操作系统实践

10152130137 汪贻俊

Project 2

Introduction

本次实验是实现一个简易版Linux下的shell,主要是使用Linux下的一些系统调用函数，来实现shell的命令行处理功能，同时还有就是深入理解管道和重定向后实现这两个重要的功能。

Solution

1. 程序框架(myproc.c)

就是首先接受一个命令行，然后进行分析，之后就是看属于哪一种类型的命令，然后命令的类型执行就好了。

2. 命令行读入/批处理下从文件读入(mysh.c)

通过main函数的命令行参数来区别是命令输入，还是批处理。

3. 命令预处理与分析（parse_command.c）

预处理是消除空格划分命令参数，然后命令的分析就是确定命令的类型根据相应的特殊命令标识。

4. Builtin 命令（builtin.c）

Builtin 命令有 exit,pwd, cd, wait, wait 命令和 Background Job 一起介绍，所以就剩下三个命令：

exit: 调用 exit(0)

pwd: 调用 getcwd(NULL, 0)

cd: 先判断有没有参数，没有过的话就是用 getenv("HOME") 获取，否则使用 chdir()函数改变所在目录

5. Background Job(myproc.c)

当出现后台作业时，首先将后台子进程的 pid 记录下来，然后就不用等待子进程结束直接进入下一个命令，而遇到 wait 命令时，要将等待记录的所有的子进程结束后再进入下一条命令。

6. Redirection(myproc.c)

使用 open 获取文件描述符，然后使用 dup2()函数重定向输出即可

7. Pipe(mypipe.c)

首先得确定 pipe 管道要建立多少个子命令，然后首先创建相应数量的管道，然后将前一个命令的输出重定向到一个管道的写端口，之后将后一个命令的输入重定向到管道的读端口，依次将所有命令全部连接起来即可。

8. 其余的命令(myproc.c)

fork()一个子进程，在子进程中调用 execvp()执行命令即可。