Windows 操作系统

C/C++ 程序实验

姓名：_____陈展博_____

学号：____1221001003_____

班级：__计科一班_____

院系：___信息工程学院____

_____2024_____年__10_月_9_日

实验一　Windows 进程观测

一、背景知识

二、实验目的

通过对 Windows 编程，进一步熟悉操作系统的基本概念，较好地理解 Windows 的结构。

三、实验内容与步骤

1、D:\> CL Hello.cpp

来创建可执行的 Hello.EXE。

操作能否正常进行？如果不行，则可能的原因是什么？

操作无法正常运行原因为.cpp 文件中使用了中文字符

步骤 4：运行 Hello.EXE 程序，产生用户键入的一行文字。

运行结果 (如果运行不成功，则可能的原因是什么？) ：

运行成功，输出 Hello,Windows!

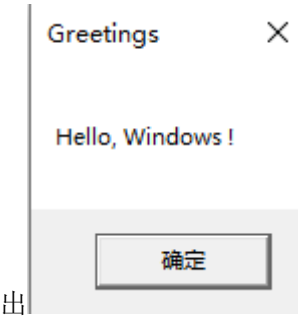
请按任意键继续. . .

2、也可以利用任何其他文本编辑器键入程序代码，如果这样，例如使用 WORD 来键入和编辑程序，则应该注意什么问题？

使用 WORD 来键入和编辑程序应该注意区分中英文字符（标点符号）。

D:\> CL 1-2.cpp

运行结果 (试将其中的信息与清单 1-1 程序的运行结果进行比较) ：

运行成功，输出 

相较清单 1-1 多了图形交互界面

3、D:\> CL 1-3.cpp

运行结果：Current process priority: Normal

请按任意键继续. . .

选作: 如何修改程序 1-3，检验进程优先级修改前后的结果对比。

```
#include <windows.h>
    # include <iostream>
     // 确定自己的优先权的简单应用程序
    void main( )
    {
  char a;
  std::cout<<"Enter Character a To Terminate It..."<<std::endl;
  std::cin>>a;
  while (a!='a'){
      // 从当前进程中提取句柄
      HANDLE hProcessThis = :: GetCurrentProcess( ) ;
      // 请求内核提供该进程所属的优先权类
      DWORD dwPriority = :: GetPriorityClass(hProcessThis) ;
       // 发出消息，为用户描述该类
 std :: cout << "Current process priority: " ;
      switch(dwPriority)
      {
        case HIGH_PRIORITY_CLASS:
          std :: cout << "High" ;
          break;
        case NORMAL_PRIORITY_CLASS:
          std :: cout << "Normal" ;
          break;
        case IDLE_PRIORITY_CLASS:
          std :: cout <<"Idle" ;
          break;
        case REALTIME_PRIORITY_CLASS:
          std :: cout << "Realtime";
```

```
            break;
        default:
            std :: cout << "<unknown>" ;
            break;
    }
    std::cout<<std::endl<<"Enter Character a To Terminate It..."<<std::endl;
std::cin>>a;
}


    system("pause");

    }
```

4、在“命令提示符”窗口输入如下命令，产生 1-4.EXE 文件，然后运行。

D:\> CL 1-4.cpp

运行结果：Process ID: 1204, EXE file: sihost.exe, % in kernel mode: 53

Process ID: 3948, EXE file: svchost.exe, % in kernel mode: 21

Process ID: 3984, EXE file: svchost.exe, % in kernel mode: 39

Process ID: 2660, EXE file: taskhostw.exe, % in kernel mode: 65

Process ID: 6140, EXE file: ctfmon.exe, % in kernel mode: 66

Process ID: 6436, EXE file: explorer.exe, % in kernel mode: 64

Process ID: 6604, EXE file: QQPCTray.exe, % in kernel mode: 37

Process ID: 6696, EXE file: ChsIME.exe, % in kernel mode: 0

Process ID: 6984, EXE file: svchost.exe, % in kernel mode: 50

Process ID: 7444, EXE file: StartMenuExperienceHost.exe, % in kernel mode: 42

Process ID: 7728, EXE file: RuntimeBroker.exe, % in kernel mode: 70

Process ID: 8084, EXE file: TextInputHost.exe, % in kernel mode: 44

Process ID: 7484, EXE file: SearchApp.exe, % in kernel mode: 38

Process ID: 7264, EXE file: RuntimeBroker.exe, % in kernel mode: 65

Process ID: 8612, EXE file: RuntimeBroker.exe, % in kernel mode: 84

Process ID: 9076, EXE file: StudentMain.exe, % in kernel mode: 66

Process ID: 8868, EXE file: EzMonitor.exe, % in kernel mode: 95

Process ID: 8592, EXE file: secmon.exe, % in kernel mode: 100

Process ID: 6156, EXE file: dllhost.exe, % in kernel mode: 100

Process ID: 8564, EXE file: SystemSettings.exe, % in kernel mode: 47

Process ID: 3144, EXE file: ApplicationFrameHost.exe, % in kernel mode: 100

Process ID: 8720, EXE file: UserOOBEBroker.exe, % in kernel mode: 100

Process ID: 8008, EXE file: svchost.exe, % in kernel mode: 100

Process ID: 3120, EXE file: ShellExperienceHost.exe, % in kernel mode: 67

Process ID: 1644, EXE file: RuntimeBroker.exe, % in kernel mode: 80

Process ID: 8356, EXE file: taskhostw.exe, % in kernel mode: 39

Process ID: 2088, EXE file: WINWORD.EXE, % in kernel mode: 41

Process ID: 5132, EXE file: WINWORD.EXE, % in kernel mode: 23

Process ID: 5692, EXE file: cmd.exe, % in kernel mode: 77

Process ID: 6416, EXE file: conhost.exe, % in kernel mode: 82

Process ID: 8588, EXE file: svchost.exe, % in kernel mode: 50

Process ID: 8316, EXE file: Code.exe, % in kernel mode: 36

Process ID: 4368, EXE file: Code.exe, % in kernel mode: 100

Process ID: 2552, EXE file: Code.exe, % in kernel mode: 68

Process ID: 1944, EXE file: Code.exe, % in kernel mode: 50

Process ID: 5520, EXE file: Code.exe, % in kernel mode: 24

Process ID: 4776, EXE file: Code.exe, % in kernel mode: 38

Process ID: 1932, EXE file: Code.exe, % in kernel mode: 44

Process ID: 8576, EXE file: Code.exe, % in kernel mode: 19

Process ID: 4136, EXE file: smartscreen.exe, % in kernel mode: 100

Process ID: 8804, EXE file: 1-4.exe, % in kernel mode: 50

Process ID: 8980, EXE file: conhost.exe, % in kernel mode: 66

Process ID: 4484, EXE file: cmd.exe, % in kernel mode: 0

Process ID: 5408, EXE file: 1-4.exe, % in kernel mode: 100

Process ID: 4744, EXE file: conhost.exe, % in kernel mode: 50

Process ID: 1280, EXE file: cmd.exe, % in kernel mode: 0

Process ID: 5956, EXE file: 1-4.exe, % in kernel mode: 100

Process ID: 9112, EXE file: conhost.exe, % in kernel mode: 88

Process ID: 4432, EXE file: cmd.exe, % in kernel mode: 100

Process ID: 1572, EXE file: 1-4.exe, % in kernel mode: 50

Process ID: 4264, EXE file: conhost.exe, % in kernel mode: 16

Process ID: 7580, EXE file: cmd.exe, % in kernel mode: 0

Process ID: 8624, EXE file: 1-4.exe, % in kernel mode: 0

Process ID: 3468, EXE file: conhost.exe, % in kernel mode: 33

程序的功能描述

程序通过对所有正在运行的进程进行循环遍历将当前运行进程名和消耗在内核模式下的时间百分数都显示出来。尝试多开 1-4.exe 后可以观察到不同的 kernel mode 百分比占比。