

C 空类的默认成员函数总结计算机等级考试 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/625/2021_2022_C___E7_A9_BA_E7_B1_BB_E7_c97_625221.htm

编辑特别推荐: 全国计算机等级考试 (等考) 指定教材 全国计算机等级考试学习视频 全国计算机等级考试网上辅导招生 全国计算机等级考试时间及科目预告 百考试题教育全国计算机等级考试在线测试平台 全国计算机等级考试资料下载 全国计算机等级考试论坛

```
class Empty { public: Empty(). // 缺省构造函数 Empty(const Empty&. operator=(const Empty&(). // 取值运算符 const Empty* operator&. ) ;
```

当我们使用一个对象来初始化另一个对象时，编译器将自动生成上述构造函数（称为复制构造函数，因为它创建对象的一个副本）。现在我们来看看我们没有定义复制构造函数的情况下调用隐式复制构造函数将会出现什么情况。从构造函数定义的代码片断可以看到，当中使用new操作符初始化了一个指针str，而隐式的复制构造函数是按值进行复制的，那么对于指针str，将会进行如下复制：
sailor.str = sports.str；这里复制的不是字符串，而是一个指向字符串的指针！也就是说，我们将得到两个指向同一个字符串的指针！由此会产生问题将不言而喻。当其中一个对象调用了析构函数之后，其str指向的内存将被释放，这个时候我们如果调用另一个对象，其str指向的地址数据会是什么？很明显将会出现不可预料的结果。所以由此可见，如果类中包含了使用new初始化的指针成员，应当定义一个复制构造函数，以复制指向的数据，而不是指针，这被称为深度复制。因为默认的浅复制（或成为成员复制）仅浅浅的赋值指针信

息。 我们再看以下代码片断，我们稍做修改：`StringBad headline1 (" Celery Stalks at Midnight ") ; StringBad knot ; knot = headline1 ;` 100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com