

CSCD 340

Lab 8

We will develop a shell and then add specific features. The goal of this assignment is to write a shell (called *mysh* which will stand for “my shell”). Specifically, you are to write the framework for a Linux shell using the C programming language. Once the shell is executing it will prompt the user with the “?: ” prompt. The user now may issue any command line Linux command. I have provided the loop for *mysh*. You can’t change my code; however, you will need to add your code.

Here is the general battle plan

- 1) In lab you wrote a `forkIt` function that executed a valid Linux command or did nothing if the command was invalid. Enhance the `forkIt` function (if you still want to call it that) to handle the `&`. Meaning I should be able to execute a command that will run in the background. If a command is invalid print the appropriate error message to the screen.
- 2) In lab you wrote a `pipeIt` function. You will need to enhance the `pipeIt` function to handle multiple pipes. You should be able to handle `ls -l | sort | wc -w`
- 3) Implement a path extension feature, e.g., `PATH=$PATH:/data/myprogs`, at the command line.
- 4) The *mysh* shell is to read input from a file called *.myshrc* and execute each line in this file as a command. When the program is run *.myshrc* will be read. This makes the *.myshrc* file a batch file. (BASH reads from *.bashrc*)
- 5) Implement a path extension feature where the user can have predefined paths in the *.myshrc* file. For example in *.myshrc* the user has already defined the path to be `PATH=$PATH:/data/myprogs`. This will be preloaded.
- 6) Implement a "history" mechanism which will allow to scroll though the last *N* shell commands. Your *.myshrc* will contain two environment variables `HISTCOUNT` and `HISTFILECOUNT`. (BASH has `HISTSIZE` and `HISTFILESIZE`). It is up to you to determine how `HISTSIZE` and `HISTFILESIZE` are used by BASH. Your shell will history mechanism will mimic BASH including each environment variable. At a minimum you should be able to recall commands by number, for example **!513**, or **!!** to re-execute the last command. Similar to BASH the history is stored in a file called *.mysh_history*.
- 7) Implement redirection (`<` `>` `>>`). You should be able to redirect stdin and/or stdout.
- 8) Implement the `cd` command.

NOTES:

1. Your shell has to be BASH syntax compatible. If you divert from this, your shell will not pass the input used for testing! If it works in BASH it should work in yours.
2. Design your code so it is **clean** and **modular**. **Don't wait to start** you won't finish if you don't chip away at this. Don't try to do everything in one sitting. Pick and choose and work on one topic.
3. Draw a design document before you start coding. You need to understand how all the pieces work together before you start coding. If you don't you will quickly start breaking things by trying to add the next item.
4. Your linked list will come in really handy, multiple times.
5. You may have very few global variables.
6. History will act like the history command of BASH

GENERAL FLOW

- The program starts up and checks for:
 - *.myshrc* – if it exists reads in the variables – file format is below – doesn't exist then default HISTCOUNT and HISTFILECOUNT are used
 - *.mysh_history* is loaded if it exists
 - prompt is displayed
 - each command is handled appropriately
 - upon exit *.mysh_history* is written from the executed commands

. myshrc file format – if exists is guaranteed to be similar to below

```
HISTCOUNT=value  
HISTFILECOUNT=value
```

```
PATH=$PATH:/data/myprogs
```

TO TURN IN

- The source code of your program. Name your file that contains main cscd340lab8.c
- A Makefile so your code can easily be compiled with the target of mysh
- Some output captures to illustrate you shell is working. This does not need to be extensive. We will test your code this is just to illustrate you did your own testing. Name this cscd340lab8out.txt

You will submit a zip file named your last name first letter of your first name lab8.zip. (Example steinerslab8.zip)