

上机作业 4: Shell 脚本程序模拟

088 于海鑫

2017211240

name1e5s@bupt.edu.cn

版本: 8

更新: May 26, 2020

1 题目要求

使用 `fork()`, `exec()`, `dup2()`, `pipe()`, `open()`, `wait()` 等系统调用编写 C 语言程序完成与下列 Shell 命令等价的功能。

```
1 grep -v usr < /etc/passwd | wc -l > r.txt; cat r.txt
```

(提示: 为简化编程, 不需要用 `strtok` 断词, 直接用 `execvp` 实现能达到 Shell 命令相同功能的程序即可)

例如: `execvp("grep", "grep", "-v", "usr", 0);`

2 作业答案

本次作业我们只需使用一些简单的函数调用对 Shell 命令进行模拟即可, 主要是模拟管道这一操作。代码如下:

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <fcntl.h>
4 #include <unistd.h>
5 #include <sys/wait.h>
```

```

6
7 int main(void) {
8     int fd[2];
9     pipe(fd);
10
11     if(fork() == 0) {
12         close(0);
13         close(1);
14         close(fd[0]);
15         int passwd_fd = open("/etc/passwd", O_RDONLY);
16         dup2(passwd_fd, 0);
17         dup2(fd[1], 1);
18         execlp ("grep", "grep", "-v", "usr", NULL);
19     }
20     if(fork() == 0) {
21         close(0);
22         close(1);
23         close(fd[1]);
24         int r_fd = open("r.txt", O_WRONLY | O_CREAT, 0777);
25         dup2(r_fd, 1);
26         dup2(fd[0], 0);
27         execlp("wc", "wc", "-l", NULL);
28     }
29     close(fd[0]);
30     close(fd[1]);
31     wait(NULL);
32     wait(NULL);
33     execlp("cat", "cat", "r.txt", NULL);
34 }

```

在这里我们使用 `pipe` 初始化管道，之后使用 `fork` 创建子进程，然后使用 `dup2` 替换掉子进程的输入/输出，最后分别调用 `execlp` 替换进程映像。这样即可模拟我上面的函数。

测试结果：

```

1 name1e5s@sumeru:~$ gcc t.c
2 name1e5s@sumeru:~$ ./a.out
3 8
4 name1e5s@sumeru:~$ grep -v usr < /etc/passwd | wc -l > r.txt; cat r.

```

5	txt
8	

符合要求。