感想心得

1. 本章主要的学习内容是派生类与继承，继承也是c++三大特性之一，是学习中的重点。
2. 继承方式分为三种，有公有继承、私有继承和保护继承。三种继承对基类成员的访问权限不同，在考试中会给出相关表格，掌握这方面的内容。
3. 使用派生类可以提高代码的可重用性。
4. 当一个派生类具有多个基类时，称为多继承。在多继承函数构造函数定义中，几个基类构造函数用（，）分隔。
5. 构造函数和析构函数的执行顺序很重要，在多继承中，先执行父类的构造函数，然后执行子类的构造函数，析构函数与构造函数的顺序相反。
6. 虚基类是为了解决二义性。

4.1、 公有继承,私有继承和保护继承。

         基类的私有成员，无论哪种继承方式都不能访问。

         公有继承不改变基类的公有和保护成员的访问限制。

         私有继承将基类的公有和保护成员都变成私有。

         保护继承将基类的公有和保护成员都变成保护。

4.2、不能。可以在基类里添加一个公有成员函数来访问私有成员，派生类就能继承这个公有成员函数，实现对基类私有成员的访问。

4.3、当类的继承方式为公有继承时 .基类中的所有保护成员在派生类中仍以保护成员的身份出现，在派生类内可以访问这些成员，但派生类外部不能访问它们，而在下一层派生类内可以访问它们。  
当类的继承方式为私有继承时，基类中的所有保护成员在派生类中都以私有成员的身份出现在派生类内可以访问这些成员.但派生类外部不能访问它们。

4.4、构造函数是先执行父类的构造函数，再执行类中其他类对象的构造函数，再执行本类的构造函数。如果同级有多个构造函数，则按声明顺序执行。

       析构函数与构造函数的执行顺序刚好相反。

4.5、派生类名（参数总表） ： 基类名（参数总表）

         {

                   派生类新增数据成员的初始化语句

　　　}

派生类中的析构函数与基类无关。

4.6、当一个派生类具有多个基类时，这种派生方法称为多继承。  
多重继承的构造函数的执行顺序与单继承构造函数的执行顺序相同，也是遵循先执行基类的构造函数，再执行对象成员的构造函数,最后执行派生类构造函数体的原则。处于同一层次的各个基类构造函数的执行顺序.取决于声明派生类时所指定的各个基类的顺序，与派生类构造函数中所定义的成员初始化列表的各项顺序没有关系。析构函数的执行顺序则刚好与构造函数的执行顺序相反。

4.7、如果一个类有多个直接基类，而这些直接基类又有一个共同的基类，则在最底层的派生类中会保留这个简介的共同基类数据成员的多份同名成员。在访问这些同名成员的时候，会产生二义性。为了解决二义性，引入了虚基类。

1)         如果虚基类中定义有带形参的构造函数，并且没有定义默认形式的构造函数，则整个继承结构中，所有直接或间接的派生类都必须在构造函数的成员初始化表中列出对虚基类构造函数的调用，以初始化在虚基类中定义的数据成员。

2)         建立一个对象时，如果这个对象中含有从虚基类继承来的成员，则虚基类的成员是由最远派生类的构造函数通过调用虚基类的构造函数进行初始化的。该派生类的其他基类对虚基类构造函数的调用都自动被忽略。

3)         若同一层次中同时包含虚基类和非虚基类，应先调用虚基类的构造函数，再调用非虚基类的构造函数，最后调用派生类的构造函数。

4.8、A

4.9、C

4.10、C

4.12、Constructor B1.

         Constructor B2.

         Constructor A.

         3

         2

         1