

送给初学者的礼物：C 游戏编程起源连载一 PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/133/2021\\_2022\\_\\_E9\\_80\\_81\\_E7\\_BB\\_99\\_E5\\_88\\_9D\\_E5\\_c97\\_133523.htm](https://www.100test.com/kao_ti2020/133/2021_2022__E9_80_81_E7_BB_99_E5_88_9D_E5_c97_133523.htm) 第一章 Windows编程基础简介本章目的是介绍WINDOWS编程基础。在本章结束时，你应该能够很好的工作了，虽然可能是简单的WINDOWS程序。你需要有C语言的基础知识，我很少将C++的代码扩充到程序中。当然，由于WINDOWS本身就是面向对象的，一点类的知识是不会对你有什么损害的。如果你不熟悉C++，没有关系，我想你还是能从我这里学到大部分的东西。所有的程序代码都通过了MICROSOFT VISUAL C++ 6.0的编译，如果你还没有合适的编译器，那就用它好了，还是很棒的。开动吧！多数的Windows程序都需要Windows.h和Windowsx.h这两个头文件，要确保使用它们。当然，你还需要其它的标准的C的头文件，象stdio.h，conio.h等。除了这些，你还会经常看到在程序的开始有这样一行代码：`#define WIN32_LEANAND_MEAN`它表示Windows的头文件中将拒绝接受MFC的东西，这将加速你的build时间。如果你从没有打算应用MFC在你的游戏编程中，那就使用它吧。如果你以前从没有看过这种声明类型在#define后，直接加上一个“单词”，那么它的作用就是有条件编译。看看下面的例子：`#ifdef DEBUG_MODE printf("Debug mode is active!").#endif`意思是：如果程序的开始包含#define DEBUG\_MODE，那么就printf()，否则退出。这个对于你跟踪程序的逻辑错误是很有帮助的。WinMain()函数 DOS下的C语言从main()开始，Windows下的C语言从WinMain()开始

，一个空的WinMain()函数是这样的：`int WINAPI WinMain(HINSTANCE hinstance, HINSTANCE hPrevInstance, LPSTR lpCmdLine, int nCmdShow){ return(0).}`一个函数即使什么也没做，也应该返回一个值。是的，有好多东西我们不熟悉。首先的首先，WINAPI是个什么声明？WINAPI是在windows.h头文件中定义的一个宏，它把函数调用翻译成正确的调用约定。当我们在程序中需要用到汇编语言的时候，我们在来深究它好了，记住，如果要用WinMain()，就必须要有WINAPI。下一步让我们来看看括号里的四个参数：

**HINSTANCE hinstance**：HINSTANCE是一个句柄类型的标识符。变量hinstance是一个整数，用于标识程序实例。Windows设置这个参数的值，并把它传递给你的程序代码。很多Windows函数都要用到它。

**HINSTANCE hPreInstance**：你不用担心这个参数，它已经被废掉了。它只是为古老的Windows版本服务的。你将还会看到类似的情况。

**LPSTR lpCmdLine**：是一个指向字符串的指针，它仅在程序名是从DOS命令行输入或是从Run对话框中输入时才起作用。因此，很少被程序代码所用。

**int nCmdShow**：决定了窗口在初始显示时的状态。Windows通常给这个参数分配一个值。通常是SW\_打头的一个常量。例如SW\_SHOWNORMAL表示默认状态，SW\_MAXIMIZE或SW\_MINIMIZE分别表示最大和最小模式等等。以上大体上是WinMain()的参数的介绍。

下面对变量、参数、常量、类等的命名方法介绍一下。

一、消息 当你在DOS下编程的时候，你不必担心其它程序的运行，因为DOS是独占模式。但你在Windows平台上编程时，你不得不考虑其它正在运行的程序。鉴于此，Windows通过“

消息”来连接操作申请和具体操作。简单的说，就是我们指示程序或程序本身向Windows发出诸如移动窗口、放大窗口、关闭窗口等申请，Windows再根据申请，考察实地情况，拒绝或发出指令，让程序（计算机）作出相应的动作。再例如，鼠标随时向Windows发出消息，汇报光标位置，左键或右键是否按下等，Windows再根据消息作出相应的反应。总之，无论何时，Windows都要随时掌控所有的消息，而且，Windows是一直不断地接收到各种消息。这种功能是通过一种被命名为CALLBACK函数类型实现的。不用害怕，消息传递来传递去都是由Windows自己完成的，你只要声明一个CALLBACK函数就可以了，就像WINAPI用在WinMain()前一样。如果还没有明白，不要紧，往下看你就明白了。现在，我要离开这个话题一会儿，因为你只有先建立窗口（Windows），传递消息才有可能实现。

## 二、窗口类

现在谈论一点C++的知识，因为要想建立一个窗口，你就得先建立一个窗口类。窗口类包含所有的有关窗口的信息，如用什么样的鼠标符号，菜单样式等等。开发任何一个窗口程序，都离不开窗口类的建立。为了达到此目的，你必须填写WNDCLASSEX结构。EX的意思是“扩充”的意思，因为有一个老的结构叫作WNDCLASS，这里，我们将使用WNDCLASSEX结构，它的样子如下：

```
typedef struct  
_WNDCLASSEX {  
    UINT cbSize;  
    UINT style;  
    WNDPROC lpfnWndProc;  
    int cbClsExtra;  
    int cbWndExtra;  
    HANDLE hInstance;  
    HICON hIcon;  
    HCURSOR hCursor;  
    HBRUSH hbrBackground;  
    LPCTSTR lpszMenuName;  
    LPCTSTR lpszClassName;  
    HICON hIconSm;  
} WNDCLASSEX;
```

这个结构有

不少成员，讨厌的是，你必须为窗口类设置每一个成员。莫发愁，纸老虎一个。让我们来速成。 `UINT cbSize`：指定了以字节为单位的结构的大小。这个成员是通过 `sizeof(WNDCLASSEX)` 实现的。你将会经常看到它，尤其是你使用了 DirectX。 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)