

C基础:try{}catch(...){}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_E5_9F_BA_E7_A1_80_tr_c97_645837.htm 以前都是用try{} catch(...){}来捕获C中一些意想不到的异常，今天看了Winhack的帖子才知道，这种方法在VC中其实是靠不住的。例如下面的代码：
try { BYTE* pch . pch = (BYTE*)00001234 . //给予一个非法地址 *pch = 6 . //对非法地址赋值，会造成Access Violation 异常 }
catch(...) { AfxMessageBox("caught") . } 这段代码在debug下没有问题，异常会被捕获，会弹出 " caught " 的消息框。但在Release方式下如果选择了编译器代码优化选项，则VC编译器会去搜索try块中的代码，如果没有找到throw代码，他就会认为try catch结构是多余的，给优化掉。这样造成在Release模式下，上述代码中的异常不能被捕获，从而迫使程序弹出错误提示框退出。那么能否在release代码优化状态下捕获这个异常呢，答案是有的。就是__try, __except结构，上述代码如果改成如下代码异常即可捕获。 __try { BYTE* pch . pch = (BYTE*)00001234 . //给予一个非法地址 *pch = 6 . //对非法地址赋值，会造成Access Violation 异常 } __except(EXCEPTION_EXECUTE_HANDLER) { AfxMessageBox("caught") . } 但是用__try, __except块还有问题，就是这个不是C标准，而是Windows平台特有的扩展。而且如果在使用过程中涉及局部对象析构函数的调用，则会出现C2712的编译错误。那么还有没有别的办法呢？当然有，就是仍然使用C标准的try{}catch(..){}，但在编译命令行中加入 /EHa 的参数。这样VC编译器不会把try catch模块给优化掉了。找到一

篇比较好的英文文章谈这个问题：

http://members.cox.net/doug_web/eh.htm 用C 那么长时间了，居然这么基础的问题都搞错，真是汗颜。要加紧学习啊，Stay Hungry, Stay Foolish! 编辑特别推荐: 重解几道华为经典C语言面试题 应该怎样学习C语言？ C语言学习应该注意的地方 计算机二级C语言程序设计实战 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