Unix/Linux

- powerful class of operating systems
- multiuser, unified file system
- command-line interface
  - text-based commands, no graphical user interface (GUI)
  - “long-term lazy”: slower to learn, faster to use
  - allow for easy automation of series of commands

Linux vs. Windows vs. Mac

Unix/Linux

- the term *directory* means the same thing as folder
- Unix/Linux commands are case-sensitive
- ~ denotes the home directory of your account
- . denotes the current directory
- .. denotes the parent directory (one level up from current)
- when you log into ugrad, you'll be in your own home directory

Connecting to ugrad demo

- If you have a Windows machine
  - Download and install PUTTY for free (https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html)
  - Open PuTTy and connect to ugrad:
    - Open connection with hostname ugradx.cs.jhu.edu
    - Username and password are as supplied on account sheet (e.g. ips120xx or CS/CE major/minor account)
  - Once logged in, change password:,
    - Type passwd, then re-enter old password
    - Choose 1, enter old password again, then new password twice
    - Remember new password you choose - we can't reset it for you!
  - Log out of ugrad by typing exit
  - Log back into ugrad; ensure new password works; log back out
- Alternatives to ugradx are ugrad1, ugrad2, . . . ugrad24

Connecting to ugrad from your own computer

- Windows: download PuTTy and repeat what we did today (www.chiark.greenend.org.uk/~sgtatham/putty/download.html)

- Mac: open Terminal application, then type the command ssh <your-username>@ugradx.cs.jhu.edu

Unix/Linux: basic commands

- pwd - print working directory
  - `pwd`
- ls - list directory contents
  - `ls`
  - `ls -l`
  - `ls -la`
- cd - change directory [specify directory name]
  - `cd <folder_name>`
  - `cd ..`
  - `cd ~`
- mkdir - make new directory within current directory [specify name]
  - `mkdir <folder_to_create>`
- less - view text file screenful at a time [specify file name]
  - `less <file_to_view>`

Unix/Linux: basic commands, continued

- mv - change location of file or folder [specify source, then destination]
  - `mv <source> <destination>`
  - `mv hellooo.c hello.c`
- cp - make a copy of file or folder [specify source, then destination]
  - `cp <source> <destination>`
- rm - remove a file [specify file to remove]
  - `rm <file_to_remove>`
- locations can be *relative* to current directory
  - `cp hello.c folderForToday/hello.c`
- locations can be *absolute*, based on full path in file system, so you don't have to change to a directory to move or copy files in from/to that location
  - `cp ~/oldFolder/hello.c ~/folderForToday/hello.c`

Items to do!

- Make sure you have a cs account (either your permanent one or a temporary one you received from the instructor)
- Finish a Unix/Linux command line boot camp (tutorial)
  - http://korflab.ucdavis.edu/bootcamp.html