### 心得总结（第三章）

我认为第三章是C++中最重点的一章，它也是程序运算编译中必不可少的一部分，本章的习题也大多数是有关类、构造、析构函数设计的。通过这些课后题目，使我对类和对象的概念，构造函数与析构函数的作用有了更加清楚的认知。

C++提供了构造函数来处理对象的初始化。构造函数的特征：  
1、构造函数是一种特殊的成员函数，与其他成员函数不同，不需要用户来调用它，而是在建立对象时自动执行。  
2、构造函数的名字必须与类名同名，而不能由用户  
任意命名，以便编译系统能识别它并把它作为构造函数处理。  
3、它不具有任何类型，不返回任何值。

有关构造函数的使用，有以下说明：  
（1）在类对象进入其作用域时调用构造函数。  
（2）构造函数没有返回值，因此也不需要在定义构造函数时声明类型，这是它和一般函数的一个重要的不同之点。  
（3）构造函数不需用户调用，也不能被用户调用。  
（4）在构造函数的函数体中不仅可以对数据成员赋初值，而且可以包含其他语句。但是一般不提倡在构  
造函数中加入与初始化无关的内容，以保持程序的清晰。  
（5）如果用户自己没有定义构造函数，则C++系统会自动生成一个构造函数，只是这个构造函数的函数体是空的，也没有参数，不执行初始化操作。

析构函数也是一个特殊的成员函数，它的作用与构造函数相反，它的名字是类名的前面加一个“～”符号。在C++中“～”是位取反运算符，从这  
点也可以想到：析构函数是与构造函数作用相反的函数。以下情况出现时，会自动执行析构函数  
①如果在一个函数中定义了一个对象（它是自动局部对象），当这个函数被调用结束时，对象应该释放，在对象释放前自动执行析构函数。  
②static局部对象在函数调用结束时对象并不释放，因此也不调用析构函数，只在main函数结束或调用exit函数结束程序时，才调用static局部对象的析构函数。

③如果定义了一个全局对象，则在程序的流程离开其作用域时（如main函数结束或调用exit函数）时，调用该全局对象的析构函数。  
④如果用new运算符动态地建立了一个对象，当用delete运算符释放该对象时，先调用该对象的析构函数。析构函数的作用并不是删除对象，而是在撤销对象占用的内存之前完成一些清理工作，使这部分内存可以被程序分配给新对象使用。

C++类成员的三种访问权限：  
·public：可以被该类中的函数、子类的函数、友元函数访问，也可以由该类的对象访问；  
·protected：可以被该类中的函数、子类的函数、友元函数访问，但不可以由该类的对象访问；  
·private：可以被该类中的函数、友元函数访问，但不可以由子类的函数、该类的对象、访问。  
但是，有时需要定义一些函数，这些函数不是类的一部分，但又需要频繁访问类的私有（private）成员和保护（protected）成员，这时可以将这些函数定义为友元函数。除了友元函数，还有友元类，两者统称为友元(friend) 。借助友元，可以使得普通函数或其他类中的成员函数可以访问某个类的私有成员和保护成员。