Lecture 3 (1 -15-26)

Review from last lecture

- What is git?
  - Basically a checkbook where you can keep track of a set of files
    - make sure you submit to appropriate branch
    - only make pull request once
      - if you need to update, just add commit and push again
      - you **should not** merge your pull request
- What is a shell?
  - an application
- What is Unix?
  - Family of operating systems

New Stuff

- **Computing Stack** :
  - **Bottom**
    - Hardware (logic design, comp arch)
    - Operating Systems
    - Libraries
    - Application such as shell (fund comp, data structs, programming paradigms)
  - **Top**
  - (Systems programming and operating systems) is in the middle of operating system and libraries
- **Where is `ls` located?**
  - `ls` is located under **user bin**
    - `echo $PATH` -> returns paths to potential commands
  - The `PATH` environmental variable is used to find paths to commands
  - Remember environmental variables do not persist from session to session
- `man hier` command gives you the description of the filesystem hierarchy
- **Files: Hierarchy**
  - **Root**
    - `bin` (system applications / programs)

        - `dev` (hardware / devices)
        - `etc` (configuration)
        - `tmp` (scratch space)
            - everyone can read and write to it
                - files here go away after reboot
        - `var` (application data)
        - `usr` (user applications)
- **How do I switch users?**
    - If I go into someone elses home directory, I can't see anything. WHY?
        - Do `id -a` to see your info
        - you can do `su apatel8`
            - this will switch your identity to `apatel8`
- **How do I see the size of bashrc?**
    - `bashrc` is a **hidden file**
        - so you neen `ls -a` to see it
        - you can do `ls -l bashrc` to see the size
    - A filesytem stores metadata in an **inode**
        - a filesystem is just a **tree**
        - The metadata is:
            - Mode
            - Owner
            - Size
            - TimeStamp
        - You can use `stat .bashrc` to se the inode data
- **How do I create a shortcut to a file/directory?**
    - a **hardlink** is extremely efficient
    - **hardlink points to the data itself**
    - **but you cannot use hardlink across partitions**
        - When you create one the inodes have the same exact item number
    - To create a **softlink or simlink:**
        - A softlink creates a new file
            - The inodes have different item numbers
        - A **softlink** points to the path to the data
        - **You can use softlink across partiions**