;**

**cyl=&cyclinder;**

**cout<<(\*p);**

**cout<<(\*cir);**

**cout<<(\*cyl);**

**//+++++++++++++++++++**

**cir->area();**

**cir=&cyclinder;**

**cir->area();**

**//+++++++++++++++++++**

**return 0;**

**}**

1. **整合程序，使主函数可以运行；运行结果是什么？**

**[3.5,6.4]**

**[4,5], r=6**

**[5,6], r=8, h=0**

**[3.5,6.4]**

**[4,5], r=6**

**[5,6], r=8, h=0**

1. **为Circle类的area()成员声明添加virtual, 再运行主函数，运行结果是什么？**

**[3.5,6.4]**

**[4,5], r=6**

**[5,6], r=8, h=0**

**[3.5,6.4]**

**[4,5], r=6**

**[5,6], r=8, h=0**

1. **为Point类添加area()成员:**

**float area() const { return 0;}**

**为主函数添加下列代码，可以执行**

**p=&circle;**

**p->area();**

**p=&cyclinder;**

**p->area();**

**point &p1=circle;**

**p->area();**

**point &p2=cyclinder;**

**p2->area();**

**4) 在Point类添加的area()成员前加virtual**

**再运行主函数，运行结果是什么？**

**[3.5,6.4]**

**[4,5], r=6**

**[5,6], r=8, h=0**

**[3.5,6.4]**

**[4,5], r=6**

**[5,6], r=8, h=0**

**113.097**

**402.124**

**113.097**

**402.124**

**113.097**

**402.124**

**5）增加一个抽象类Shape**

**class shape{**

**public:**

**virtual float area()=0;**

**};**

**让Point成为Shape的子类。看看对主函数有什么影响？**

**无影响**

**6）将Point中的area（）成员删除，看看有什么影响？**

**无影响**