**操作系统课程设计实验报告**

实验名称： 进程控制

姓名/学号：马浩元 1120172723

1. **实验目的**

熟悉不同系统下如何创建进程。

学习在编程中实现进程控制。

学习不同系统下的进程控制。

1. **实验内容**

设计并实现Unix的“time”命令。“mytime”命令通过命令行参数接受要运行的程序，创建一个独立的进程来运行该程序，并记录程序运行的时间。

1. **实验环境**

CPU：intel core i7

内存：8G ddr4

windows系统：win 10

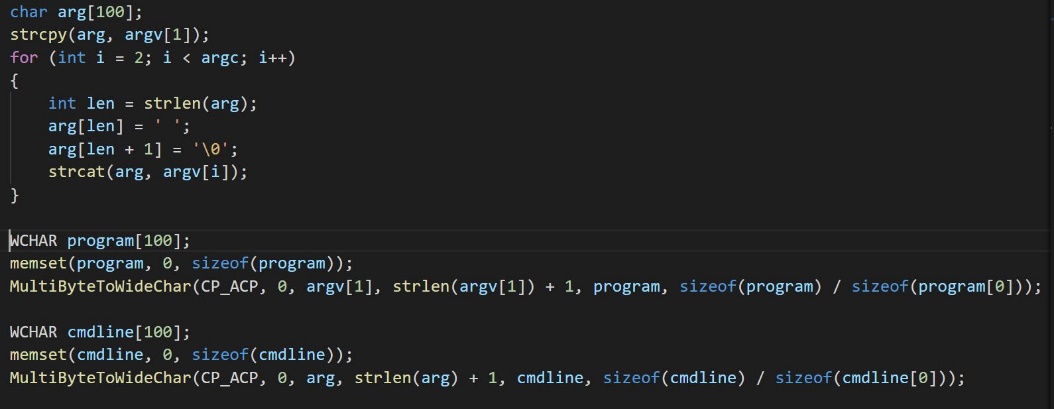
Linux系统：Ubuntu 18.04.3

虚拟机平台：VMware 14

Linux 内核：Linux 5.3.11

开发工具：Visual Studio 2017 社区版

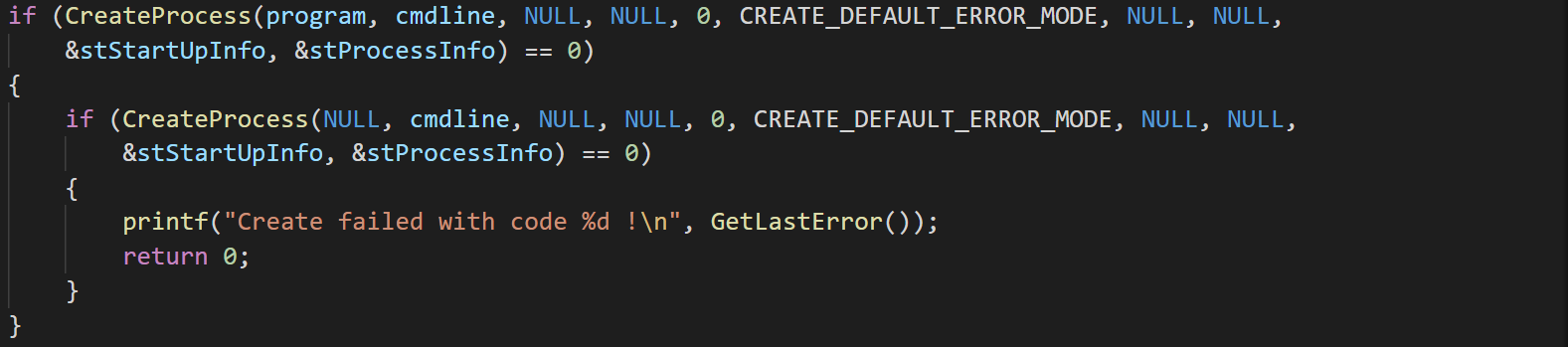
1. **程序设计与实现**
2. windows
3. 分离出目标程序路径（argv[1]），将目标程序的命令行参数组合到同一字符串。并使用MultiByteToWideChar（）将其参数转换成宽字符类型。



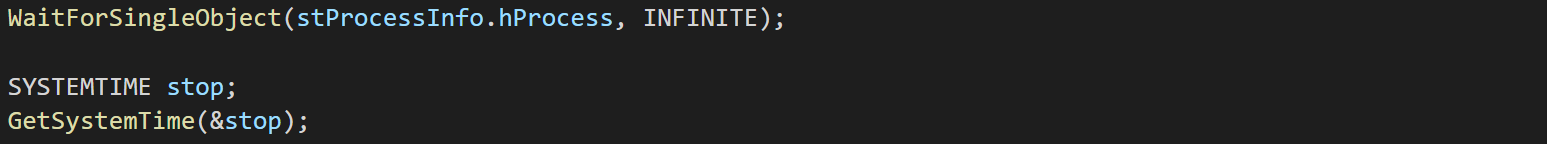
1. 使用GetSystemTime（）的到当前系统时间。



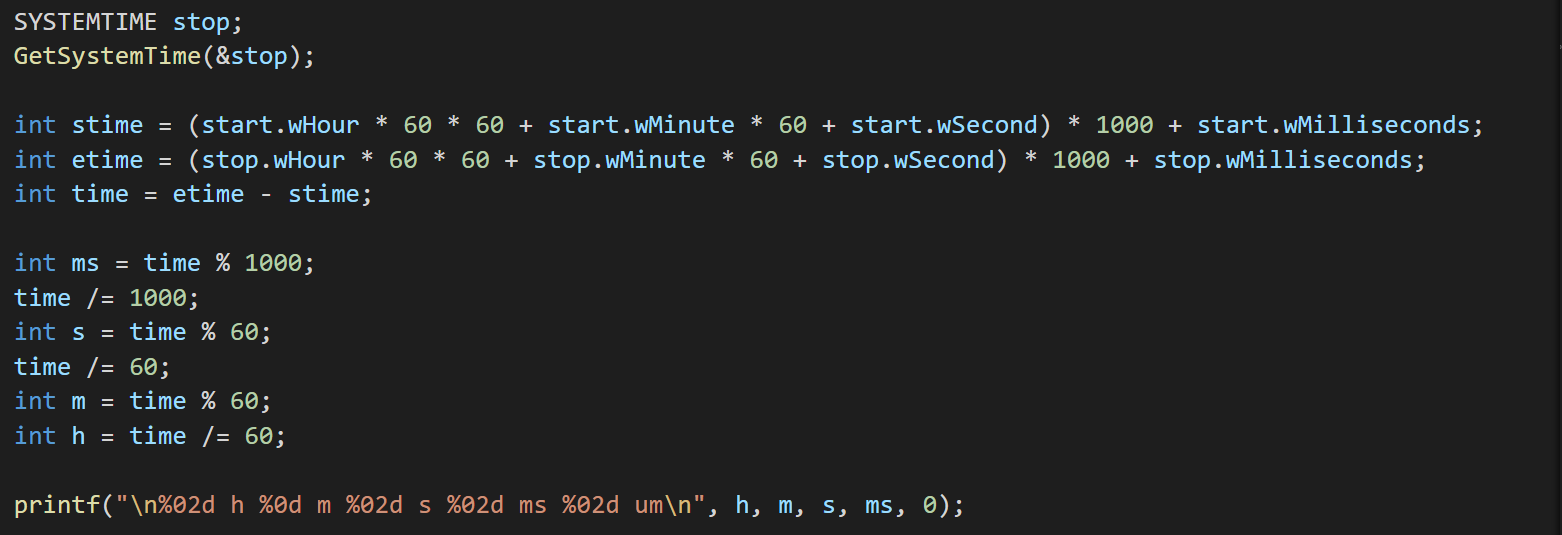
1. 尝试使用CreateProcess(program, cmdline,…)创建进程，其中program是新程序名，cmdline是其命令行参数。
2. 若3）失败说明当前目录下未找到相应程序，则新程序为系统程序，使用CreateProcess(NULL, cmdline,…)继续创建



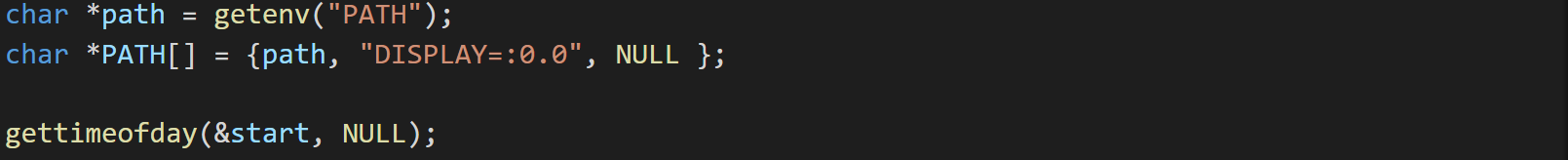
1. 使用WaitForSingleObject（）等待新进程运行完毕。
2. 再次使用GetSystemTime（）的到当前系统时间。



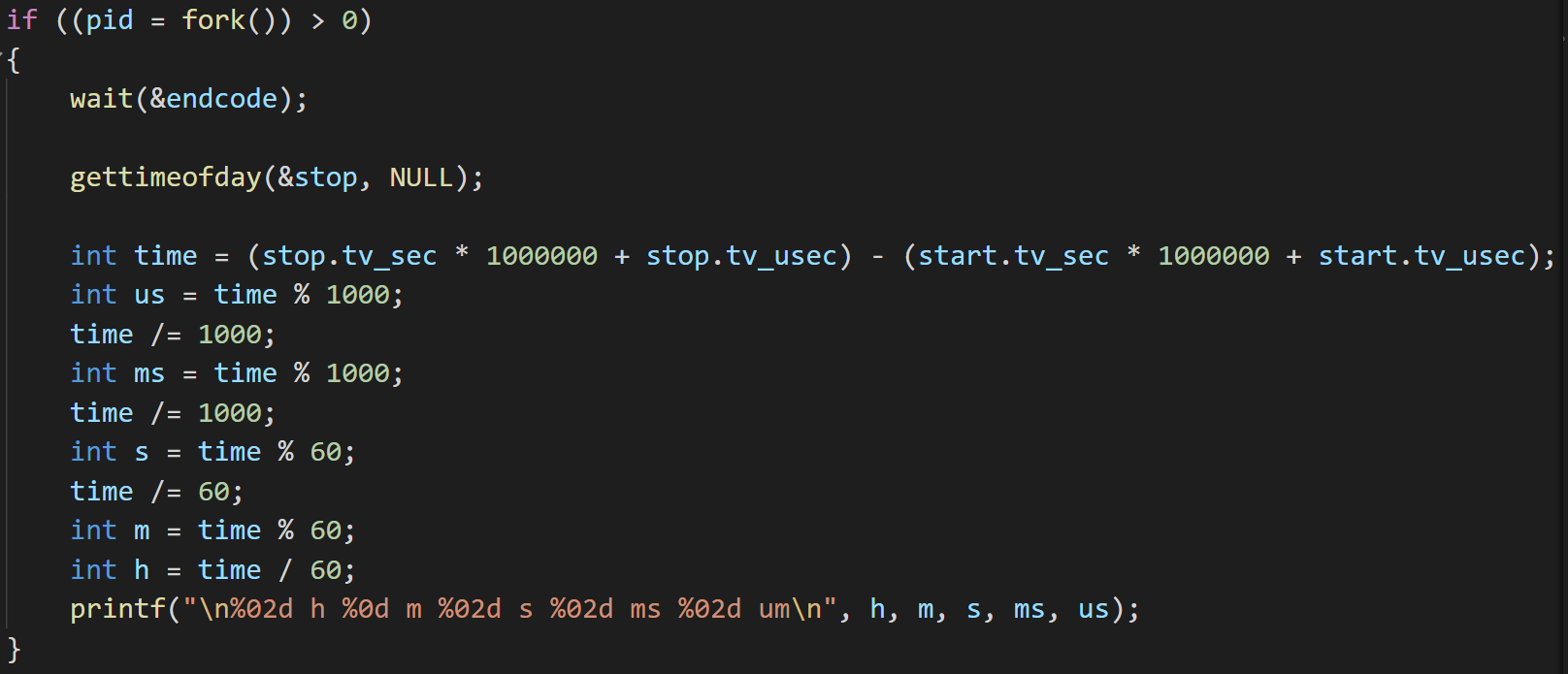
1. 计算所用时间并格式化输出。



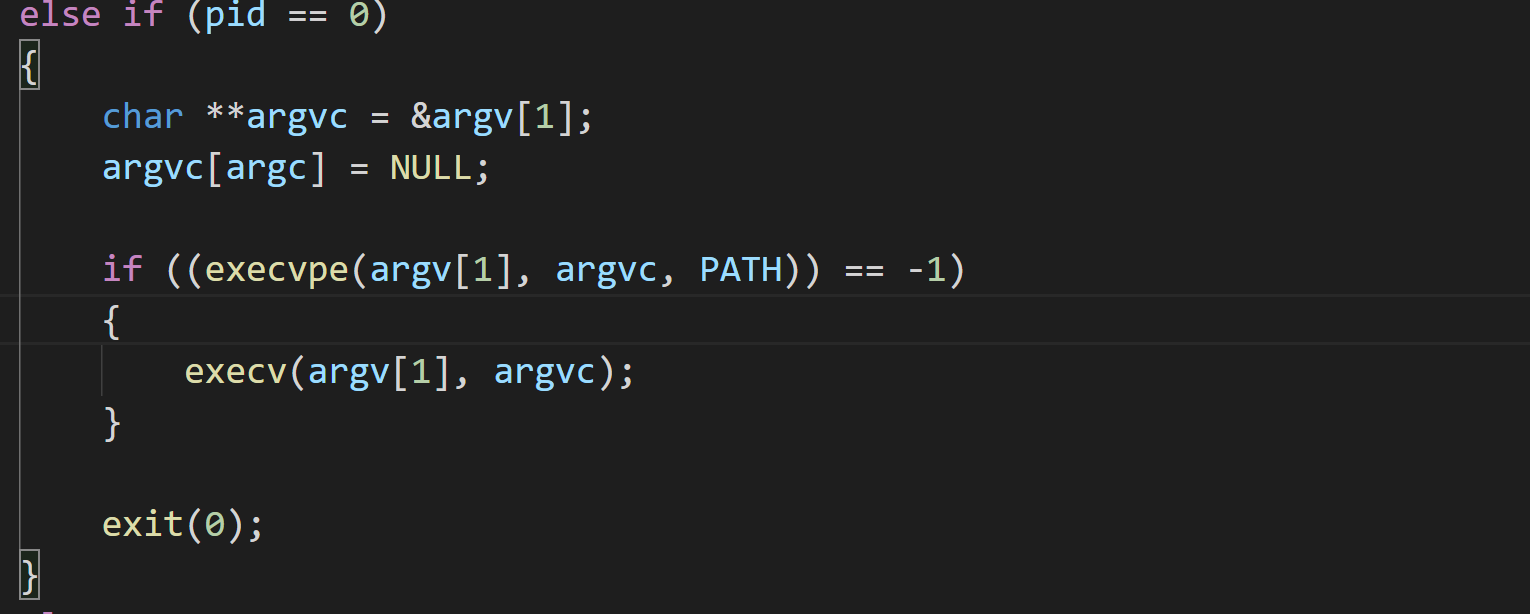
1. Linux
2. 得到环境变量，使用gettimeofday（）得到当前时间。



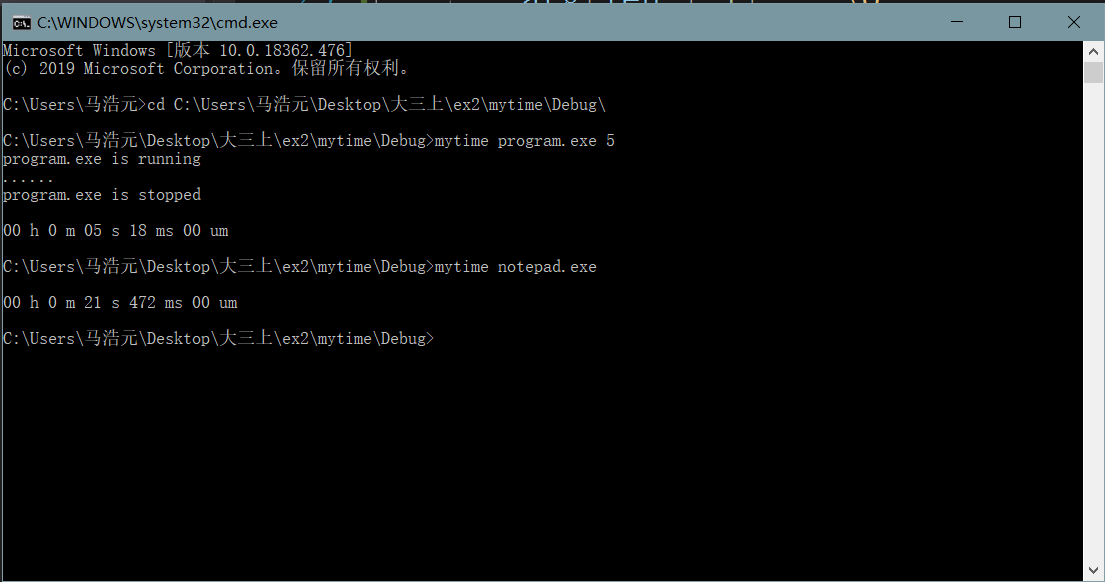
1. 使用fork（）创建子进程。
2. 在父进程使用wait（）等待子进程结束。并再次使用gettimeofday（）得到结束时间，计算所用时间并格式化输出。



1. 在子进程中尝试使用execvpe（）运行新程序，若3）失败，说明新程序不是系统程序，使用execve（）运行。

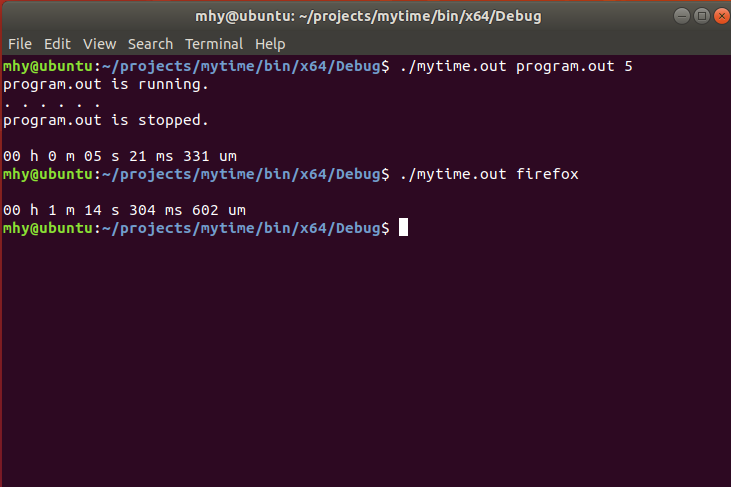


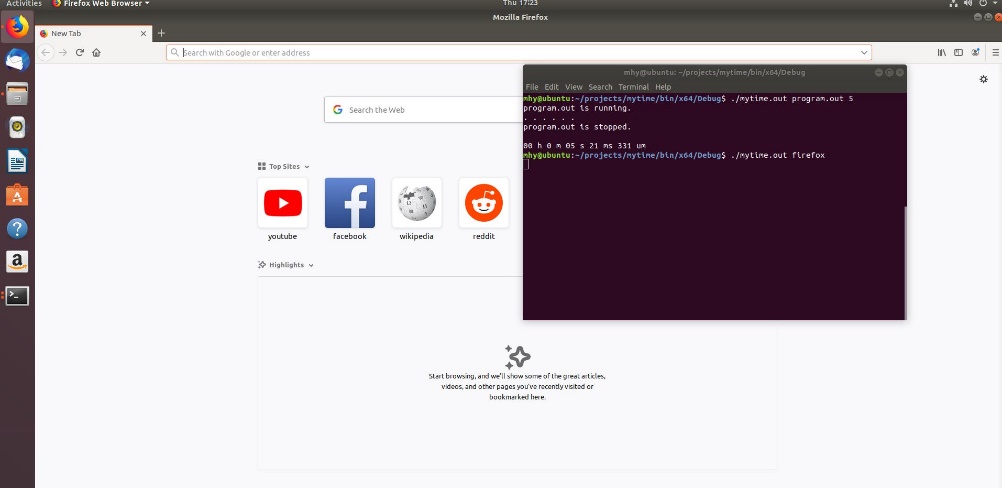
1. 实验结果
2. Windows





1. linux





1. **实验收获与体会**

通过本次实验，我进一步学习了进程控制在编程中的应用，并且掌握了不同系统中对进程的不同处理方式。影响最深刻的还是linux系统中，exec族函数的用法。这部分内容，结合网上的解释和自己动手调试，让我深刻了解了他们之间的区别与不同。而且这次实验是第一次调用这些系统层面的接口函数，更加了解进程的概念。进一步提高了自己的编程水平。