CMPE142 : Operating Systems
Assignment 1: Creating a UNIX shell
Version 1

Due: Mar 15 2019 Before Midnight

In this assignment you will create a simple UNIX shell. A shell, or *command interpreter*, is what you are already using today when typing commands into a terminal window. The shell is responsible for parsing the command typed by the user and executing the command requested. This assignment is worth up to 20 points and may be done in groups of up to **three people max.** Each team member can receive up to 20 points. It is expected that groups of more than one student will find an equitable way to distribute the work outlined in this assignment. Groups of fewer than three people will still be expected to do the same assignment (eg, no skipping parts if you are working solo).

Prerequisites

- You will need a Linux development environment. We outlined how to create this in class.
- A github repository to store your code (see below)

The Assignment

The assignment requirements are outlined at the following URL, which has been created by the class textbook author:

https://github.com/remzi-arpacidusseau/ostep-projects/tree/master/processes-shell

Start by creating a new github repository for your code. You can sign up for a free github account online if you don't have one already. Each team member should have their own account, but it does not matter which account owns the repository for this assignment. The repository can be either public or private.

Next, develop the basic framework – reading commands and executing processes. After you have that working, you can split up the remaining work among your team members. It is up to each team to agree on an equitable division of work among the team members.

Teams are **strongly** urged to "commit early, commit often" - this means as soon as you have something working, push your changes to your team's repository. This ensures that everyone has access to the latest code, and also serves as a backup in case your copy of the code gets messed up.

The author calls his shell 'wish' for "Wisconsin Shell" since that's where he teaches. You may name your shell whatever you want (see question 3 at the end of this document though).

Functionality to Implement

The functionality to implement is given in the URL above.

Testing

I will be running 15 randomly selected command lines in your shell. Some will be simple things like "ls" and others will be more complex. The commands will be similar to the sample test commands located at:

https://github.com/remzi-arpacidusseau/ostep-projects/tree/master/processes-shell/tests

In the above URL, you will see four files for each test. One file ends in ".desc" and is the description of the test. The ".in" and ".out" specify the input to the test (eg, the command like you would type) and the expected output. The ".err" file contains the text you would see if the test failed. You are urged to try these tests out on your implementation and see if you pass them all.

Submission and Grading

Your team will be awarded 1 point for each successful command test. Each team member will be awarded up to 5 points based on the answers to the questions below.

On or before the due date, send answers to the questions below via E-Mail or Canvas, and grant me access to your team's github repository (my github ID is 'mlarkin2015'). **Each team member must answer the questions separately.**

The answers to the questions below shall be made via email to the email addresses listed in the class syllabus/green sheet, or via Canvas. DO NOT WAIT UNTIL LATE ON THE DUE DATE, as email server lags or delays may result in a late submission. This is one area that I am extremely picky with – even 1 second late will result in a zero score for that part of the assignment.

I will be comparing all submissions to ensure no collaboration has taken place. Make sure you do not copy another group's work. If you copy another group's work, members of both groups will receive an F in the class and be reported to the department chair for disciplinary action. If you are working in a group, make sure your partners do not copy another group's work without your knowledge, as all group members will be penalized if cheating is found.

Make sure your final changes are pushed to github before midnight of the due date. I will be cloning your repository "as of" midnight on that date. This means any changes made afterward will not be considered.

Questions

1. Provide the name of the github repository (including the owning account, eg "mlarkin2015/142-assignment-1")
2. Provide instructions I need to follow to build your shell.
3. What is the name of your shell? (Eg, what do I type to invoke it?)
4. For each member in your team, provide 1 paragraph detailing what parts of the lab that member implemented / researched. This may seem obvious but this is the way I use to ensure that each team member was actually contributing usefully to the team. If you worked on the assignment by yourself, you can just send me an email saying "I did the project by myself", and that will be sufficient.