Project 1

Nathan Jarus

March 11, 2016

Introduction

This is the first of two projects in this class. The purpose of these projects is for you to explore a topic we have covered in lab in more detail. This will require you to do some research on your own; however, if you get stuck, please ask me for help. You should choose ONE of the following projects, complete it, and put your results in the git repository for this project. Please make a note of which project you have chosen in the `README.md` file.

If you would like to do another project that is not on this list (but is related to the topics we have discussed so far in class), please ask me before Friday, March 11.

For each project, you must submit a text file containing answers to questions, as well as any configuration files or scripts you wrote for the assignment. Your submission for this project should be similar to the lab assignments you have submitted so far.

The project is due at noon on March 25, 2016.

Project A: Text Editors and IDEs

For this project, you will get more friendly with at least one text editor or IDE. (Needless to say, your editor will need to be something a little more complicated than jpico, nano, or notepad++. Use one of the ones presented in class, or, if you have another you would like to use, please ask me first.)

1. Set up and use at least five plugins. List which plugins you chose, what they do, and how to use them.

2. Find at least ten new editor features. List what each feature does and how to use it. (Hint: google 'vim shortcuts' or the like.)

Project B: Shell Scripting

For this project, you will discover some fancy shell scripting features.

1. What is command substitution? Write at least two example shell scripts that use it. (Hint: `bc` will do non-integer math.)

2. Read the job control section of the week 2 slides.

   (a) What happens to paused or background jobs when you log out?

   (b) What does the command `disown` do?

   (c) What does the command `wait` do? Write a shell script that uses `wait` on at least two jobs (you can use `sleep` to make processes that run for a while).

Project C: Version Control

For this project, you will get to experiment with some more complicated git features.

1. `git rebase` (Hint: make a test repository to experiment with! Don't try it out in the repository for this project.)

   (a) Make two branches with commits on each. Instead of merging one branch into another, rebase it instead. What does the commit graph look like, and how does it differ from a merge?

   (b) Check `git log` before and after a rebase. What changes about the commits you have rebased?

   (c) Perform an interactive rebase (`-i`). What things can you choose to do with each commit? Try some of them out (and document what you did).

2. What does `git stash` do?

3. How can you use `git checkout` to get the contents of a file from a specific commit?

Project D: Regular Expressions

Write a program (in C++ or another language of your choice) that, given an HTML file, does the following:

1. Prints the text of every hyperlink (`<a>` tag) and the URL it links to.

2. Prints the alt-text of every image (`<img>` tag) and the URL of the image.

Hints:

- `wget` or `curl` can download webpages for you.

- Use regular expressions! ~~(Recall, C++ 11 has a regular expression library.)~~ G++ 4.8.4 doesn't have complete support for C++ 11 regexes. However, the boost library comes with a regex library that is practically the same. Note that when using boost, you must compile your code with `g++ *.cpp -lboost_regex` to include the boost library.