Project 7
Simple Shell Interface[*]

Jiashuo Wang
5100309436

May 12, 2013

Instructor: Ling Gong

I. Objective

Write a program that can imitate a simple shell and execute some basic commands. You can see the detail in the end of chapter 3 of *OPERATING SYSTEM CONCEPTS WITH JAVA(Seventh Edition)*, page 127.

II. Algorithm

II.1 Simple Shell

The key is to create an external process to let the Linux operating system to help me execute most of the command based on its command document in the Linux. So this program can only be executed in the OS like Linux, not Windows because the command in Linux is in Command documents. This kind of commands is like *ls*, *cat*, *pwd*.

Some command, however, is not in the system, which means we have to implement it by ourselves. These are like *cd*, *history*, *exit*.

1. ewfdww
    (a) sasa
        i. dsds
            A. ds
            B. fe
            C. fe
        ii. dsd
        iii. as
    (b) sas
    (c) dsd
2. wdsd
3. dssds

[*] Designed by LaTeX

II.2 *cd* Implementation

To implement *cd*, we have to take several situation into account. For example:

$$cd \;\; .. \tag{1}$$

$$cd \;\; /usr/bin \tag{2}$$

$$cd \;\; src \tag{3}$$

So first, we have to get the current directory with *getProperty()*. For (1), we can get the directory by wiping out the last file. But if the current directory has already been the root one, the directory has to be "/". For (2)(3), we have to judge whether the directory is legal by using *isDirectory*. Then for (2), we can replace the directory by the content after the command *cd*. For (3), we can add the content after the original directory.

II.3 *history* Implementation

The *history* function is designed to imitate that in Linux *Terminal*.

Every time the user input something, the *ArrayList* will help to store them if the input command is not the same as the last one. It is a dynamic array. So when you input *history* command, the historical commands will print.

When you input the command *!!*, the last command will be executed automatically. Mean while, if you input not only *!!*, but also something else, the rest one will be add to the historical command automatically. For example, if the last command is *ls*,and now you input *!! -l*, the command to be executed will be *ls -l*.

When you input the command *!* and a number, that means you want to execute the historical command with the index of that number. And that command will be executed automatically.

II.4 More details

If the command one input is not Linux command, an *IOException* will be threw. And the shell should tell the user the error message and continue the shell. And when the command is *!!* or like *!4*, the limit of the history will be considered.

And if the user inputs the *exit*, the endless loop should be broken to terminate the program.

III. Results and Conclusions

III.1 Environment

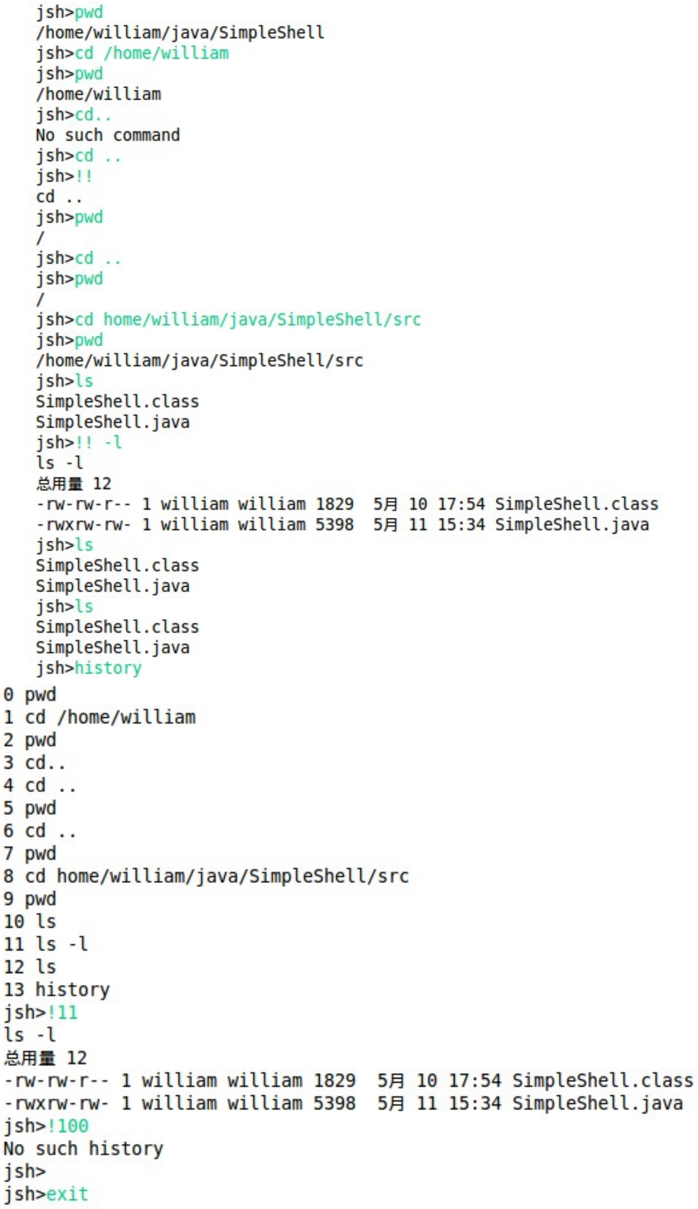
- Ubuntu 12.10
- Eclipse jee juno SR2 for Linux

III.2 Screenshots of the result

Use JVM to compile and execute the program in Figure 1.

III.3 Thoughts

It is amazing to make a program just like an OS, which is quite interesting.



```
jsh>pwd
/home/william/java/SimpleShell
jsh>cd /home/william
jsh>pwd
/home/william
jsh>cd..
No such command
jsh>cd ..
jsh>!!
cd ..
jsh>pwd
/
jsh>cd ..
jsh>pwd
/
jsh>cd home/william/java/SimpleShell/src
jsh>pwd
/home/william/java/SimpleShell/src
jsh>ls
SimpleShell.class
SimpleShell.java
jsh>!! -l
ls -l
总用量 12
-rw-rw-r-- 1 william william 1829  5月 10 17:54 SimpleShell.class
-rwxrw-rw- 1 william william 5398  5月 11 15:34 SimpleShell.java
jsh>ls
SimpleShell.class
SimpleShell.java
jsh>ls
SimpleShell.class
SimpleShell.java
jsh>history
0 pwd
1 cd /home/william
2 pwd
3 cd..
4 cd ..
5 pwd
6 cd ..
7 pwd
8 cd home/william/java/SimpleShell/src
9 pwd
10 ls
11 ls -l
12 ls
13 history
jsh>!11
ls -l
总用量 12
-rw-rw-r-- 1 william william 1829  5月 10 17:54 SimpleShell.class
-rwxrw-rw- 1 william william 5398  5月 11 15:34 SimpleShell.java
jsh>!100
No such history
jsh>
jsh>exit
```

Figure 1: *Screenshots of Matrix Multiplication*