

## Lesson 10

1. 拷贝构造函数不能为 `A(A a)` 的形式，因为 `A a` 这种形参能够触发拷贝语义（copy semantics），而拷贝构造函数本身是为了解决拷贝语义而存在的，所以它自身要避免拷贝。
2. 一个空类，编译器自动合成默认无参构造函数、析构函数、拷贝构造函数、赋值运算符。
3. 在编写类的时候，必须严格区分对象是否可复制。
4. 运算符重载可以有成员函数和友元函数的形式，后者比前者多一个参数。
5. `=` 和 `+=` 必须重载为成员函数的形式。
6. 输入和输出必须为友元函数的形式。而且输入操作符要考虑输入失败的情况。
7. 运算符重载为成员函数的形式，那么该操作符的第一个操作数必然为该类的对象。以 `+=` 为例，`s += "hello"` 那么等价于 `s.operator+=(“hello”)`
8. `Integer(int data)` 类型的构造函数，具有一种转化语义，能够将 `int` 转化为 `Integer`，而加上 `explicit` 就禁用掉了转化语义。加上该关键字后，只能采用原生的构造方式如 `Integer t(100)`，而不能使用转化形式 `Integer t = 33;`
9. 目前碰到的程序语义：
  - a) `const`: 常量语义、保护语义

b) 非 const: 修改语义

c) pass-by-value: 复制语义

d) A(int a)类型构造函数: 转化语义

10.发生异常跳出 try 的时候, 栈内存是一定会被回收的, 堆内存无法被正确回收。

11.智能指针利用了栈对象的生存期, 将资源的获取放在构造函数里面, 资源的释放放在析构函数里面, 从而保证了资源一定会被正确释放。这就是 C++中的 RAII 技术, (Resource Acquirement Is Initialization)。

12.自增操作符, 前置和后置的区别:

a) 前置直接修改原对象, 直接返回。

b) 后置需要暂存之前的结果, 修改对象后, 将旧的对象返回。

13.作业: 编写一个 Complex 类, 实现以下功能:

a) 默认构造函数

b) 带参数的构造函数

c) +、-、\*、/

d) <<、>>

e) 求模的大小