Comp161 - Lab 1 and Homework 1

Spring 2016

Your first homework and lab assignments get you moving with the Linux command line interface[1] and the GNU Emacs text editor. You'll be using these tools on an almost daily basis for this class and it is vital that you get comfortable with them ASAP. The best way to do this is by using them. **A lot**. To get you going you'll be working through two tutorials:

1. William Shotts, *Learning the Shell* from `http://linuxcommand.org`.
2. The Emacs built-in tutorial. You can launch the Emacs tutorial from within Emacs by using the command *C-h t*. Emacs itself is launched from the CLI with the command *emacs*.

Be sure to run all the commands Shotts shows you as he shows them to you.[2]

[1] CLI

[2] Some commands won't work because you lack sufficient privileges or the system doesn't support them

Lab 1

For your first lab your goal is to work through some of the Shotts tutorial and some of the the Emacs tutorial.[3]

By the end of lab you must:

1. Create a folder in your home directory named *comp161* and in that folder create a folder named *lab1*.
2. Use emacs to create a file named lab1report.txt. This file should be located in the folder *lab1* you created in the previous step. In the file respond to the following questions[4]:
   (a) What did you do over Winter break?
   (b) Have you ever used Linux before? If so, give a brief explanation of your history with Linux.
   (c) Have you ever worked with the Linux or Windows CLI before? If so, give a brief explanation of your history with the CLI. If not, what is your initial reaction to working with Linux and the CLI?

   Don't go over board with your responses, I just want to learn a bit about your background. When you're done *use the handin program*[5] *to submit the lab1 folder*.

[3] Be certain you can get in and out of the CLI, EMACS, and the EMACS tutorial so that you can complete your homework outside of lab, and be certain to review the homework assignment before you get going on the lab as you'll want to work on that as you progress through the tutorials.

[4] Don't forget to put your name on the file!

[5] see below

Homework 1

Due at the start of lab on Wednesday 1/20

For homework you'll need to complete the remainder of the Shotts tutorial and the Emacs tutorial. My way of checking that you've done this is by having you show me two things:

1. A shell[6] reference sheet *of your own making* that allows you to quickly remind yourself of the commands covered in Shotts' tutorial. The sheet should be your own work. You can copy the design of sheets found online, but you must create the sheet from scratch. It may be handwritten or typed. You're free to add anything else you want to this[7]. I count something like 40+ commands in Shotts' tutorial. Some are given to you in passing. Others are discussed at length. *I'm looking for your reference to have some organization to it*[8]. *Hastily thrown together notes with little to know organization will likely receive a 1 for the assignment.* T
2. The Emacs reference sheet with all the commands discussed in the Emacs Tutorial highlighted. This means you need to go to the course website, find the sources document, follow the link to the Emacs reference sheet, and print the sheet.

[6] bash really

[7] like things from the resources given below

[8] I recommend something like this: `http://www3.uah.es/clima/staff/gianni/doc/practicas/Extra/bash_reference_sheet.pdf`

You really need to commit to learning these tools, go beyond the assignments and really see what you can do with this stuff[9]. The CLI has systems in place to help you work quickly and efficiently[10]. Every year there are students that try to get these tutorials done as fast as possible and don't really dig into the material in them. These students typically never really learn to work quickly and efficiently with these tools and spend much of their time frustrated. So, take your time with the tutorials and build a really usable shell reference. You'll thank yourself later.

[9] I won't force you to do this but I will pester you about it if you don't

[10] be on the lookout for auto-completion and command history!

Emacs

Emacs is the text editor we'll be learning in this class. We've discussed what you need for these assignments in class, but here's a quick reminder. Commands usually require you to combine some keys with the *ctrl*[11] key or the meta key[12]. For example, the command to close Emacs is written *C-x C-c*. That means, "press and hold ctrl then x, then release them, then press and hold ctrl then c, and release them." It should feel like your rolling through keys starting with ctrl. If you're familiar with the windows command *ctrl-alt-del*, then you know what I'm talking about.

[11] shown as **C** on the sheet

[12] Shown as M. See below.

If you're at the CLI, you need two things really:

- *To launch emacs*: emacs
- *To open/create a file with emacs:* emacs *filename*

Once you're in Emacs you'll need at least these three emacs commands.

- *To start the tutorial*: C-h t
- *Save current file:* C-x C-s
- *Close Emacs:* C-x C-c

The tutorial will walk you through a host of other essential of Emacs commands. Like the CLI, you can fight the Emacs way of doing things or you can buy-in and take the time to learn and use the commands. When you do buy-in, you'll find that Emacs is insanely powerful and will save you a lot of time and frustration down the line. Professionals use it for a reason.

Meta Key

If you're on a linux or windows machine, then you have an *alt* key. That's your meta key. So commands like *M-b* are telling you to press and hold alt then b, then release both. If, however, you're on a Mac, you lack and alt key. You have two options[13]: use the *Esc* key or tell your terminal to use *option* as the meta key. If you go the route of *esc*, then I don't believe you hold the the key down[14].

[13] `http://stackoverflow.com/a/3566557/1042494`

[14] I could be wrong about that.

Handin

The *handin* program is a shell script that deposits files into a directory where the instructor can then collect them with some other scripts. It's how you'll be submitting most of the work for this class[15]. The command *handin -h* displays the help text for handin, which in turn tells you everything you need to know about using it to manage the submission of your work. *Read the handin help text to figure out how to submit your lab assignment.* The assignment designation for labs will always be something like *lab1* or *lab7*. This week is, of course, **lab1**. The course designation in this class is always *comp161*. For this lab, you could simply submit the one text file, lab1report.txt, but all other assignments will make use of multiple files and require the submission of a whole directory. So, to practice, you'll submit a directory of files with *handin* this week.

[15] you should add it to your list of commands on your reference

Other Sources

Along the lines of really committing to these tools, there are a few excellent additional resources I want to point out to you. Zed Shaw is a programmer that writes tutorials/online classes. His CLI course is

great if you really just want to drill the commands into your fingertips. It's worth checking out and it's free.

- Shaw, Zed. *The Command Line Crash Course: Controlling Your Computer From The Terminal*. Dec 2011. `http://cli.learncodethehardway.org/book/`

Additionally, Eric Nodwell's quick tutorial is nice because it really just gets down to the stuff you use most often.

- Nodwell, Eric. *Introduction to Commandline Linux*. 2003. `http://www.phas.ubc.ca/~mbelab/computer/linux-intro/html/`

Finally, William Shotts' tutorial continues on to introduce you to shell scripts. It's worth your time. Furthermore, all his online material is drawn form his book. The book is free and worth downloading or even buying `http://linuxcommand.org/tlcl.php`.

The built-in Emacs tutorial is great. But if you want another perspective on Emacs, check out this write-up.

- Wacelna, Keith. *A Tutorial Introduction to GNU Emacs*. 2009. `http://www2.lib.uchicago.edu/keith/tcl-course/emacs-tutorial.html`