Lecture 12 (2-9-26)

Exam 1 Review

- Part 1: **On paper** (multiple choice, fill in the blank)
  - **Closed** book, **closed** notes
  - Allowed **one cheat sheet** (front and back)
- Part 2: **Code submission** to assignments repository on Github
  - **Open** book, **open** notes, **open** internet
- may want to have scratch paper for one of the questions

WRITTEN PORTION

1 point : COMMANDS

- How should you prepare your git repository to submit work for **exam01**?
- Which of the following commands can you use to get the size of a file?
  - `stat`, `ls -l`, `du`?

1.5 points: SHELL

- How would you handle a permission denied error when running a scirpt?
- After installing a program called **doit** to `~/bin` what would you change so you can simply type **doit** to run the program?
  - modify PATH environmental variable

1.5 Points: Files and Processes

- How do you translate octal modes for a file?
- How would you interrupt, suspend, and kill a process?
  - Interrupt: `CONTROL-C`
  - Suspend: `CONTROL-Z`

- Kill : `kill -9 {PID}` or `pkill -9 {NAME}`
  - by default `kill` and `pkill` do sig term (so you need to specify `-9`)

---

1 Point: I/O redirection

- What is being redirected and where in the following pipeline:
  - `cowsay 2> /dev/null <message | wc -l`
  - `cowsay 2>&1 /dev/null | wc -l`
  - DRAW THE DIAGRAM FOR THIS ONE
- What is Unix Philosophy?

1.5 Points Networking

- What is the difference between public and private IP address?
  - 10 xxxx -> private
  - 192 168 -> private
  - remember local host
- What commands would use to:
  - Translate a domain name into an ip address?
  - Make a HTTP request
    - `curl` , `wget` , `ncp` (but tedious)
  - Measure latency
  - Scan ports on a remote machine
    - `nmap`
  - Securely copy files to/from a remote machine
    - `scp`

2.5 Shell Scripting

- Write a shell script that parses command line arguments:
  - Use a for loop
  - Use a while loop
  - will be select and scrammble
- Utilizes short circuit evaluation to replace conditionals

- ON HW2 QUIZ
- REVIEW FOR THIS:
- checking if file exists using short circuit evaluation:
    - ```
      [ -e sam ]
      echo $?

      ------

      [ -e sam ] && echo EXISTS
      #if sam exists, echo EXISTS

      [ ! -e sammy ] && echo MISSING
      #if sammy does not exist, echo MISSING

      ------

      [ -e sammy ] || echo Missing
      # This echos MISSING since sammy does not exist
      ```

CODING PORTION

- **FILTERS**
    - Looks like reading03
    - will have to download make file and some other file
    - will have to complete 4 pipelines
        - only have to do 2
- Most of them will need 5 or 6 commands
- also a regex expression
      ```