

全国计算机等级考试二级:C 对象的拷贝与赋值操作 PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/181/2021\\_2022\\_\\_E5\\_85\\_A8\\_E5\\_9B\\_BD\\_E8\\_AE\\_A1\\_E7\\_c97\\_181125.htm](https://www.100test.com/kao_ti2020/181/2021_2022__E5_85_A8_E5_9B_BD_E8_AE_A1_E7_c97_181125.htm) 计算机等级考试训练软件《百宝箱》我发现一些同事在编写一个类时，知道什么时候需要实现拷贝构造函数和赋值操作，但不知道什么时候拷贝构造函数被调用，什么时候赋值操作被调用，甚至把二者混为一谈。要弄明白这个问题，最简单的做法莫过于写个测试程序试一下。不过那样做也未必是好办法，实验的结果往往导致以偏概全的结论。不如好好想一下，弄清楚其中的原理，再去写程序去验证也不迟。拷贝构造函数，顾名思义，等于拷贝构造。它肩负着创建新对象的任务，同时还要负责把另外一个对象拷贝过来。比如下面的情况就调用拷贝构造函数：`CString str = strOther`. 赋值操作则只含有拷贝的意思，也就是说对象必须已经存在。比如下面的情况会调用赋值操作。`str = strOther`. 不过有的对象是隐式的，由编译器产生的代码创建，比如函数以传值的方式传递一个对象时。由于看不见相关代码，所以不太容易明白。不过我们稍微思考一下，就会想到，既然是根据一个存在的对象拷贝生成新的对象，自然是调用拷贝构造函数了。两者实现时有什么差别呢？我想有人会说，没有差别。呵，如果没有差别，那么只要实现其中一个就行了，何必要两者都实现呢？不绕圈子了，它们的差别是：拷贝构造函数对同一个对象来说只会调用一次，而且是在对象构造时调用。此时对象本身还没有构造，无需要去释放自己的一些资源。而赋值操作可能会调用多次，你在拷贝之前要释放自己的一些资源，否则会造成资源

泄露。明白了这些道理之后，我们不防写个测试程序来验证

```
一下我们的想法：#include #include #include class CString{
public: CString(). CString(const char* pszBuffer). ~CString().
CString(const CString&. operator=(const CString&. other){
if(this == amp. other)\n"). m_pszBuffer = other.m_pszBuffer !=
NULL ? strdup(other.m_pszBuffer) : NULL.}const CString&.
other){ printf("const CString&. other)\n"). if(this ==
&amp;.other) { return *this. } if(m_pszBuffer != NULL) {
free(m_pszBuffer). m_pszBuffer = NULL. } m_pszBuffer =
other.m_pszBuffer != NULL ? strdup(other.m_pszBuffer) : NULL.
return *this.}void test(CString str){ CString str1 = str. return.}int
main(int argc, char* argv[]){ CString str. CString str1 = "test".
CString str2 = str1. str1 = str. CString str3 = str3. test(str). return 0.}
```

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](http://www.100test.com)