阅读内容：阅读C++ prime plus 第13 章并总结关键内容

总结：

1. 面向对象编程的主要目的之一是提供可重用的代码。必须考虑的细节越少，越能专注于程序的整体策略。
2. 类继承：从已有的类派生出新的类，而派生类继承了原有类（基类）的特征，包括方法。
3. 继承可以完成的工作：

* 可以在已有类的基础上添加功能；
* 可以给类添加数据；
* 可以修改类方法的行为。

1. 使用公有派生，基类的公有成员将成为派生类的公有成员，基类的私有部分也将成为派生类的一部分，但只能通过基类的公有和保护方法访问。
2. 派生类需要自己的构造函数，可以根据需要添加额外的数据成员和成员函数。构造函数必须给新成员和继承的成员提供数据。
3. 创建派生类对象时，程序首先创建基类对象。从概念上说，这意味着基类对象应当在程序进入派生类构造之前被创建。
4. 如果不调用基类构造函数，程序将使用默认的基类构造函数。除非要使用默认构造函数，否则应显式调用正确的基类构造函数。
5. 释放对象的顺序与创建对象的顺序相反，即首先执行派生类的析构函数，然后自动调用基类的析构函数。
6. 基类指针可以在不进行显式类型转换的情况下指向派生类对象；基类引用可以在不进行显式类型转换的情况下引用派生类对象。
7. 公有继承最常用，建立一种is-a关系。
8. 使用了virtual关键字的方法被称为虚方法。关键字virtual只用于类声明的方法原型中。
9. 编译器处理虚函数的方法是：给每个对象添加一个隐藏成员。隐藏成员中保存了一个指向函数地址数组的指针。这种数组称为虚函数表。
10. 关键字protected与private相似，在类外只能用公有类成员来访问protected部分中的类成员。不同的是，对于外部世界来说，保护成员的行为与私有成员相似；但对于派生类来说，保护成员的行为和公有成员相似。