CS246 Lab Notes #1 Unix and Emacs
Updated Jan 27, 2016

Part 1: (Choose a **driver**, who will login and type, and a **navigator** who will actively observe and comment)
The UNIX operating system consists of three parts: the kernel, the shell and the programs.
The kernel is the heart of Unix, it allocates time and memory, handles file operations and system calls.
The shell acts as an interface between the user and the kernel. The shell is what you are typing to at the Unix prompt after you log in.

- Basic Unix Knowledge
    - Login
    - Find out default shell
        - echo $SHELL
            - SHELL is a variable, to access it's value prepend the $ to get $SHELL, this works with any variable in your environment
            - To list environment variables use the env command
    - ssh
        - ssh ${USER}@powerpuff
        - Connect to powerpuff.cs.brynmawr.edu
        - exit
    - **man (RTFM)**
        - man <manentry> (women)
        - Displays the manual for a given command, whether a unix command or a C function.
        - Flags you should know
            - –k Search the man pages for the given word.
            - –a Displays all entries for the command instead of the first one (if there are more than one). This is important when shell commands and C functions have the same name
            - –s Display man page in the given section.
    - File manipulation
        - ls
            - Displays all the files in the directory.
            - Flags:
                - –l displays additional information for each file
                - –a displays all files, files that begin with "." are not displayed automatically
                - –F displays "/" after directories, and "*" after executables
                - –t sort by timestamp instead of alphabetically
        - cd
            - cd /rd/cs246s2016/$USER

        - Switches the current working directory. All file names that you enter are relative to the current working directory.
        - You can use this as your primary directory for storing your project repositories. lab0 is already here, but from now on we'll initialize repositories in our class user directory.
        - cd ..   goes up a directory
        - cd -   goes to the previous directory you were in
        - cd with no argument (cd ~)   takes you home
        - cd -
    - mkdir
        - mkdir lab01
        - this will be your lab01 directory (enter it now)
    - pwd
        - Displays the current working directory
        - It should be /rd/cs246s2016/$USER/lab01
  - Understand the Unix path
    - The Unix file system groups all files together in a hierarchical tree structure.
    - The top of the hierarchy is traditionally called the root.
    - Root is denoted by /
    - Any path name that starts with a / is a so-called absolute path name
    - Traverse the path (/rd/cs246s2016/$USER/lab01), e.g. whatever pwd returned:
        - cd /
        - ls
        - cd rd
        - ls
        - cd cs246s2016
        - ls
        - cd $USER
        - ls
        - cd lab01
        - ls
- Emacs
  - Buffer-based editor
  - Actually a fully functional LISP interpreter, but most people use it for editing purposes mostly. EXTREMELY POWERFUL; can do things like run a shell, do compilation internally, etc.
  - To start emacs: emacs filename (if filename does not exist, it will be created)
  - Emacs Commands
    - Syntax for commands
        - C-<letter> means "Control + <letter>"
        - M-<letter> means "Escape, followed by letter"
    - C-x C-s

        - Saves the current document
    - C-g
        - Cancel any mistaken command
    - C-_
        - Undo the last action.
    - C-k
        - Cuts the current line (starting from current cursor position).
    - C-spacebar
        - Sets the mark for the beginning of a text region
    - C-w
        - Cuts text from the last mark set to the current cursor. Note: in Unix, blocking text automatically copies it to the clipboard.
    - C-y
        - Paste. Can also be done with middle click.
    - C-a
        - Works just like the home key in Windows.
    - C-e
        - Works just like the end key in Windows.
    - M-x global-font-lock-mode
        - Colors your text for easy reading, similar to that in other editors.
    - M-x recover-file
        - If Emacs crashes, a "#" file will be created. Use this command to restore that file.
    - C-h t
        - Bring up the Emacs tutorial. Feel free to read this in addition to the Emacs guide sheet for other commands. Read until you can't stand it anymore. No one every finishes.
    - C-x C-c
        - Quits Emacs, will give opportunity to save.
        -

- Unix revisited
  - Some more commands now that we can make files
    - cp
        - cp test.txt and long.txt from /rd/cs246s2016/shared/lab01
        - cp test.txt test2.txt
        - Copies a file from one location to another.
    - mv
        - mv test.txt test3.txt
        - Moves a file from one location to another. Works across directories.
    - cat
        - cat test2.txt
        - cat test2.txt test3.txt

    - Displays multiple files in succession.
- more (less)
    - more long.txt
    - Displays a file in multiple pages determined by terminal size.
- clear
    - Clears all text on the screen except for a single shell prompt.
- hg
    - make sure you are in /rd/cs246s2016/$USER/lab01 using the pwd command, then
    - hg init
    - hg add
    - hg commit –m "initial revision"
- rm
    - rm test2.txt
    - **Unix does NOT have undelete!!! (But don't fear because we have version control)**
    - Flags you should know
        - –r Recursive Delete
        - –i Always Confirm
        - –f Force Delete
- nfs4_getfacl (get advanced file permisions: accesss control lists)
    - make sure you are in /rd/cs246s2016/$USER/
    - nfs4_getfacl lab01
    - you should see something like this
    - **A:fd:dgc@cs.brynmawr.edu:rwaDxtTnNcy**
    - **A::OWNER@:rwaDxtTnNcCoy**
    - **A:g:GROUP@:rxtncy**
    - **A::EVERYONE@:rxtncy**
    - You'll want to add your partner to the access control list for this lab. For this you'll use the next command, but first review your Unix Cheat sheet with your partner discussing file permissions (page 4).
- nfs4_setfacl (setting advanced file permissions over nfs4)
    - make sure you are in /rd/cs246s2016/$USER/
    - nfs4_setfacl -R -a A:fd:[partneruserid]@cs.brynmawr.edu:RWX lab01
    - nfs4_setfacl -a A:fd:[partneruserid]@cs.brynmawr.edu:X .

Part 2: (switch roles from driver to navigator and from navigator to driver)
Exercise:
Switch Drivers. Current Driver Logout, New Driver Login. In your home directory, go to your CS246 directory and clone lab01 from your partner's class home, with the following command

```
hg clone --pull /rd/cs246s2016/[partneruserid]/lab01 lab01
cd lab01
```

Note: you didn't have to do all of the umask commands when going in this direction.

1. Copy the file /rd/cs246s2016/shared/lab01/lab01.cpp.
2. Compile and run.
3. Use emacs to make the following changes to your copy of the file lab01.cpp:
   a. Delete the first two **`cout`** lines using C-k
   b. Now yank them back (C-y)
   c. Delete them again using set region then cut (C-spacebar, C-w)
   d. Now paste them back to the end of the program (C-y)
   e. Add your name and your partner's name to the header, and any comments you feel are appropriate.
   f. Save and quit
4. hg add
5. hg commit -m " added modified lab01.cpp"
6. Compile and run
7. hg push

- Unix FAQ
  - http://www.faqs.org/faqs/unix-faq/faq/part1/preamble.html
- History of Unix
  - http://www.bell-labs.com/history/unix/
- Emacs HOWTO:
  - http://www.tldp.org/HOWTO/Emacs-Beginner-HOWTO.html
- Emacs tutorial (yet another)
  - http://web.nwe.ufl.edu/writing/help/others/emacs/tutorial/index.html
- **Google and use key words unix tutorial, emacs tutorial**