

OS 2016

Homework5: Shell

(Due date: 2017/01/12 23:59:59)

Requirements

- Implement your own shell
 1. Can execute external programs by commands
 - Either by relative or absolute pathnames
 - The command-line arguments should be passed to external programs
 2. Two built-in commands
 - For changing working directory, i.e., *cd*
 - For exiting current shell, i.e., *exit*
 3. Support pipeline execution (pipe) and I/O redirection

Implement Shell

- Print prompt – “0\$”
- Get a command line
- Parse the command
- Execute the command
 - a built-in command, i.e., cd and exit
 - not a built-in command
 - I/O redirection
 - Pipeline execution

```
os@os:/home/os/hw5$ ./shell
0$ pwd
/home/os/hw5
0$ ls -a
.  ..  Hw5.c  shell
0$ cd /home
0$ pwd
/home
0$ ls -a
.  ..  os
0$ exit
os@os:/home/os/hw5$
```

Pipeline Execution and I/O Redirection

- Support “|” so that user can do pipeline execution

```
0$ ls | grep a
abc bac cab
0$ ls -l | cut -c 1-10 | sort | uniq | wc -l
13
0$
```

- Support “>” and “<” so that user can do I/O redirection

```
0$ echo -e abc\\nccc\\nabe > 123.txt
0$ grep a < 123.txt
abc
abe
0$
```

Input Format

- Only three operators: |, > and <
 - No quotation marks(" or '), e.g., "string", 'string'
- Use single space character to separate labels(tokens)
 - Tips: *strtok_r* or *strsep* to extract labels(tokens) from string
- Input redirection (<) only show up after first program
- Output redirection (>) only show up after last program

```
0$ cat < 123.txt | sort | uniq | wc -l > 456.txt
0$ cat 456.txt
4
0$ cat > 789.txt < 456.txt
0$ cat 789.txt
4
```

Bonus

- Support jobs control
 - Ctrl + Z : stop foreground process
 - Ctrl + C : terminate foreground process
 - Built-in commands:
 - jobs : List background processes
 - bg : Put the process in background
 - fg : Put the process in foreground
 - Execute process in background with an **&** at the end of the command line
- Lists of commands, separated by ";" "&&" "||"
- Maintain a history of commands previously executed
 - Support arrow key up(↑) and down(↓)
- Other functions support by common shell(e.g., sh, csh)

Reference

- Linux Manual Page
 - [getline](#)
 - [chdir](#)
 - [fork](#)
 - [exec series](#)
 - [wait, waitpid](#)
 - [pipe](#)
 - [dup2](#)
 - [open](#)
 - [strtok_r](#), [strsep](#)
 - [perror](#)
- [GNU Libc Manual Page – Implementing a Shell](#)