

UCLA CS35L:
Software Construction
Laboratory

Section 1

BH3760, MW 10:00~11:50

Instructor info

- Instructor in charge
 - Prof. Paul Eggert
 - BH4532J
 - Office hour: Mondays 10:00–11:00 and Thursdays 09:30–10:30
- Section 1 instructor
 - Hongyi Wang
 - hywang@cs.ucla.edu
 - Please include “CS35L” in subject
 - Office hour: BH2432, Thursday 11:30~13:30

Who is this course for?

- Prerequisite: CS31
- Interested in
 - Linux
 - Commonly-used open source software tools
 - “Software construction”
 - Preparation for upper-division CS courses (CS111, etc.)

Course workload

- Laboratory, 4 hours; outside study, 2 hours
- 10 Assignments
 - Laboratory exercise
 - Homework
 - Presentation and report

Assignments

- Laboratory exercise
 - Expected to be done in the lab session
 - Answers/outputs for the given questions
 - Lab notes required
 - A log of your actions for others to reproduce your work
- Homework
 - Experiments, programming works, etc.
- DO NOT share your code with others

Assignments

- Presentation & report
 - Select a topic from the given sources
 - Individual work
 - Presentation: 10~15min
 - Report: 300~1200 words
 - Choose your topic and presentation schedule from 2nd week
 - One slot for each topic
 - First come, first served
 - Tentative presentation schedule: start from 3rd week, 4 talks per week

Grading

- 50% homework
 - Equally weighted
 - Delay penalty: $2^N\%$ of the assignment's value if submitted between N and $N+1$ full days late
 - Exception: not accepted after than the last day of instruction
- 50% final exam
 - Open book & open notes
 - No written makeup exams will be given

Syllabus

- 1. Introduction, files and editing
- 2. Commands and basic scripting
- 3. More scripting, VMs, and construction tools
- 4. Change management
- 5. Low-level construction and debugging
- 6. Systems programming
- 7. Faults, failures, errors, and holes
- 8. Security basics
- 9. Parallelism
- 10. The crystal ball

Useful links

- All the details regarding the course can be found on the course page:

<http://cs.ucla.edu/classes/winter14/cs35L/>

Introduction to Linux 1

Week 1 - Monday

A Brief History of Operating Systems

- The Dark Ages
 - No OS until 1960s
 - Manually loaded programs
 - Reboot after each program
- Batch OS
 - Unified application development across systems
 - Output via printer, later via monitor
 - I/O via magnetic tape or disk
 - Written in assembler (e.g., OS/360)
 - Multiprocess

A Brief History of Operating Systems

- Timesharing OS
 - Multiuser
 - Multics (1964)
 - Segmented memory
 - Paged virtual memory
 - Applications written in many languages
 - Shared multiprocess memory
- Personal Computer
 - Single machine for single user
 - OS must manage screen and input devices
 - Window, Icon, Menu, Pointing Device (WIMP, e.g., MacOS, 1984)
- Cutting-Edge OS
 - High performance computer (HPC) clusters (e.g., BlueGene/L at LLNL rated at 280.6 teraFLOPS)
 - Cell phones, video
 - Video games
 - Browsers

CLI vs. GUI

CLI

- Steep learning curve
- Pure control (e.g., scripting)
- Cumbersome multitasking
- Speed: Hack away at keys
- Convenient remote access

GUI

- Intuitive
- Limited Control
- Easy multitasking
- Limited by pointing
- Bulky remote access

File System

- Everything is a file
 - Regular file/directory
 - Device is also a file!
- Tree structured hierarchy
 - “/u/cs/grad/hongyi/Documents”
 - Compared with Windows, “C:\windows\”

Permission

- Every file has its own permission control
 - By user group
 - The owner of the file
 - Users belonging to the same group of the owner
 - Other users
 - By function
 - Read
 - Write
 - Execute
 - Why my script cannot be executed?

Commands: The basics

- <up arrow>: previous command
- <tab>: auto-complete
- Get lost? Need help for a command?
 - man: display the manual page of a command/function/file
 - “man ls”
 - “man man”
 - A brief help is usually available via “--help”
 - But not always available
 - Google

Commands: Moving Around

- `pwd`: print working directory
- `cd`: change working directory

- `/`: root directory, or directory separator
- `~`: home directory
- `.`: current directory
- `..`: parent directory

Commands: Dealing with Files

- mv: move a file (no undos!)
- cp: copy a file
- rm: remove a file
- mkdir: make a directory
- rmdir: remove a directory
- ls: list contents of a directory
 - -d: list only directories
 - -a: list all files including hidden ones
 - -l: show long listing including permission info
 - -s: show size of each file, in blocks

Commands: File Name Matching

- `?`: matches any single character in a filename
- `*`: matches one or more characters in a filename
- `[]`: matches any one of the characters between the brackets. Use `'-'` to separate a range of consecutive characters.

Commands: Changing File Attributes

- touch
 - update access & modification time to current time
- chmod
 - Change the permission settings of a file

Commands: Search a file

- `find`
 - `-type`: type of a file (e.g., directory, symbolic link)
 - `-perm`: permission of a file
 - `-name`: name of a file
 - `-prune`: don't descend into a directory
 - `-o`: or
 - `-ls`: list current file

Commands: Look These Up

- echo
- cat
- head
- tail
- du
- wc
- sort
- diff/patch
- which
- cron
- ln

Preparing your Linux

- Recommended distribution: Ubuntu
 - Recommended version: 13.10
 - Downloading & installing guides:
<http://www.ubuntu.com/download/desktop>
- If you are an expert, feel free to use any distribution