Linux下的类shell实现
实验报告

汇报人：张桓 CS 191220156

时间：2020年3月

类shell会实现

- cp (-r)file file
- cp -r dir1 dir2
- cp (-r)file dir
- cat file(…)
- wc (-c/-l/-w) file(…)
- cmp file1 file2
- man command
- sh file
- 命令纠错功能

目录

实验目的

实现自己的shell。这些命令分别是：cp，cmp，wc，cat、man，sh命令。这些命令的格式与功能应该与Linux下的一致。

把命令写成函数，执行不同命令时调用不同函数即可，模拟操作系统Shell的实现，通过编程来实现类似的功能

需求分析

#通过函数实现要求的命令，函数可以模拟shell命令的功能
#要与shell界面尽可能相似

*有输入提示，并能处理输入的一行行命令
*不断接受命令并执行直到输入退出命令
→主体部分利用while循环
→各个命令分别用函数实现
→对输入的命令有对应函数处理使其可以直接对应到功能函数

实验过程及结论

main函数主题采用while循环方式，cout出输入提示并不断fgets读取输入的原始命令字符串

处理原始命令通过operate函数，对字符串进行分割并通过判断分隔后的各个部分决定调用哪个功能函数：

这里主要利用strtok函数进行的字符串分隔查阅资料后得知strtok的相关用法，可按照给出的字符一段一段的将字符串分隔

问题：strtok函数会改变原来的字符串本身，实验过程中不了解该情况，他拆开之后还当作一个完整的字符串使用！导致错误

结论：若要继续使用先strcpy一个字符串备份

```
char *func  = strtok(str,"\t ");
```

```
char *file1=strtok(NULL,"\t ");
char *file2=strtok(NULL,"\t ");
char *th=strtok(NULL,"\t ");
```

实验过程及结论

处理分隔后的字符首先判断第一段确定功能区域，再在对应的区域进行预判断后调用**对应函数或报错**（即operate函数功能）

#cmp命令(cmp file1 file2)

问题：在于判断参数文件是否为目录并报错

```
struct stat st1;
stat(file1,&st1);
if(S_ISDIR(st1.st_mode))
{
        printf("cmp: %s: 是一个目录\n",file1);
        return ;
}
```

结论：查阅资料后利用stat结构体得到文件信息并通过S_ISDIR函数即可判断

同样在后面的cp等命令时也用到了类似内容诸如S_ISREG函数等等stat结构体的应用，不再赘述

实验过程及结论

#copy命令

问题：在于复制目录文件(cp -r dir1 dir2)时不知如何创建目录并且如何实现递归的创建即复制

结论：查阅资料后可以利用makedir函数，并且有类似于FILE文件指针的DIR目录指针来指向目录里的各个文件，利用dirent结构体可以获取目录里各个文件的详细信息，利用文件的**路径**实现在目录里递归复制，可以确定下一步是继续递归调用还是调用复制文件函数

*顺此思路添加了把文件复制到目录里的功能

```
struct dirent* filename;
while(filename=readdir(dp))//一个个读源目录里的文件错误或读完返回NULL
{
        char* path1=new char[1024];
        char* path2=new char[1024];
        memset(path1,0,sizeof(path1));
        memset(path2,0,sizeof(path2));
        if(whatend(dir1,'/'))//源目录路径结尾是/ 直接连接即可
                path1=strcat(path1,dir1);
        else//结尾不是/则需要添加
        {
                path1=strcat(path1,dir1);
                path1=strcat(path1,"/");
        }
        if(whatend(dir2,'/'))//目标目录路径结尾是/直接连接
                path2=strcat(path2,dir2);
        else//结尾不是/则需要添加
        {
                path2=strcat(path2,dir2);
                path2=strcat(path2,"/");
        }
        path1=strcat(path1,filename->d_name);//在dirent结构中的文件名称
        path2=strcat(path2,filename->d_name);

//如果目录里的这个文件还是个目录递归调用
//! 注意源目录里开头有. ..默认目录! 表示同级目录和上级目录! 不能递归
        if(ifdir(path1))
        {
                if(!whatend(path1,'.'))//是目录而且不是默认的目录
                copydir(path1,path2);
        }
        else//不是目录就是文件了
                copyfile(path1,path2);
}
```

实验过程及结论

#wc命令(wc (-l/-w/-c)file1 file2…)

问题：在于对任意多的参数文件可以一次输出各个文件的信息并且输出总和

结论：在之前只能输出一个文件信息的基础上，在operate部分分隔命令将任意多文件名记录在指针数组中遍历数组调用原函数

#man命令

问题：为实现shell中可以自动忽略大小写的功能

结论：查阅资料后发现有简单的库函数isupper tolower函数可以直接对字母大小写处理

实验过程及结论

#纠错功能

问题：在于何时纠错以及判断标准，穷举不是很现实

结论：在之前这里考虑输错命令时只修改在结尾多输了字母的情况，考虑到若判断字母个数比较相似度可能导致cp命令和cmp命令误判，所以只对多输字母进行纠错(correct函数)

并且在执行.sh文件时同样也有纠错功能可以自动纠正文件中错误的命令

操作示例

- 首先在命令行输入 make 命令成功后会出现 myshell可执行文件以及Function.o main.o文件

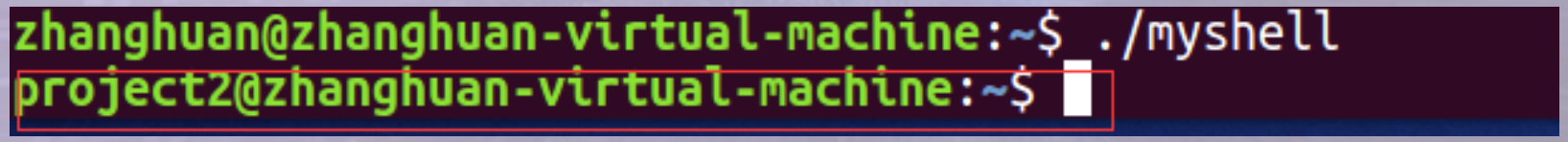
```
zhanghuan@zhanghuan-virtual-machine:~$ make
g++ -g -c Function.cpp -o Function.o
g++ -g -c main.cpp -o main.o
g++ define.h main.o Function.o -o myshell
zhanghuan@zhanghuan-virtual-machine:~$ ls
a.txt    cmp.txt  define.h      main.cpp  man.txt  公共的  图片  音乐
b.txt    cp.txt   Function.cpp  main.o    myshell  模板    文档  桌面
cat.txt  c.txt    Function.o    makefile  wc.txt   视频    下载
```

操作示例

- 再输入 ./myshell 命令即可进入类shell界面

用 project2@zhanghuan-virtual-machine:~$_ 作为输入提示

接下来您就可以利用本类shell进行需要的操作了，下文将对各命令逐一展开



操作示例

★输入注意：

在之后为您展示的输入命令的格式中各个部分都是用一个空格隔开的，如：

```
project2@zhanghuan-virtual-machine:~$ cmp a.txt b.txt
```

您可以同系统shell那样在不同部分中间隔很多空格，如：

```
project2@zhanghuan-virtual-machine:~$ cmp     a.txt      b.txt
```

同样可以被正确识别和执行

但是为保证命令的确实实现，需要正确隔开不同的部分，否则会有相应报错

操作示例

★输入注意：

同样您输入的参数位置并不只能同示例那样在紧跟在命令之后，如：

```
project2@zhanghuan-virtual-machine:~$ cp -r a.txt b.txt
project2@zhanghuan-virtual-machine:~$ wc -w a.txt b.txt
```

您可以同系统shell那样，把参数放在输入命令行的任意位置，正确隔开即可，如：

```
project2@zhanghuan-virtual-machine:~$ cp a.txt -r b.txt
project2@zhanghuan-virtual-machine:~$ wc a.txt b.txt -w
```

同样可以被正确识别并执行

操作示例

★输入注意：

如果您输入了本类shell不能实现的的命令，会有一下形式的报错

```
project2@zhanghuan-virtual-machine:~$ mkdir a
未找到 'mkdir' 相关命令，请您从(cp,cmp,wc,cat,man,sh,exit)中选择
```

且：在您输入命令时请不要使用左右箭头来移动光标，请使用退格键

操作示例

- cp (-r) file1 file2

  将文件file1复制到file2中，如果file2不存在则会创建，存在则会被重新覆盖

  ★注意：(-r)表示该命令语句同系统shell一样可有可无

  如果file1不存在或者file1是个目录会提示

  如果输入多个文件参数只会处理前两个

```
project2@zhanghuan-virtual-machine:~$ cp noexit.txt a.txt
cp: 无法获取'noexit.txt' 的文件状态(stat): 没有那个文件或目录
```

```
project2@zhanghuan-virtual-machine:~$ cp a a.txt
cp: 略过目录'a'
```

请确保参数-r输入正确，否则会有相应提示，如

```
project2@zhanghuan-virtual-machine:~$ cp -rr a.txt b.txt
cp: 输入错误 请检查您的参数是否正确
Try 'man cp' for more information.
```

操作示例

- cp (-r) file dir

  将文件file复制到dir中，会覆盖掉同名文件

  ★注意：(-r)表示该命令语句同系统shell一样可有可无

  确保dir是个已经存在的目录，否则会按照cp (-r) file1 file2处理

  如果file不存在会提示，多个文件参数只会处理前两个

```
project2@zhanghuan-virtual-machine:~$ cp no.txt a
cp: 无法获取'no.txt' 的文件状态(stat): 没有那个文件或目录
```

- cp -r dir1 dir2

  将目录dir1内容复制到dir2中，会覆盖掉dir2中的同名文件

  ★注意：该命令-r不可省略否则会按照之前的命令进行相应的报错

  同系统shell，若dir2原来不存在会创建一个新目录复制dir1中的**内容**，存在会将dir1**整个目录**复制到dir2中

  若dir1不存在会提示，多个文件参数只会处理前两个

```
project2@zhanghuan-virtual-machine:~$ cp -r m a
cp: 无法获取'm' 的文件状态(stat): 没有那个文件或目录
```

操作示例

- wc file…统计各个文件file的相关信息，并按 “行数 单词数 字节数” 的顺序输出
- 而wc (-c/-l/-w)【个数随意】 file1 命令则对应只输出 字节数 行数 单词数 的一个或多个格式如下：

```
project2@zhanghuan-virtual-machine:~$ wc -w a.txt b.txt c.txt
1 a.txt
1 b.txt
wc: c.txt: 没有那个文件或目录
 2 总用量
```

```
project2@zhanghuan-virtual-machine:~$ wc -w -c a.txt a b.txt
 1 5 a.txt
wc: a: 是一个目录
 1 5 b.txt
 2 10 总用量
```

★注意：

如果file是一个目录或者不存在的话就会报错

同系统shell您可以在后面接多个文件参数，这样在最后会输出总用量

操作示例

★注意：

wc命令有三个参数可以选择，且可以跟随多个文件参数

同系统shell您在输入参数时可以重复输入，且参数位置很自由

格式如：-wll -wlc -lll -lcw 等等都可以正确识别并执行

但要确保您输入的参数在w l c之中若输入其他会有相应提示

```
project2@zhanghuan-virtual-machine:~$ wc -ccc a.txt
5 a.txt
```

```
project2@zhanghuan-virtual-machine:~$ wc -wc a.txt -lll b.txt
 1  1 5 a.txt
 1  1 5 b.txt
 2  2 10 总用量
```

```
project2@zhanghuan-virtual-machine:~$ wc -wls a.txt
wc: 无效选项 -- s
Try 'man wc' for more information.
```

操作示例

- cmp file1 file2　对比文件file1和file2的内容，并输出第一个不同处的行数和对应行的字节数，或报告其中一个文件已经结束如右图

★注意：

该命令会依次检测file1和file2，如果file1或file2为目录或者不存在就会报错

如果参数只有一个或者没有会产生右图样式的报错

如果参数有多与2个会有对应报错

```
project2@zhanghuan-virtual-machine:~$ cmp a.txt b.txt
a.txt b.txt 不同: 第 18 字节, 第 3 行
```

```
project2@zhanghuan-virtual-machine:~$ cmp a.txt b.txt
cmp: b.txt 已结束
```

```
zhanghuan@zhanghuan-virtual-machine:~$ cmp a b
cmp: a: 是一个目录
zhanghuan@zhanghuan-virtual-machine:~$ cmp noexit.txt b
cmp: noexit.txt: 没有那个文件或目录
zhanghuan@zhanghuan-virtual-machine:~$ cmp a.txt a
cmp: a: 是一个目录
zhanghuan@zhanghuan-virtual-machine:~$ cmp a.txt noexit.txt
cmp: noexit.txt: 没有那个文件或目录
```

```
project2@zhanghuan-virtual-machine:~$ cmp a.txt
cmp: (null): 没有那个文件或目录
```

```
project2@zhanghuan-virtual-machine:~$ cmp a.txt b.txt c.txt
cmp: invalid --ignore-initial value 'c.txt'
cmp: Try 'man cmp' for more information.
```

操作示例

- man command 展示命令的参考手册并输出到屏幕上

★注意：

只能展示：cp cat wc cmp man sh 六个命令对应的参考手册，如果输入了其他的命令会报错

如果输入了多个参数只会处理第一个参数

★同系统shell该命令可以实现对大写字母的自动识别，比如输入：man CAT/man cAt/man Cat等最后都会自动为您输出cat命令的参考手册

```
project2@zhanghuan-virtual-machine:~$ man cat
NAME
       cat - concatenate files and print on the standard output

SYNOPSIS
       cat [FILE]
```

```
project2@zhanghuan-virtual-machine:~$ man caa
没有 caa 的手册页条目
```

```
project2@zhanghuan-virtual-machine:~$ man CAT
NAME
       cat - concatenate files and print on the standard output

SYNOPSIS
       cat [FILE]
```

操作示例

- sh file 执行脚本文件，一行一行读取文件中的语句并执行

★注意：

如果文件不存在或者在文件中读到的语句无法执行会产生相应提示

如果参数有多个会忽略掉第二个及以后的参数

```
zhanghuan@zhanghuan-virtual-machine:~$ sh mmm
sh: 0: Can't open mmm
```

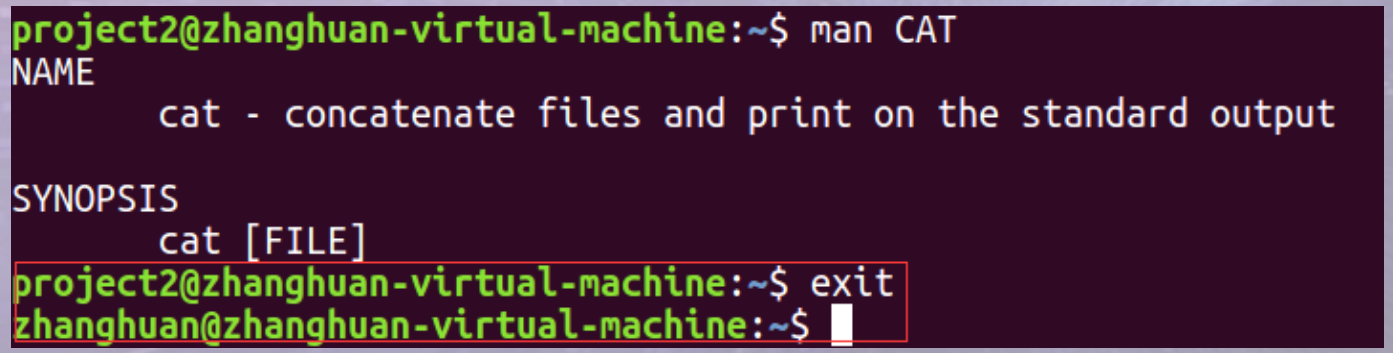
```
project2@zhanghuan-virtual-machine:~$ sh a.txt
a.txt: 1: a.txt: asda: not found
```

```
project2@zhanghuan-virtual-machine:~$ sh a.sh b.sh
asda
asda
NAME
       cp - copy files or directories

SYNOPSIS
       cp (-r) file1[SOURCE] file2[DEST]
       cp (-r) file[SOURCE] dir[DEST]
       cp -r dir1[SOURCE] dir2[DEST]
```

操作示例

- 进行完操作后退出操作为正确输入：exit 即可从myshell中退出返回到原始shell界面



演 示 完 毕　感 谢 观 看

汇 报 人： 张 桓