

# 第二次周报

袁一清 U202114974\*

## 1 C++部分

事实上第一次周报中有部分第二周学习的一些内容，所以此次周报的内容将会略显单薄，但为了不过于寒酸，我将写的详细一些。

学习《C++PrimerPlus》第十二章：

1、类声明为 char 指针，在构造函数中使用 new 分配空间，类便可以处理长度不同的字符串。

2、无论创建多少对象，程序仅创建一个静态类变量副本。不能在类声明中初始化静态成员变量，声明并不分配内存，但 const 整数类型或枚举型可在类声明中初始化。静态类成员单独存储，并不为对象的组成部分。

3、new 分配内存时必须在析构函数中使用 delete 释放，且[]得对应。

4、为避免产生二义性，仅能有一个默认构造函数。复制构造函数用于初始化过程，按值传递和返回对象时都将调用复制构造函数，按引用传递对象可以节省调用构造函数的时间以及存储新对象的空间（真不懂，太抽象了）。必须定义复制构造函数在于一些类是使用 new 初始化的指向数据的指针，所以需要进行深度复制。

5、添加赋值运算符也不是怎么懂，不是已经有复制构造函数了吗，那么这两个是配合一起用的吗，还是说各自解决各自的问题。赋值运算符实现顺序：

i、检查自我赋值情况 ii、释放成员指针以前内存 iii、复制数据并返回一个指向调用对象的引用（iii 不怎么懂，我发现一旦较深的谈到引用我就不行了）。

6、delete[]与使用 new[]初始化的指针和空指针都兼容（nullptr 用于空指针）。

7、静态类成员函数（不知所云）：函数声明必须包括 `static`，但若函数定义独立，则其中不能包括 `static` (?)，不能通过对象调用静态类且静态类成员甚至不能用 `this` 指针，若静态类成员是在公有部分声明的，可使用类名和作用域解析运算符调用(?)。

8、若有多个构造函数，必须以相同方式使用 `new`，因为仅一个析构函数。

9、若方法或函数要返回局部对象，则应返回对象，而不是指向对象的引用；若方法或函数要返回一个没有公有复制构造函数的类，必须返回一个指向这种对象的引用（不懂）；引用效率高。

10、定位 `new` 运算符能在分配内存时指定内存位置，且使用时必须确保析构函数被调用（显示地使用），与创建顺序相反的顺序进行删除。

## 2 总结

本章未进行练习，先把十三章学完吧，太生疏了，理解的太浅薄了。