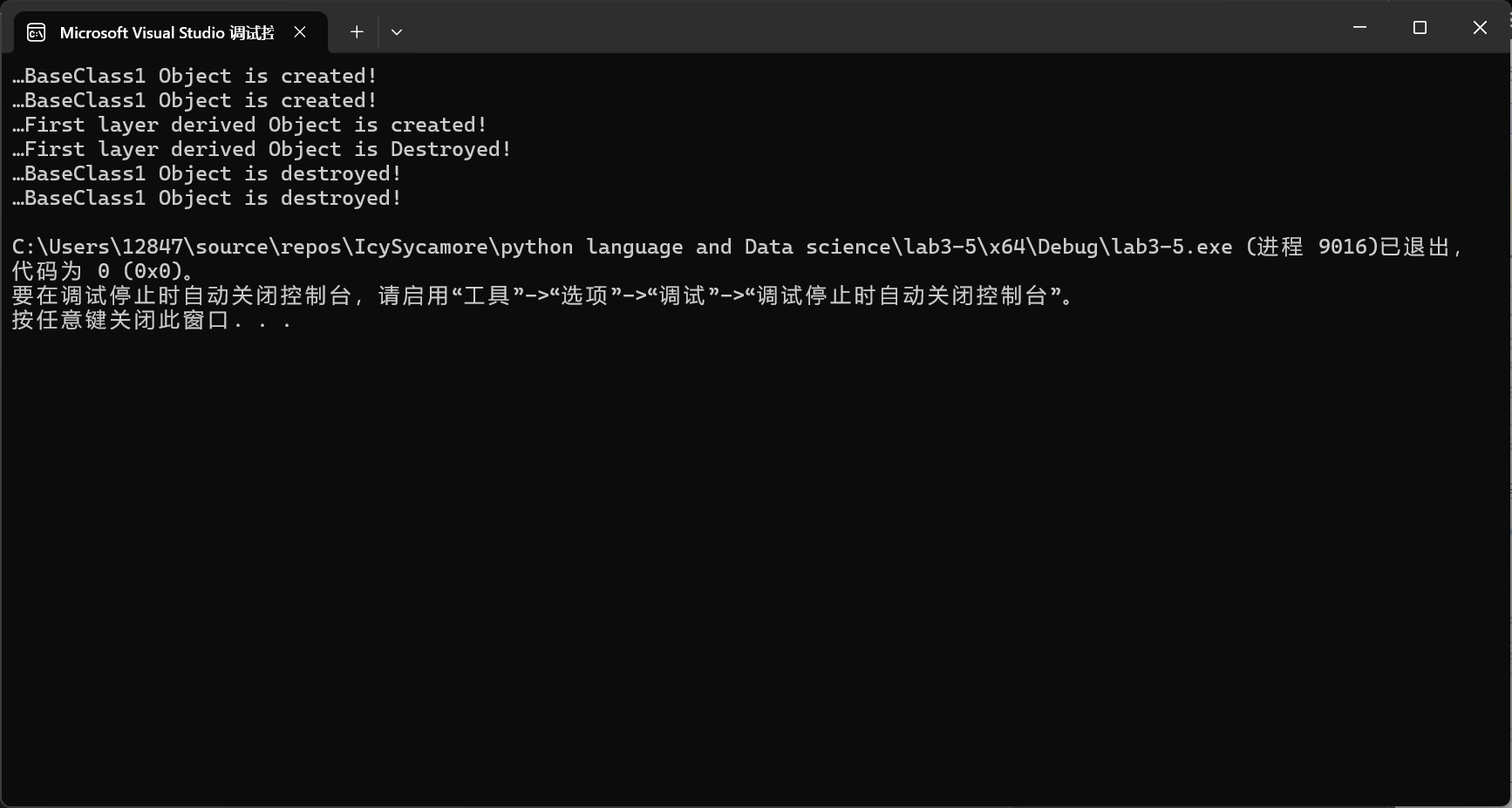
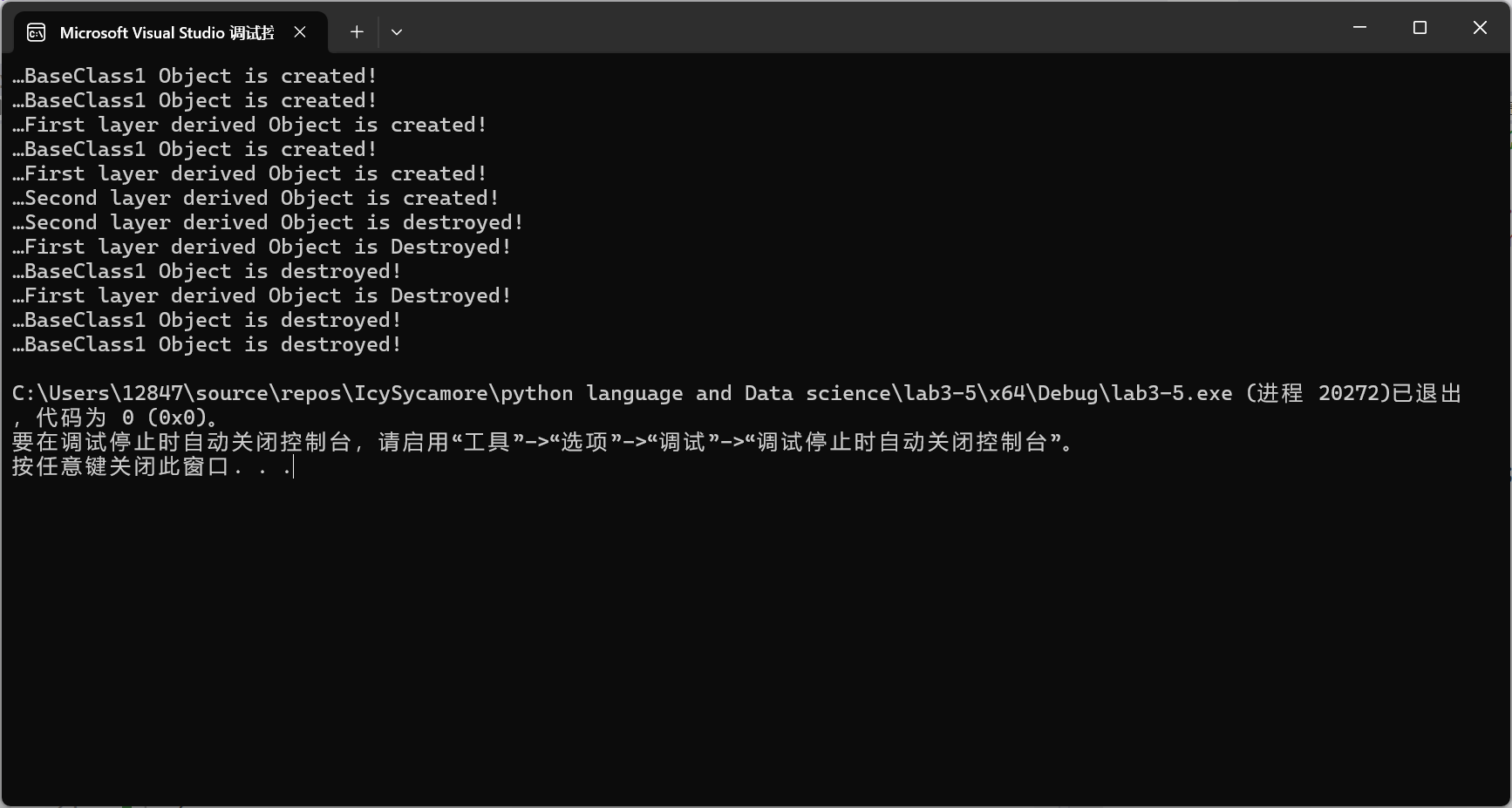
所有运行结果为更正后的

Lab3.1



|  |
| --- |
| 错误 |
| 未使用命名空间std |
| end拼写错误—>endl |
| 类后括号缺少; |

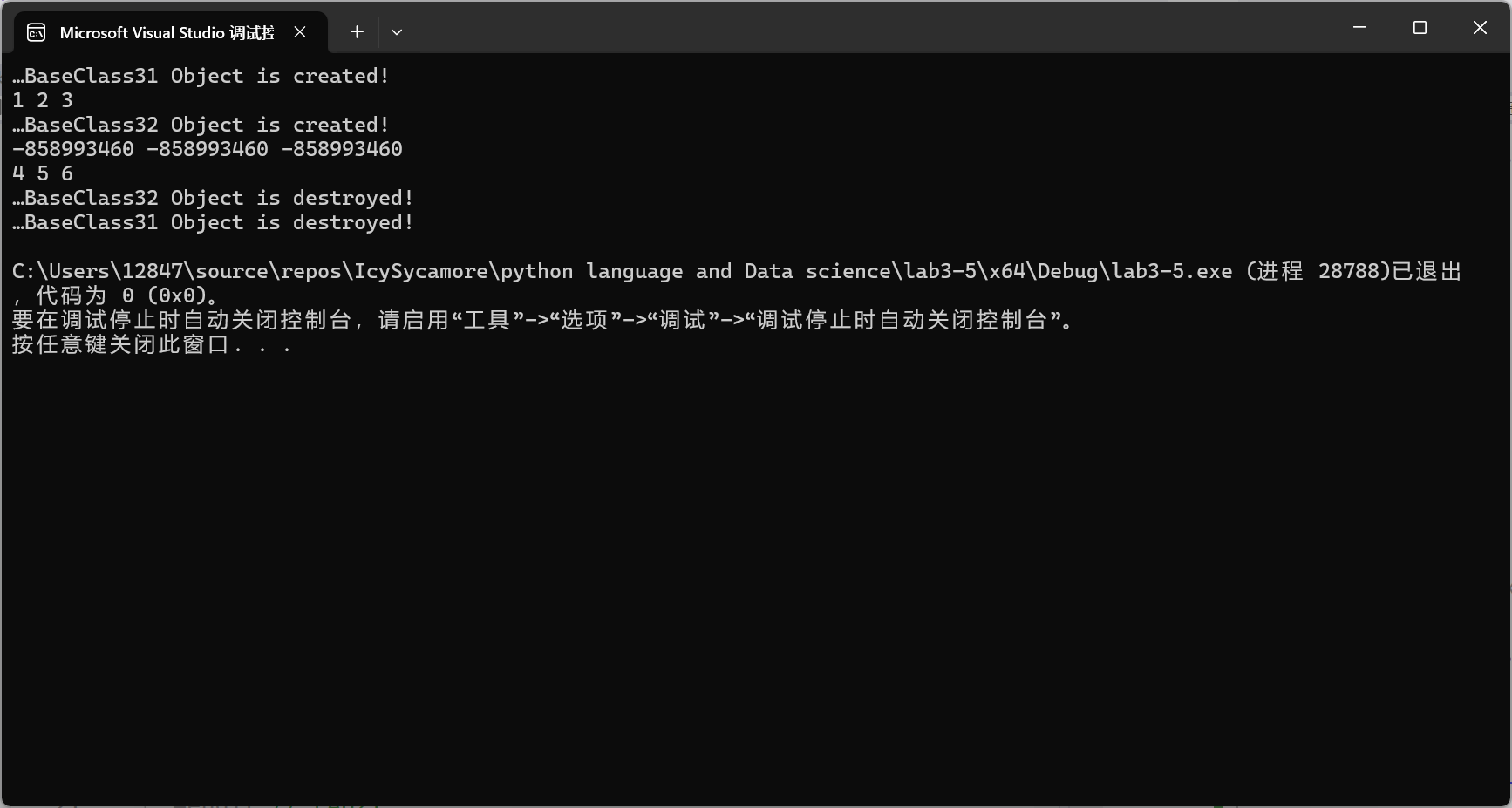
|  |  |
| --- | --- |
| 行数 | 输出分析 |
| 1 | MyBase1实例a先调用Mybase构造函数 |
| 2 | 作为MyBase派生类，MyDerived1实例b调用了MyBase的构造函数 |
| 3 | b自己MyDerived1类的构造函数 |
| 4 | b自己MyDerived1类的构造函数 |
| 5 | 作为MyBase派生类，MyDerived1实例b调用了MyBase的析构函数 |
| 6 | MyBase1实例a调用Mybase析构函数 |



|  |  |
| --- | --- |
| 1 | MyBase1实例a先调用Mybase构造函数 |
| 2 | 作为MyBase派生类，MyDerived1实例b调用了MyBase的构造函数 |
| 3 | b自己MyDerived1类的构造函数 |
| 4 | 作为MyBase派生类，MyDerived11实例c调用了MyBase的构造函数 |
| 5 | 作为MyDerived1派生类，MyDerived11实例c调用了MyDerived1的构造函数 |
| 6 | C调用了自己的构造函数 |
| 7 | C调用了自己的析构函数 |
| 8 | 作为MyDerived1派生类，MyDerived11实例c调用了MyDerived1的析构函数 |
| ⑨ | 作为MyBase派生类，MyDerived11实例c调用了MyBase的析构函数 |
| 10 | b自己MyDerived1类的构造函数 |
| 11 | 作为MyBase派生类，MyDerived1实例b调用了MyBase的构造函数 |
| 112 | MyBase1实例a调用Mybase析构函数 |

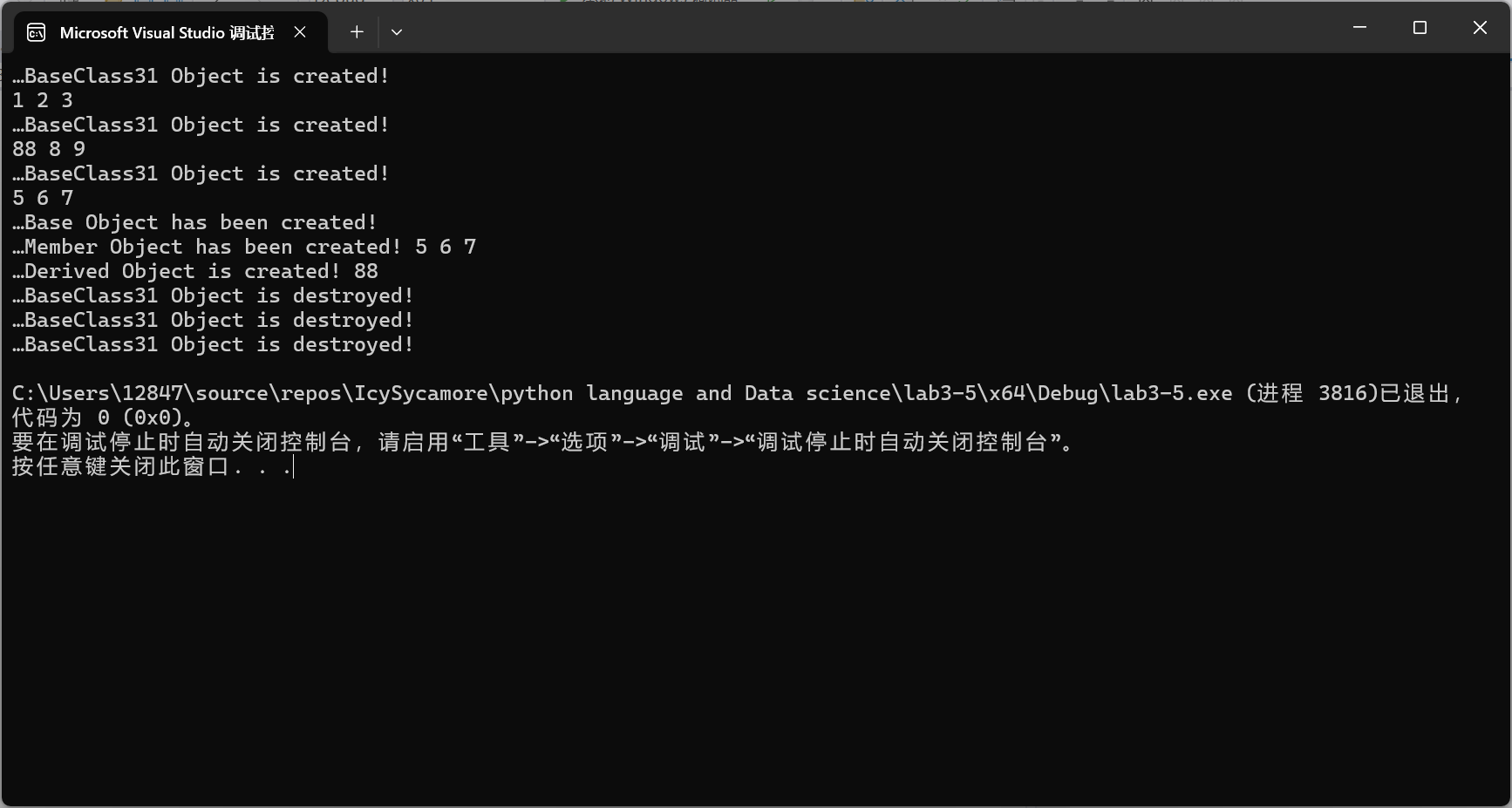
分析：基类与派生类遵循FILO规则，且一个派生类按照其继承层次依次调用构造和析构函数

# Lab3.2



|  |  |
| --- | --- |
| 行数 | 输出分析 |
| 1 | a构造,构造函数输出 |
| 2 |
| 3 | b构造 |
| 4 | 未赋值的abc输出 |
| 5 | 赋值后abc输出 |
| 6 | b析构 |
| 7 | a析构 |

|  |
| --- |
| 错误 |
| end->endl |
| 未使用命名空间std |
| 未定义的行为：在赋值前访问abc |
| 类后括号缺少; |



|  |  |
| --- | --- |
| 行数 | 输出分析 |
| 1 | a 构造+输出 |
| 2 |
| 3 | b的基类MyBase31构造函数+输出 |
| 4 |
| 5 | b内组合的MyBase31类对象a构造+输出 |
| 6 |
| 7 | b自身构造函数输出 |
| 8 |
| 9 |
| 10 | b自身析构 |
| 11 | b内a析构 |
| 12 | a析构 |

|  |
| --- |
| 错误 |
| 未添加访问控制符，默认为private数据成员 |
| MyBase31 a(5,6,7);被解释为返回值为一个MyBase31类对象，参数为定值（即使是创建一个函数也不能使用定值）的函数a而未按预期的创建一个MyBase31对象a->MyBase31 a{5,6,7}或在构造函数中赋值 |
| 对象a不含xyz数据成员，abc作为组合的MyBase31的private成员也不能直接访问->创建getA()等 或 将abc的访问权限更改为public |
| 类后括号缺少; |

C++规定对象构造顺序为 “从基类到派生类”。派生类构造函数必须显式或隐式调用基类构造函数，否则编译器会尝试调用基类的默认构造函数。若基类没有默认构造函数，则会导致编译错误。此外，*派生类对象在内存中****包含完整的基类子对象****，即派生类对象不仅拥有自身成员，还包含基类的所有成员（包括数据成员和函数成员）*

# Lab3.3

|  |
| --- |
| Fun1-3函数缺少返回值 |
| 类后括号缺少; |
| 标识符protected拼写错误 |
| 构造函数名称与类名不同 |
| .h 使用中文引号 |
| main.cpp 使用中文符号 |

|  |  |  |  |
| --- | --- | --- | --- |
| **继承方法 成员类型** | **public** | **protected** | **private** |
| public | public | protected | private |
| protected | protected | protected | private |
| private | private | private | private |

继承后的权限 = 基类成员权限与继承方式权限的等级取高（private > protected > public）

## Lab3.3.5

派生类与基类中成员名称相同时

直接使用该名称调用的是派生类成员

调用基类成员需要用一元作用域运算符：：

按照has a顺序依次,如Base::Dervied1::Derived2::Derived3::func()

# Lab3.4

|  |
| --- |
| 成员函数没有（空）函数体 |
| 使用中文引号 |
| Main.cpp中\*p语句后多余的逗号 |
| 返回成员y时大小写错误 |
| 声明并初始化指针时使用MyBase对象而非MyBase对象的地址 |
| 声明并初始化指针时使用MyDerived对象而非MyDerived对象的地址 |
| a=b 和 p=q切片后失去基类成员和方法 |
| 试图违反赋值单向性b=a |