

CSE 333

Section 9

Future Classes, Wrap-up, and Final Review

Logistics

- **HW4**

- Was due **Yesterday @ 11:00 pm**
 - Up to 2 late days allowed if you still have them – see “late days remaining” on canvas to check if you’re not sure

- **Final Exam**

- **Exam Tomorrow (8/22) @ 1:10 pm**
 - HRC 155 (same time and room as lecture)

- **Office hours**

- You’re getting extra credit for showing up!
- Audrey’s are today at 3pm, 4th floor landing in Allen

Homework Summary

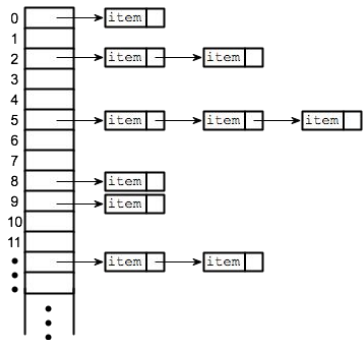
Homework Summary

Homework 1: Implemented a Doubly-Linked List and a Chained Hash Table

Homework 2: Created a local search engine (like grep)

Homework 3: Modified HW2 to save and read index files from disk

Homework 4: Creating an internet-accessible version of your search engine!



```
[attul hw2]$ ./searchshell ./test_tree
Indexing './test_tree'
enter query:
ulysses
./test_tree/books/ulysses.txt (10)
./test_tree/books/countofmontecristo.txt (2)
./test_tree/books/leavesofgrass.txt (1)
./test_tree/books/lesmiserables.txt (1)
./test_tree/books/paradiselost.txt (1)
enter query:
```



Outcomes of Doing the Homework

- Building a sample system based on the design decisions the staff has made for the homeworks
 - You can consider why these decisions were made
- Getting accustomed to larger-scale programming projects
 - Learning about the code surrounding what you are to implement
- Practicing programming idioms – error checking, style guides, etc.
- Reading Documentation!

Adding it to your Resume

- Why consider adding it in?
 - You had to work with this code base for an entire quarter
 - It demonstrates your ability to work with a large code base and system
 - It's pretty cool to add “mini-Google” in
- What might be on there?
 - **System Design** – Creating data structures to store information. Saving it into a file. Following a protocol to create a web application
 - Programming in C and C++ (handling C/C++ idioms)

What's Next?

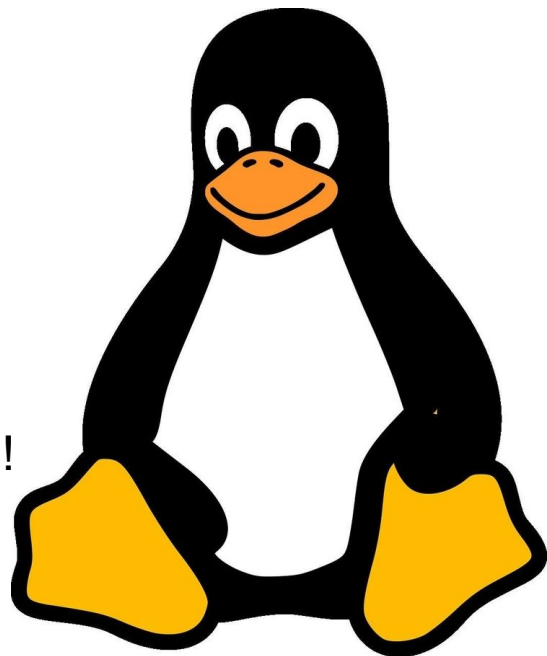
Main Course Themes



- Debugging: gdb, valgrind, and... rubber duckies
 - Solving Seg. Faults, Memory Leaks, incorrect output
 - C/C++ programming hopefully made you more cognizant of where bugs may occur at a low level
- Program Layout: Modularity, Helper functions, Inheritance
 - Improve readability and ease of maintenance
- Documentation: Reading API's and incorporating into programs
 - C++ Reference Guide, man pages
- Google Style Guide: linter, Lots of reading...
 - cpplint.py is an automated tool to help with catching style issues
 - A lot of it was learning to adapt to a system for writing and reading code in the same format

CSE 451 – Operating Systems

- CSE333 is called “Systems Programming”
 - We hope you learned to interact with the OS
 - “Magic OS stuff” kept mostly hidden in CSE 333
- If you want to understand the “magic OS” stuff, take this!
- Usually in C ☺
- Partner class!
- Lots of work, but very rewarding!



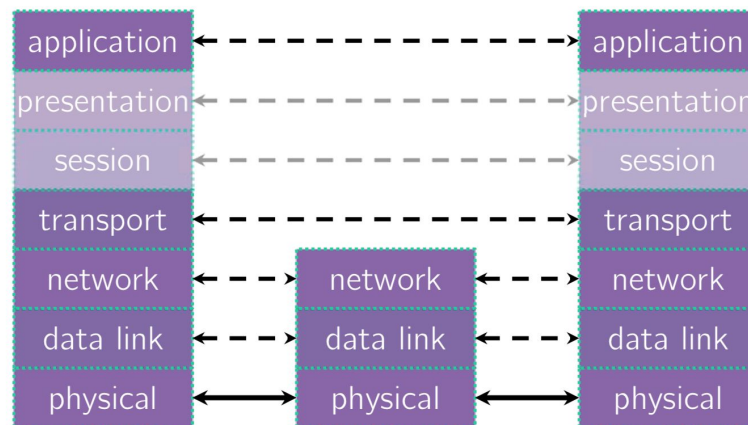
CSE 452 – Distributed Systems

- Understanding how to design massive, fault tolerant systems over an unreliable network.
 - Think Facebook, AWS, Google, etc.
- Doesn't require OS, topics rarely if at all come up
- More theory to it than you'd think
- Lots of work (project is big and confusing)
- Most thought per line of code of any class in the Allen School, also many lines of code
- Hard, but very very cool. Would strongly recommend. Huge resume booster.

CSE 461 - Networks

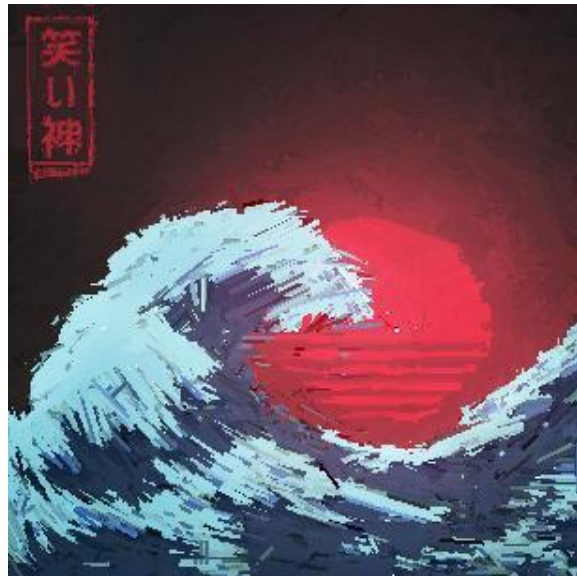
- We did a “basic” understanding of how networks work
- We mainly just understood how it works in C and with POSIX
- Mostly Python
 - First lab is “language of choice”

- Teams/Group Work



CSE 457 - Graphics

- Cool, but hard
- Lots of C/C++ in many versions, some linear algebra & physics
- If you want to do graphics, you usually have to do it in C++
- You will make cool artifacts in this course!
- Skeleton code provided is not documented well and a little bit “spaghetti” 😞



CSE 455 – Computer Vision

- Deals with bytes in an image
 - This part is in C, but maybe other languages 😊
- Deals with ML
 - “New ways” to do computer vision via ML
- Sometimes, these are the people making atttu slow 😞



Shoutout: Other Classes

- How about that “mysterious” compiler?
- Liked HTML (for some reason)?
- Want to do C on limited systems?
- Learn about more low-level stuff?
- Want to learn about CPU magic?
- Liked patching up your 333gle? ssh-keys?

401 Compilers

154 Web Dev

474 Embedded Systems

369 & 371 Digital Design

469 & 470 Comp Arch

484 Security

TA-ing

- You have the background to TA CSE333, CSE351, CSE374 and others.
- You do NOT have to get 4.0 in a class to TA it
- You do NOT have to be a super social person to TA
(Some of us are very introverted)
- TAing will reinforce your understanding of any material
- **TAs are human too. It is okay to start off imperfect and make mistakes**
- **If you think you would be interested, we would highly recommend reaching out and giving it a try. Please feel free to talk to us if you are interested!**

Final Review

We give the people what they want ;)

- Which topics or problems do you want to go over?
- Old finals can be found here:
 - <https://courses.cs.washington.edu/courses/cse333/25su/exams/index.html>
- Summer 2025 final exam topics:
 - <https://courses.cs.washington.edu/courses/cse333/25su/exams/finaltopics.html>

Topics

- C++
 - Casts
 - Templates
 - STL, containers, iterators
 - Smart pointers
 - Inheritance
 - Dynamic vs. Static Dispatch
- Concurrency
 - Efficiency
 - Processes and forks
 - Multithreading
- Networks
 - Layers
 - Packets
 - IP Addresses
 - TCP stream sockets
 - Client-side steps
 - Server-side steps
 - HTTP/HTML basics

Wrap-up Wrap-up

Before We Go...

This quarter has been tough!

This section wouldn't have been the same without all of you!

We were glad to have you! You earned your place here, you belong here (on this campus, in this program, in this course)

You know the content, take it slow, you can do it!

You've Learned A Lot! Good Luck!



Thank You for a Great Quarter!