心得总结（第四章）

本章的习题内容主要围绕派生类展开，对于派生类，通过这些习题的练习使我有了更加深刻的认知：类的 派生 和 继承 是 面向对象程序设计方法 和C++语言最 重要的特征 之一。 因为客观世界本身是有层次的，在人们认识客观世界的过程中，由 一般到特殊的演绎思维 发挥着巨大作用。 演绎的过程 在绝大多数情况下表现为 层次分类 的过程。 继承使得程序员可以在一个较一般的类的基础上很快地建立一个新类，而不必从零开始设计每个类。 由此可见，从一个或多个以前定义的类（基类）产生新类的过程称为“派生”，这个新类称为派生类。 派生的新类 同时也 可以增加或重新定义数据和操作 ，这就 产生了类的层次性 。 派生的新类同时也可以增加或重新定义数据和操作，这就产生了类的层次性。 派生的新类同时也可以增加或重新定义数据和操作，这就产生了类的层次性。 第二种称为 性质扩展 ，即 增加派生类的性质 。而派生类的继承方式有三种：公有继承、私有继承以及保护继承

其具体限制有如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 私有继承（private） | | | |
| 基类中的成员 | 私有成员 | 公有成员 | 保护成员 |
| 内部访问 | 不可访问 | 可访问 | 可访问 |
| 对象访问 | 不可访问 | 不可访问 | 不可访问 |

|  |  |  |  |
| --- | --- | --- | --- |
| 公有继承（public） | | | |
| 基类中的成员 | 私有成员 | 公有成员 | 保护成员 |
| 内部访问 | 不可访问 | 可访问 | 可访问 |
| 对象访问 | 不可访问 | 可访问 | 不可访问 |

|  |  |  |  |
| --- | --- | --- | --- |
| 保护继承（protected） | | | |
| 基类中的成员 | 私有成员 | 公有成员 | 保护成员 |
| 内部访问 | 不可访问 | 可访问 | 可访问 |
| 对象访问 | 不可访问 | 不可访问 | 不可访问 |

通过熟练运用各种继承方式能够使派生类方便继承的同时保护基类的数据安全。