

C 编译错误解决方法计算机等级考试 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/645/2021_2022_C___E7_BC_96_E8_AF_91_E9_c97_645688.htm 学习VC++时经常会遇到链接错误LNK2001，该错误非常讨厌，因为对于编程者来说，最好改的错误莫过于编译错误，而一般说来发生连接错误时，编译都已通过。产生连接错误的原因非常多，尤其LNK2001错误，常常使人不明其所以然。如果不深入地学习和理解VC++，要想改正连接错误LNK2001非常困难。初学者在学习VC++的过程中，遇到的LNK2001错误的错误消息主要为：unresolved external symbol “symbol”（不确定的外部“符号”）。如果连接程序不能在所有的库和目标文件内找到所引用的函数、变量或标签，将产生此错误消息。一般来说，发生错误的原因有两个：一是所引用的函数、变量不存在、拼写不正确或者使用错误；其次可能使用了不同版本的连接库。以下是可能产生LNK2001错误的原因：一．由于编码错误导致的LNK2001。1．不相匹配的程序代码或模块定义(.DEF)文件能导致LNK2001。例如,如果在C++源文件内声明了一变量“var1”，却试图在另一文件内以变量“VAR1”访问该变量,将发生该错误。2．如果使用的内联函数是在.CPP文件内定义的，而不是在头文件内定义将导致LNK2001错误。3．调用函数时如果所用的参数类型同函数声明时的类型不符将会产生LNK2001。4．试图从基类的构造函数或析构函数中调用虚拟函数时将会导致LNK2001。5．要注意函数和变量的可公用性，只有全局变量、函数是可公用的。静态函数和静态变量具有相同的使用范围限制。当试图

从文件外部访问任何没有在该文件内声明的静态变量时将导致编译错误或LNK2001。函数内声明的变量(局部变量)只能在该函数的范围内使用。C++的全局常量只有静态连接性能。这不同于C,如果试图在C++的多个文件内使用全局变量也会产生LNK2001错误。一种解决的方法是需要时在头文件中加入该常量的初始化代码,并在.CPP文件中包含该头文件;另一种方法是使用时给该变量赋以常数。

二.由于编译和链接的设置而造成的LNK2001

- 1.如果编译时使用的是/NOD(/NODEFAULTLIB)选项,程序所需要的运行库和MFC库在连接时由编译器写入目标文件模块,但除非在文件中明确包含这些库名,否则这些库不会被链接进工程文件。在这种情况下使用/NOD将导致错误LNK2001。
- 2.如果没有为wWinMainCRTStartup设定程序入口,在使用Unicode和MFC时将得到“unresolved external on _WinMain@16”的LNK2001错误信息。
- 3.使用/MD选项编译时,既然所有的运行库都被保留在动态链接库之内,源文件中对“func”的引用,在目标文件里即对“__imp__func”的引用。如果试图使用静态库LIBC.LIB或LIBCMT.LIB进行连接,将在__imp__func上发生LNK2001;如果不使用/MD选项编译,在使用MSVCxx.LIB连接时也会发生LNK2001。
- 4.使用/ML选项编译时,如用LIBCMT.LIB链接会在_errno上发生LNK2001。
- 5.当编译调试版的应用程序时,如果采用发行版模态库进行连接也会产生LNK2001;同样,使用调试版模态库连接发行版应用程序时也会产生相同的问题。
- 6.不同版本的库和编译器的混合使用也能产生问题,因为新版的库里可能包含早先的版本没有的符号和说明。
- 7.在不同的模块使用内

联和非内联的编译选项能够导致LNK2001。如果创建C++库时打开了函数内联 (/Ob1或/Ob2)，但是在描述该函数的相应头文件里却关闭了函数内联（没有inline关键字），这时将得到该错误信息。为避免该问题的发生，应该在相应的头文件中用inline关键字标志内联函数。

8. 不正确的/SUBSYSTEM或/ENTRY设置也能导致LNK2001。其实，产生LNK2001的原因还有很多，以上的原因只是一部分而已，对初学者来说这些就够理解一阵子了。但是，分析错误原因的目的在于避免错误的发生。LNK2001错误虽然比较困难，但是只要注意到了上述问题，还是能够避免和予以解决的。另一人给的解释 [Project] -- 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