

第四次上机作业

姓名：马嘉骥，学号：2018211149

班级：2018211303，班内序号：15

题目

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。

```
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execlp实现能达到shell命令相同功能的程序即可)，例如：execlp("grep", "grep", "-v", "usr", 0);

源代码

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <unistd.h>
4  #include <fcntl.h>
5  #include <sys/stat.h>
6  #include <sys/wait.h>
7
8  //grep -v usr < /etc/passwd | wc -l > r.txt; cat r.txt
9  //result: 5
10
11 int main()
12 {
13     int pipefd[2], pid, status;
14     pipe(pipefd); //new pipeline
15     if ((pid = fork()) == -1)
16     {
17         perror("fork");
18         exit(1);
19     }
20     else if (pid == 0)
21     { //parent open file
22         int fd0 = open("/etc/passwd", O_RDONLY);
23         if (fd0 != -1)
24         {
25             //get filefd success
26             fflush(stdout); //flush buffer
27             dup2(fd0, STDIN_FILENO); //redirect STDIN to fd0
28             close(fd0);
29         }
30         dup2(pipefd[1], STDOUT_FILENO); //redirect STDOUT to pipefd1
31         close(pipefd[0]);
32         close(pipefd[1]);
33         execlp("grep", "grep", "-v", "usr", NULL); //run command, input from
34         fd0, output to pipefd1
35         exit(1);
36     }
```

```

35     if ((pid = fork()) == -1)
36     {
37         perror("fork");
38         exit(1);
39     }
40     else if (pid == 0)
41     {
42         //create r.txt if not exist, use -rw-rw-rw-
43         int fd1 = open("r.txt", O_CREAT | O_WRONLY, 0666);
44         if (fd1 != -1)
45         {
46             //get filefd success
47             dup2(fd1, STDOUT_FILENO); //redirect STDOUT to fd1
48             close(fd1);
49         }
50         dup2(pipefd[0], STDIN_FILENO); //redirect STDIN to pipefd0
51         close(pipefd[0]);
52         close(pipefd[1]);
53         execlp("wc", "wc", "-l", NULL); //run command, input from pipefd0,
54         output to fd1
55         exit(1);
56     }
57     close(pipefd[0]);
58     close(pipefd[1]);
59     wait(&status);
60     wait(&status); //wait for 2 children
61     execlp("cat", "cat", "r.txt", NULL); //cat to STDOUT
62     return 0;
63 }

```

运行结果

```

1 root@RIDD-DESKTOP:/mnt/c/Users/Ridd/GitRepos/ShabbyToys/linux/pipeline# grep
2 -v usr < /etc/passwd | wc -l > r.txt; cat r.txt
3 5
4 root@RIDD-DESKTOP:/mnt/c/Users/Ridd/GitRepos/ShabbyToys/linux/pipeline# gcc
5 pipe.c -o pipe
6 root@RIDD-DESKTOP:/mnt/c/Users/Ridd/GitRepos/ShabbyToys/linux/pipeline#
7 ./pipe
8 5
9 root@RIDD-DESKTOP:/mnt/c/Users/Ridd/GitRepos/ShabbyToys/linux/pipeline#

```

总结

本C程序使用了 fork(), exec(), dup2(), pipe(), open(), wait() 等系统调用, 创建管道在子进程间通信, 使用dup2重定向标准输入输出, 实现了 `grep -v usr < /etc/passwd | wc -l > r.txt; cat r.txt` 等价的功能。