**操作系统课程设计实验报告**

实验名称： 复制文件

姓名/学号：马浩元 1120172723

1. **实验目的**

熟悉不同系统下如何操作文件。

学习在编程中实现对文件的访问，读写。

学习不同系统下文件属性，权限的读取与设置。

学习linux的链接机制。

1. **实验内容**

完成一个目录复制命令mycp，包括目录下的文件和子目录。复制后，不仅权限一致，而且时间属性也一致。Linux下要求支持软链接。

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. 基本设计思路

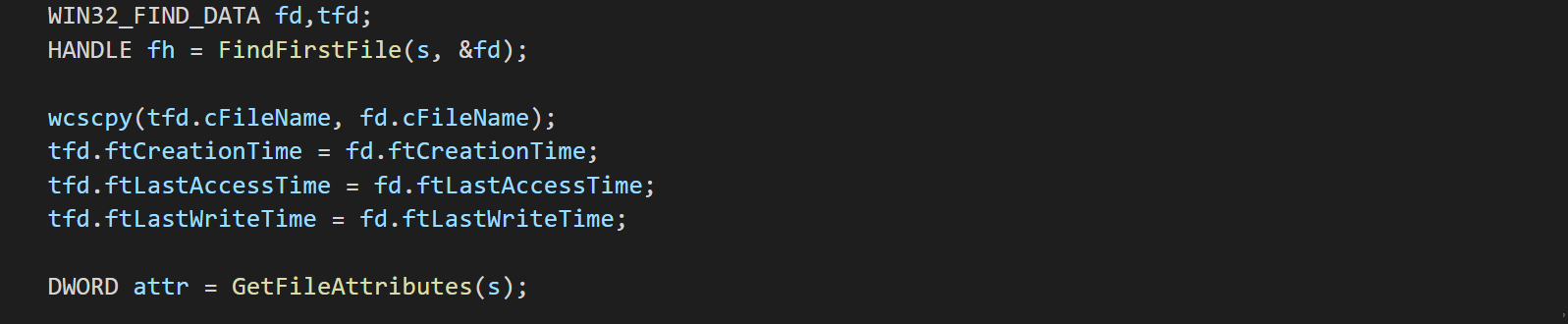
复制非目录文件的大体流程为，将源文件读入内存，转到目标目录，新建文件，从源文件中读取数据再写入新文件。对于目录文件，由于存在子目录，所以可以用递归的方法实现。对于linux下的软连接，先得到其自身的文件属性，再得到它指向的文件，最后新建链接。

1. Windows
2. 在主函数中，将得到的参数转换为宽字符，若传入的参数为相对路径，则统一转换为绝对路径，方便后续处理。最后调用cp函数完成复制。



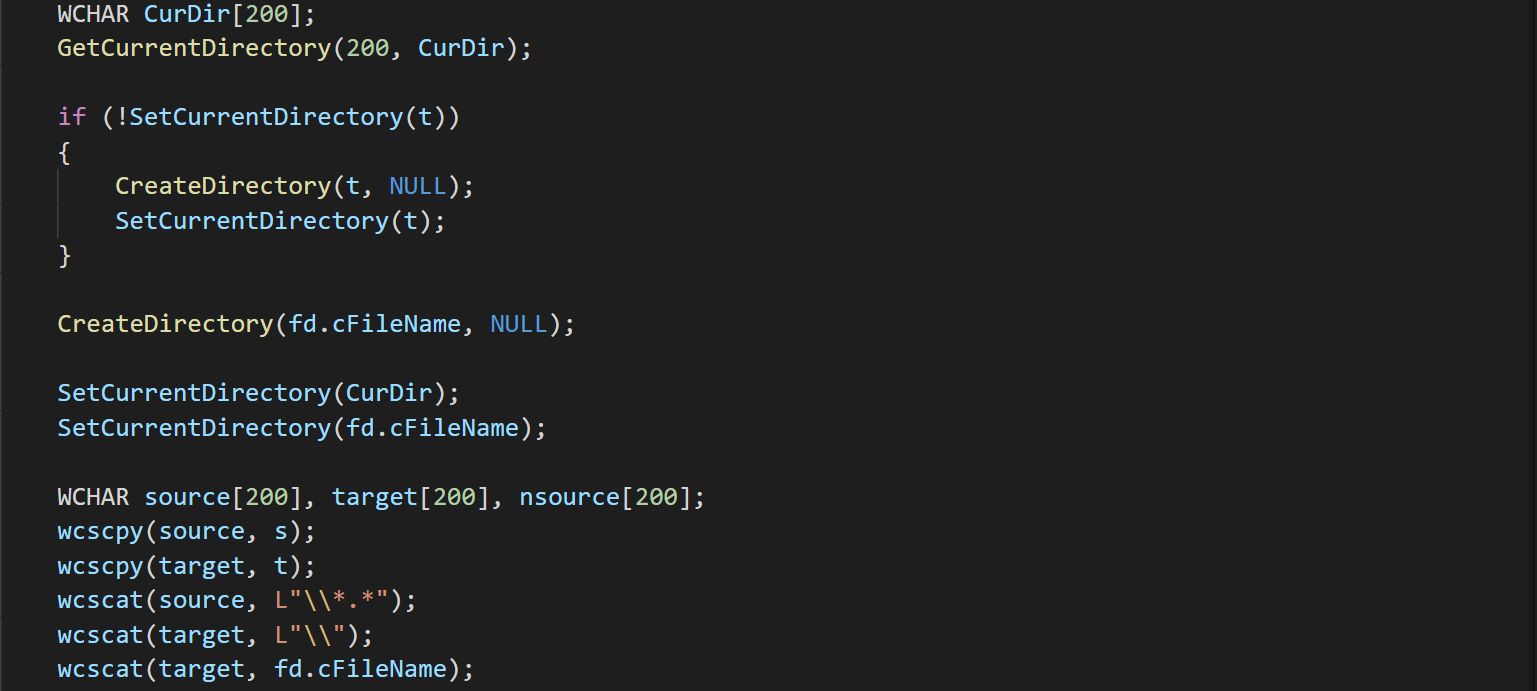
cp()函数的实现：

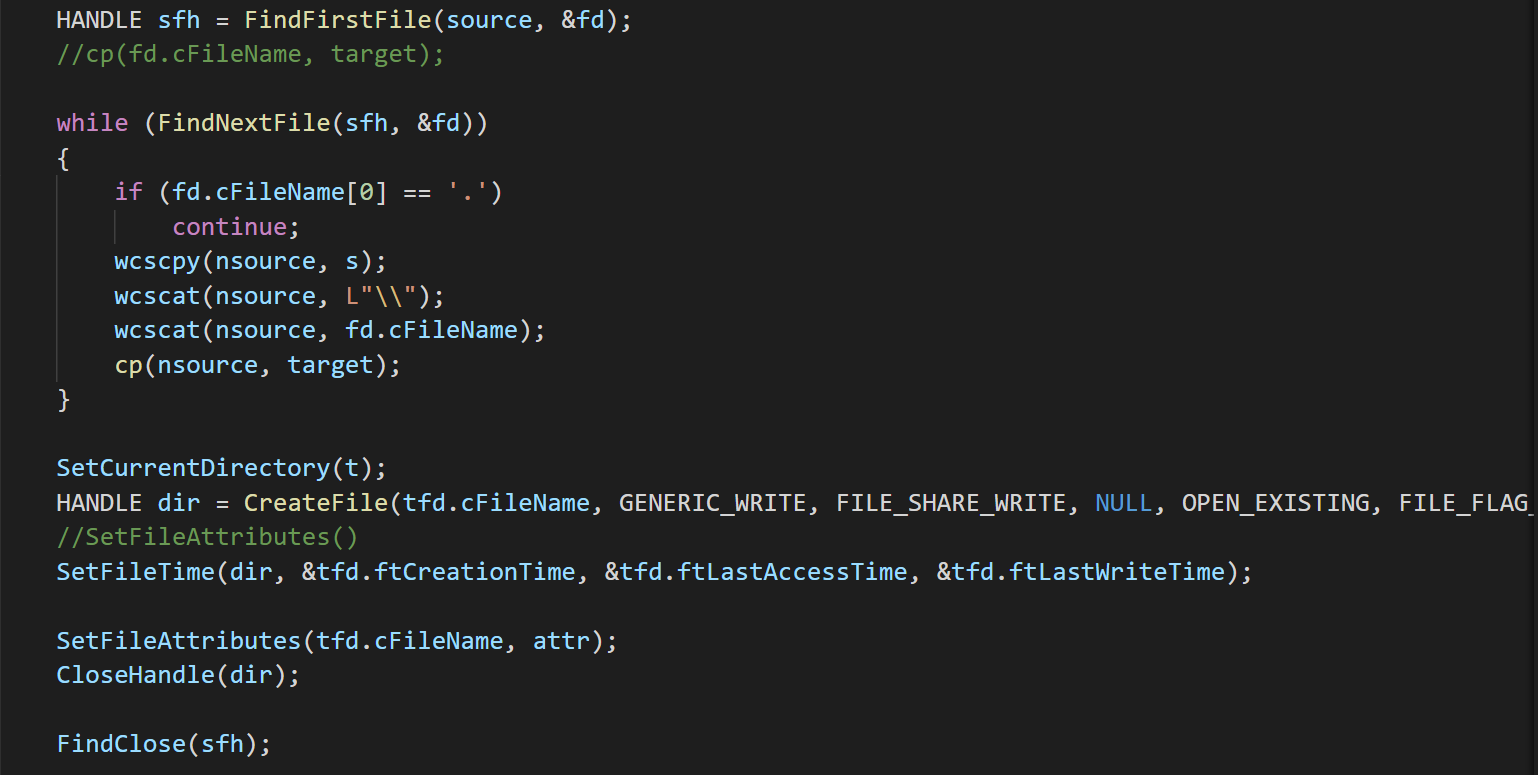
1. 调用FindFileFirst得到源文件的相关属性。同时得到其时间属性和权限属性。



1. 判断源文件是不是目录文件，若是：

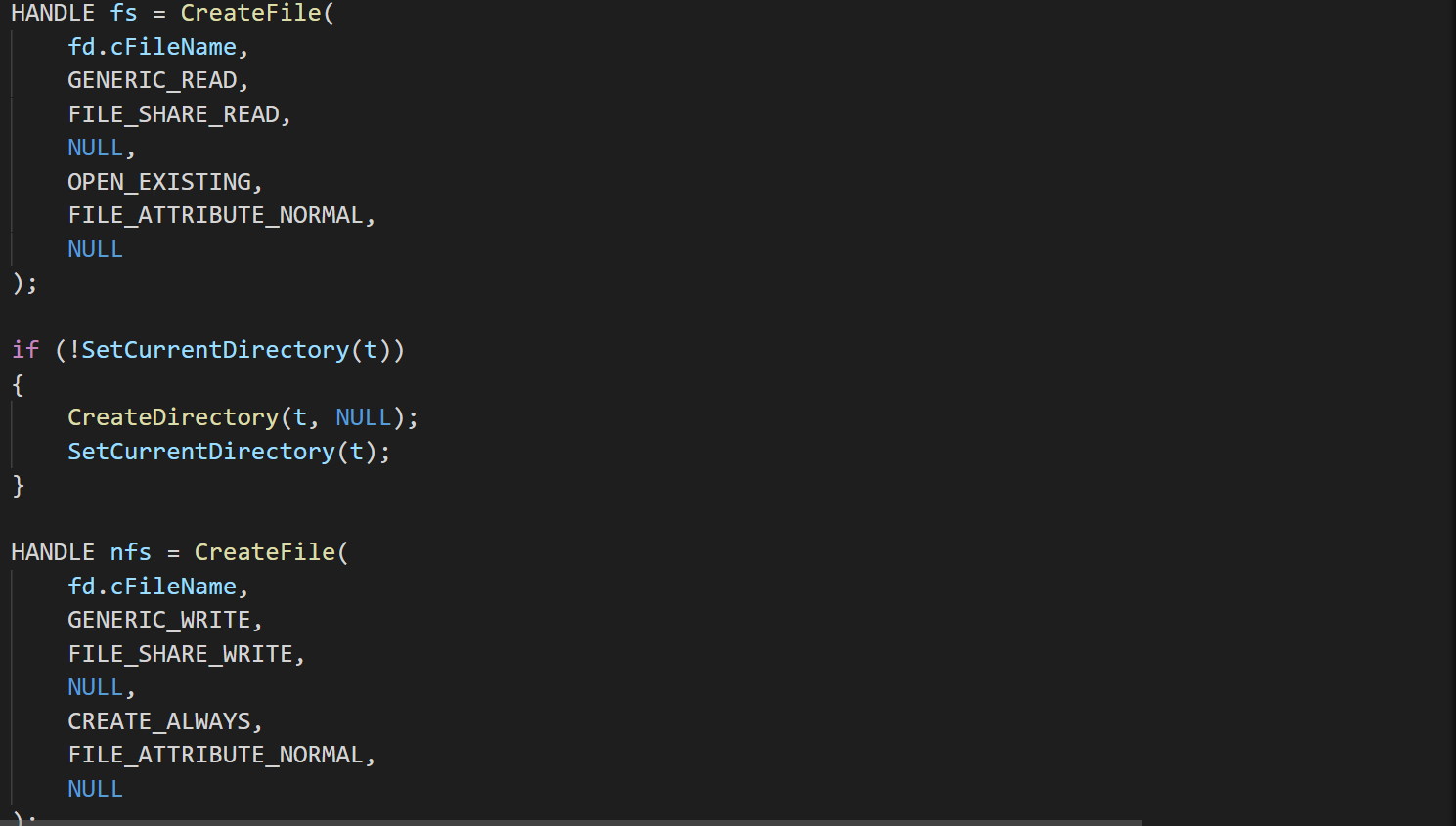
记录当前目录🡪打开（不存在则创建）目标目录🡪在新目录创建源目录的副本🡪进入源文件(目录)🡪遍历所有目录项，对于每一个目录项，更新目标目录，并递归调用cp函数将其拷贝🡪回到最初的目标目录，为新复制的目录设置时间，权限属性🡪关闭相关句柄。

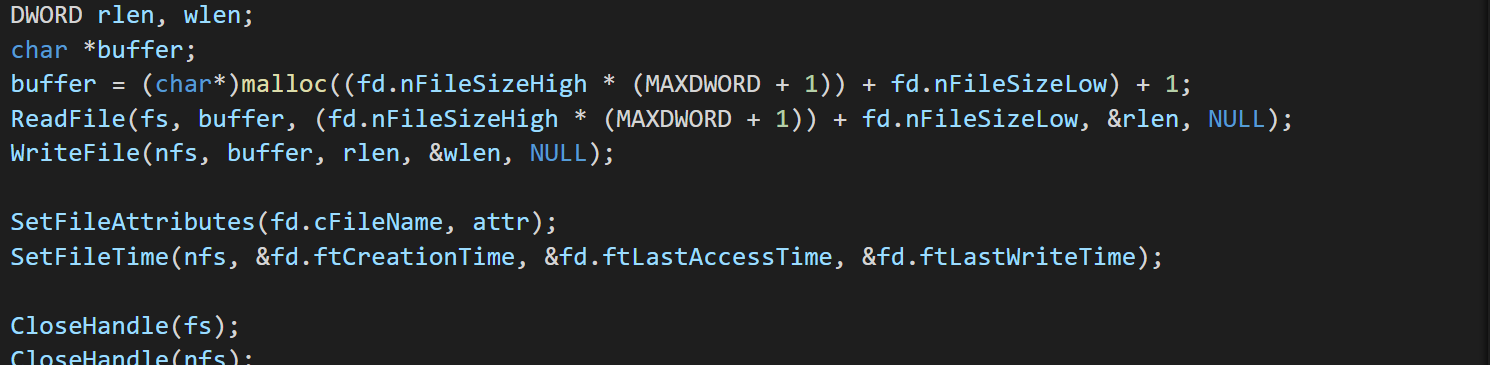




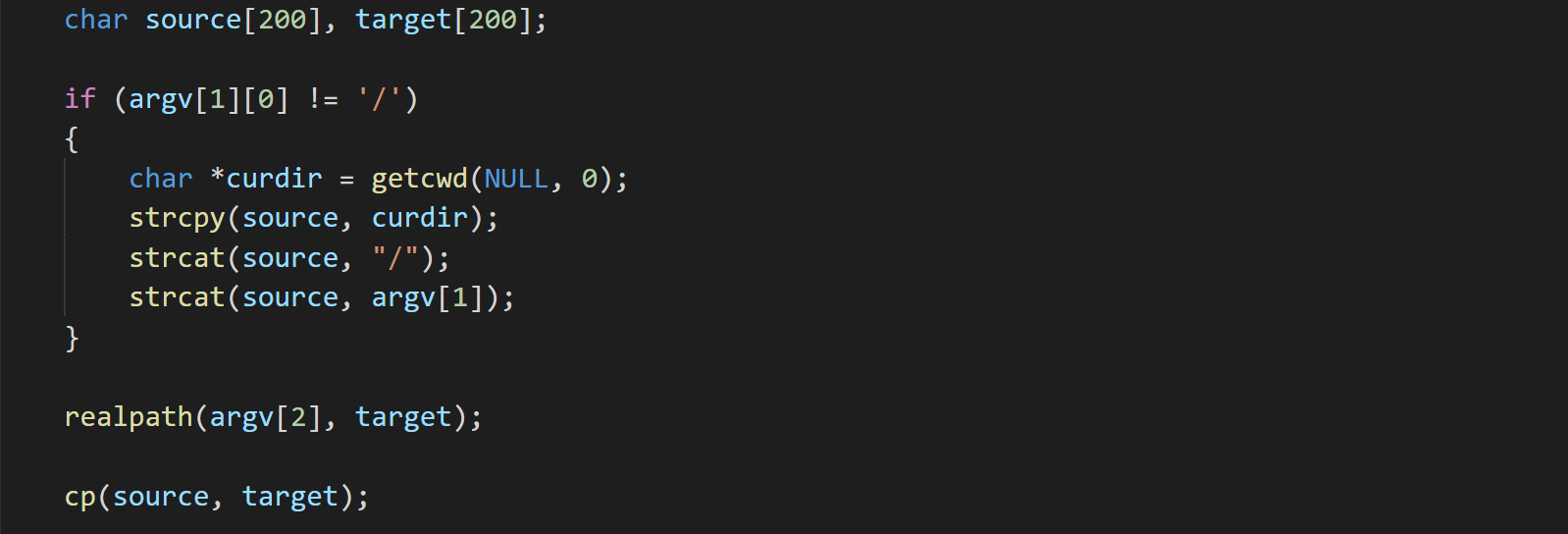
1. 若不是目录文件，则：

调用CreateFile打开源文件🡪打开（不存在则创建）目标目录🡪在新目录创建源文件的副本🡪使用ReadFile从源文件读取数据，用WriteFile将数据写入新文件🡪为新文件设置时间，权限属性🡪关闭相关句柄。

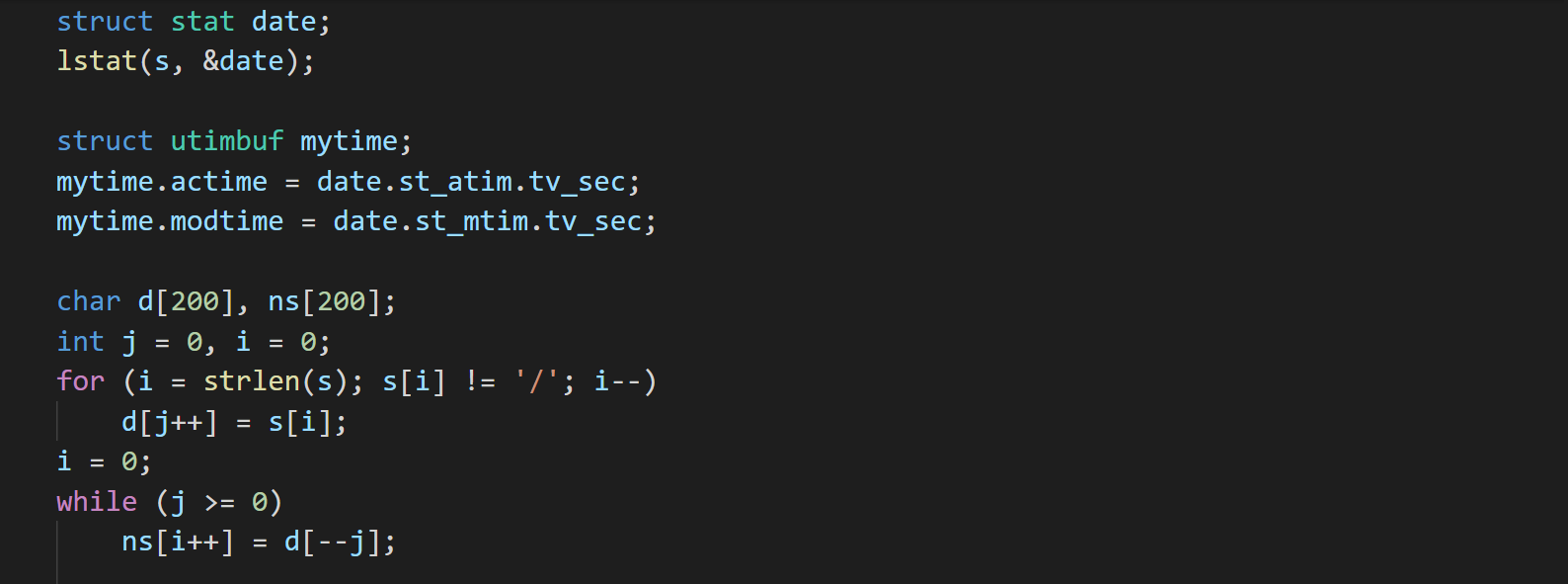




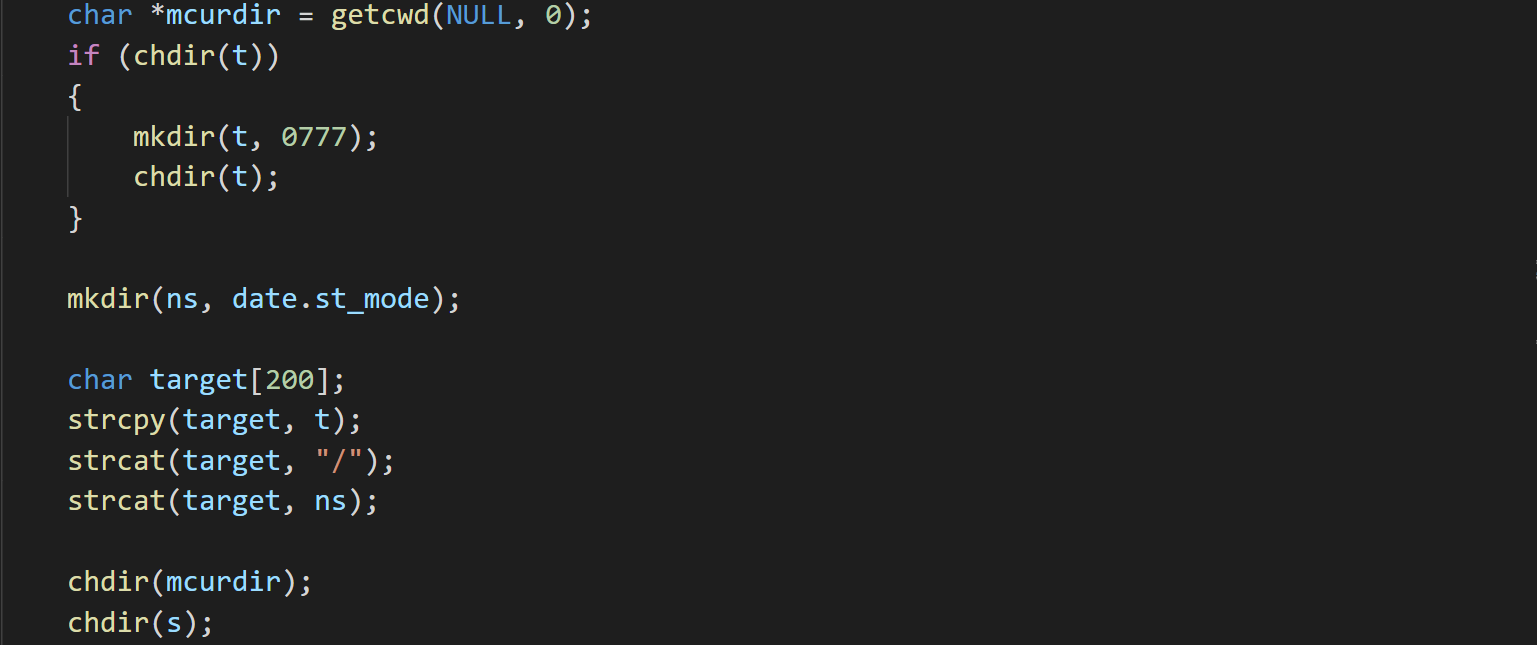
1. Linux
2. 在主程序内，若参数为相对路径，将其转换为绝对路径。调用cp函数。



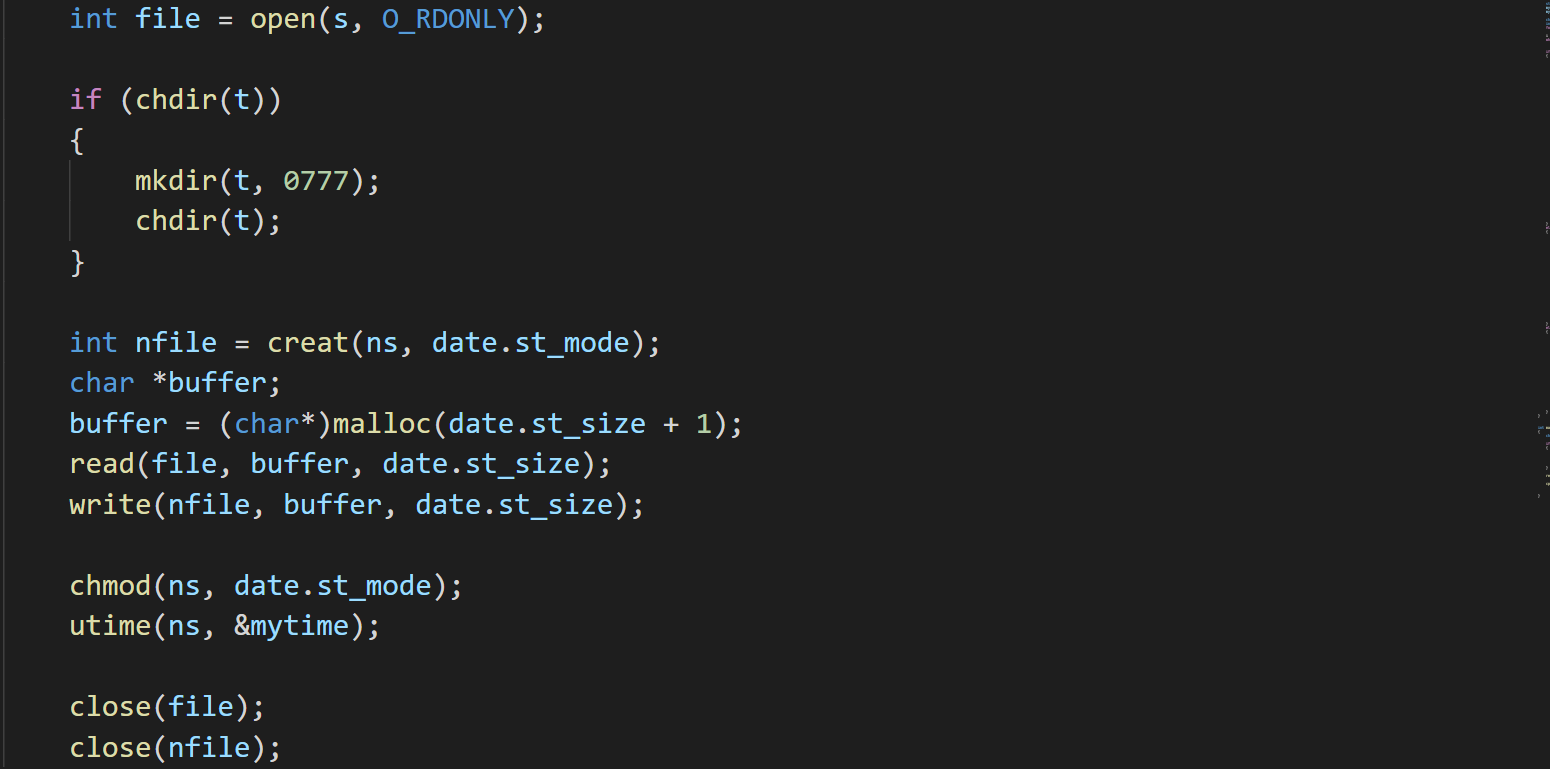
1. 使用lstat得到源文件的信息，并记录其时间信息。将源文件的文件名从路径中分离出来(ns)，方便后续处理。



1. 判断源文件类型，若为目录文件或非目录非链接文件，处理逻辑与Windows下类似。区别在于用mkdir新建目录，用chdir转到目录，用chmod，utime设置权限和时间等等。





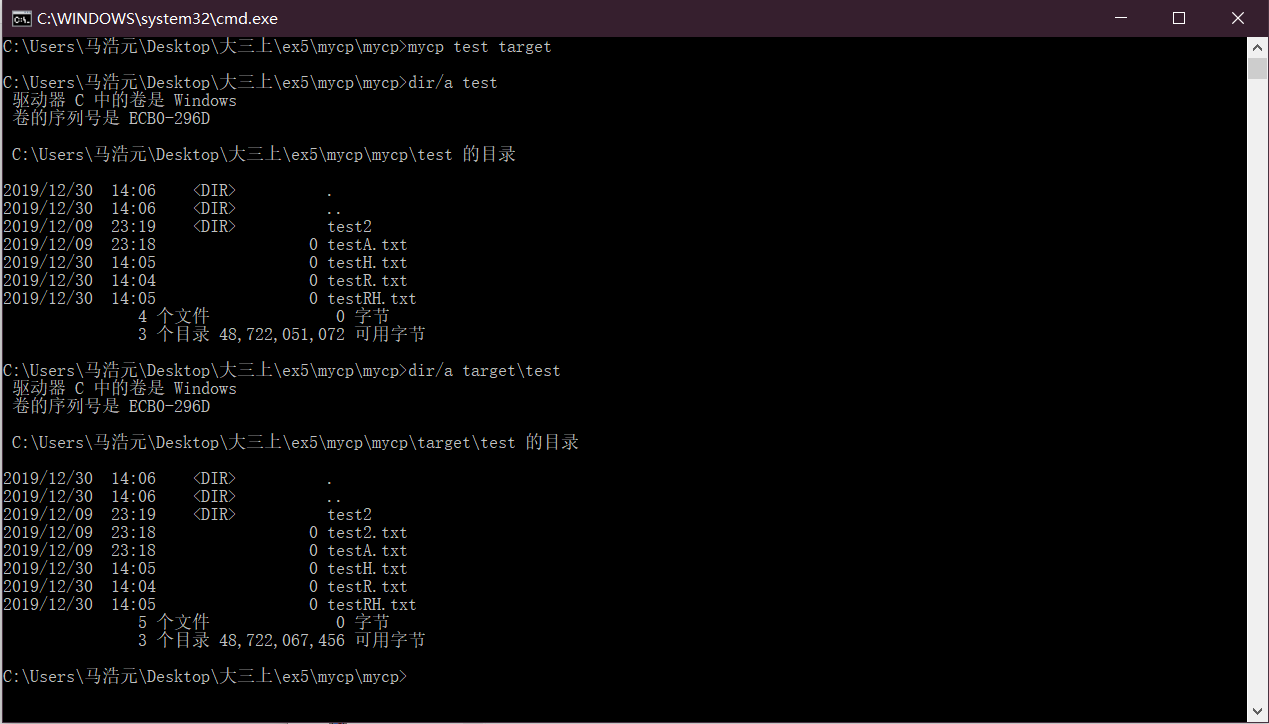


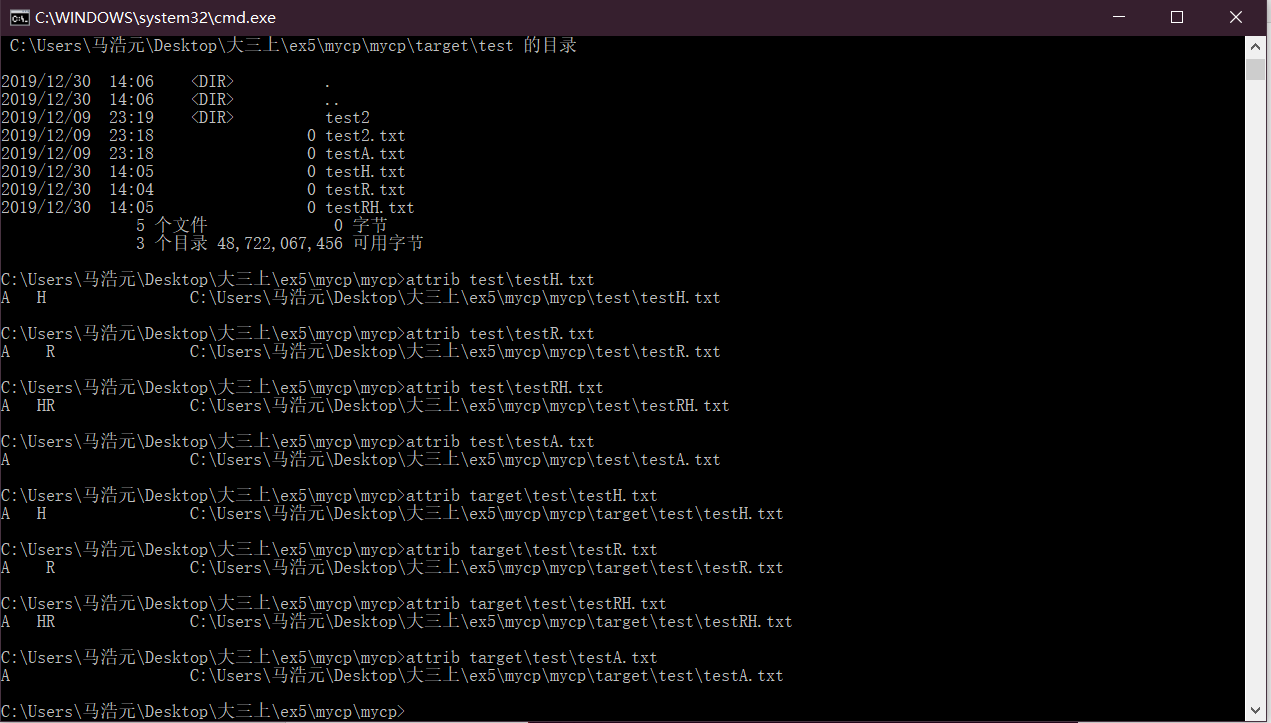
1. 若为链接文件，则：

将链接文件名拼接在目标路径后作为新链接文件的路径target🡪使用realpath得到链接文件指向的的真实路径lk🡪使用symlink(lk，target)；创建新的链接文件🡪复制原链接文件的时间信息🡪使用lchmod和lutimes设置新链接文件的权限

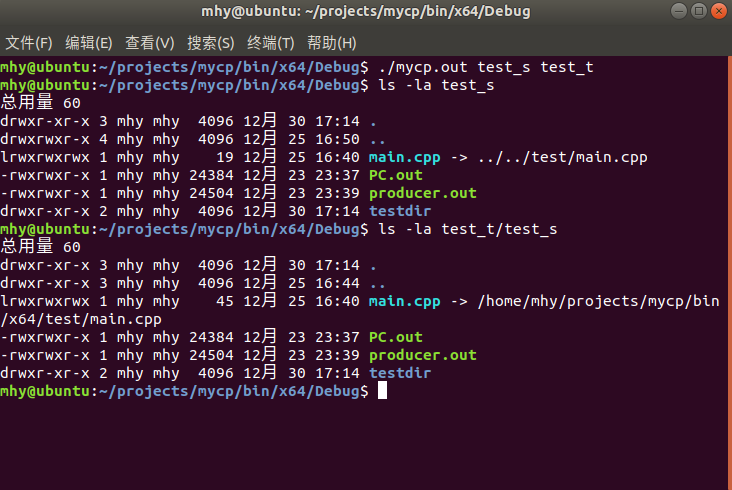


1. 实验结果
2. Windows





1. linux



1. **实验收获与体会**

通过本次实验，我进一步学习了在编程中如何处理文件，并且掌握了不同系统中对进程的不同处理方式。学习了linux下链接的概念以及在程序中如何处理。

此次实验中也遇到了不少问题：

1. 修改目录时间属性语句的位置。由于程序是递归实现，不能在目录创建时直接修改，因为后续的操作会在该目录下添加文件，新的操作会覆盖前面的修改，所以必须在程序递归处理完所有目录项后才可以修改其时间属性，除外，修改链接文件的时间属性时也遇到了麻烦，由于utime会默认修改链接文件所指向的文件的时间属性，所以不能用，查了好久才找到lutimes这个函数。相同的还有lchmod。
2. Linux下处理链接文件，由于用readlink得到的源文件路径是相对路径，复制后新的链接文件可能无法定位到真实的源文件，最终想到一个办法，直接用realpath得到链接文件所指向的文件的绝对路径，建立新的链接，就可以使复制后的的链接文件指向绝对路径。类似的问题还有，由于链接文件，在向cp函数传入源文件名前，不能直接用realpath去取绝对路径，必须手动将其变为绝对路径。