CS 202 (-002) Operating Systems

(Sorry, No Movies In This Class)

Instructor: Aurojit Panda

TAs:
- Zhanghan Wang
- Sam Frank
- Mengxi Liu
- Leanne Lu
- Brayton Lordianto
- Jeff Ma

- Course Website

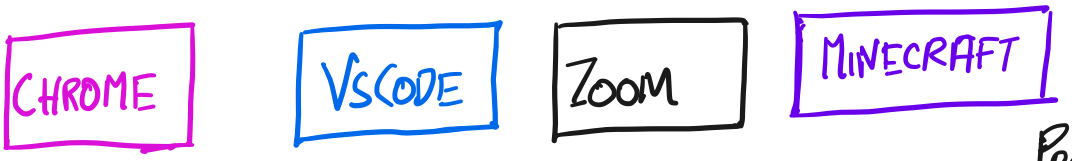
  https://cs.nyu.edu/~apanda/classes/23fa

  Should already have recieved Campuswire invite

Today

- [ ] Goals
- [ ] What is an Operating System?
- [ ] Why Study It?
- [ ] How Will We Study It?
- [ ] Mechanics, Admin, Policies
- [ ] History
- [ ] Processes

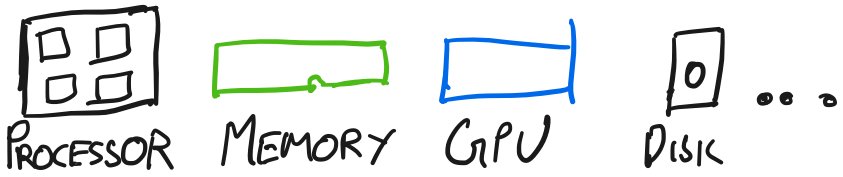
Operating Systems



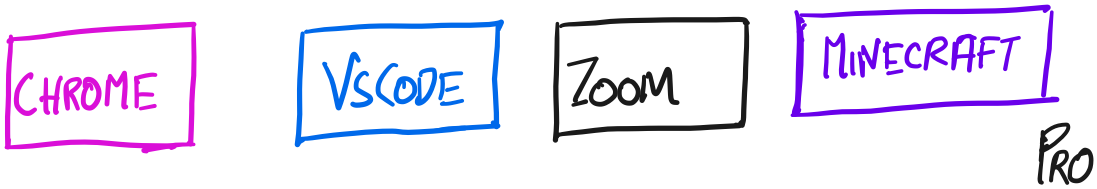
PROGRAMS

Q1. How To Share Hardware Resources B/w Processes?

Q2. How Do Programs Use Resources Assigned To Them?



HARDWARE RESOURCES



PROGRAMS

OPERATING SYSTEM

HARDWARE RESOURCES

SHARING

↳ ① Virtualization Mechanism: Allow Many Programs Access

② Isolation: Bad Program Cannot Affect Another

How Accessed

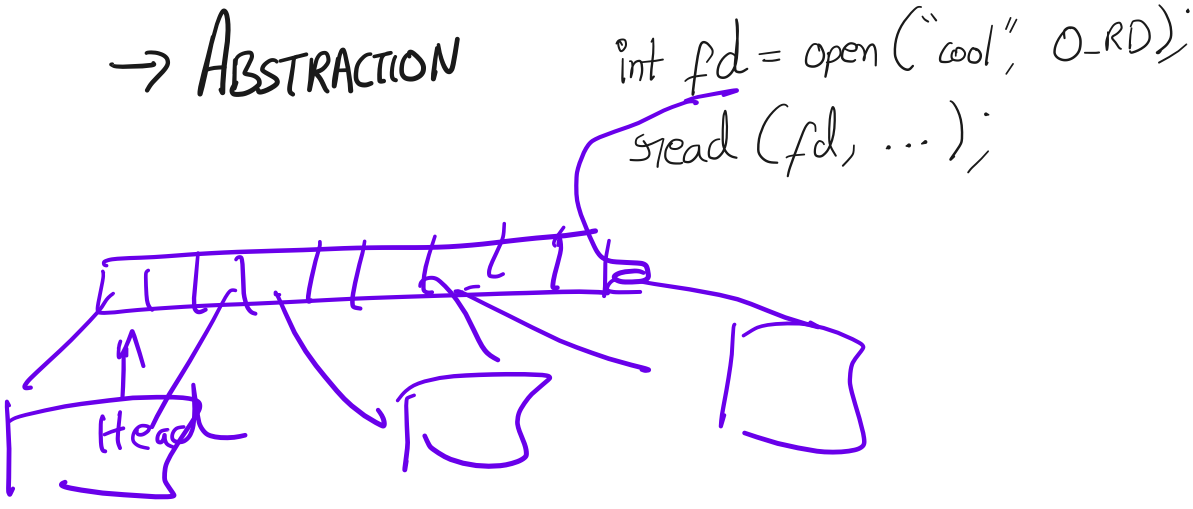
↳ Abstraction: APIs, Data Types

① File System

↳ Virtualization: —

→ Isolation: —

→ Abstraction



② Memory

Covered later in class

- Virtualization:
- Isolation:

Abstraction:

③ SCHEDULING

ABSTRACTION:

VIRTUALIZATION:

ISOLATION:

WHY STUDY (OPERATING) SYSTEMS

- Bridge between hardware and programs

→ Determines how your program looks

→ What performance it gets

→ Is it correct?

→ Is it practical?

Thinking about say LLMs

Application evolution & h/w adoption often driven by systems changes

- Ton of emerging problems that need to be solved

- Provides a deeper understanding of anything running on a computer

-How study?

- Course mechanics

Components

- Class Tu–Th.
  - ↳ Recorded & on Zoom (see Brightspace)
  - → Strongly encourage attending
  - → PLEASE ask & answer questions
  - → Provide feedback
    - ↳ Goal is to make it useful for you.
- Reading
- HW
- LABS
- Recitation / Review
  - ↳ First one 09/11 MONDAY 6–7PM
- EXAMS
  - ↳ MIDTERM 10/19 DURING CLASS
  - → FINAL

Getting Information & Help

- Check Website (daily)
- Questions About Labs & Homework
  - ↳ Post On Campuswire
  - → Please Help Answer When Possible
- Sensetive/Admin Questions

  cs202-fa23-staff@nyu.edu
- Office Hours
  - ↳ Schedule on webpage

Grading

HW	5%
Labs	25%
Midterm	25%
Final	45%

- Policies
  - ↳ All work that you hand in must be yours.

- Code

- What is allowed

→ Copilot & its ilk

→ Bottomline

History

- Batch Processing (starting 50s/60s)

- Time Sharing (coined 1950s, Built 58-...)
  - ↳ Compatible Time Sharing
  - → Joss
  - → Multics
- Unix (1969 AT&T Bell Labs)

UNIX & Plan 9

↳ Multi-user, Multi-process

→ PORTABLE: Mostly written in C starting version 4 (Nov 1973)

(First version with most things associated with UNIX / "The Unix Philosophy")

Targets minicomputers

- Parallel: Development of Personal Computers
  - → Hobbyist → Apple
    - → C64
    - → ...
  - → For "professionals" — Xerox Parc
    - — ...

- Merge late 1980s
  - PCs became more powerful, cheaper

↳ Workstations replace minicomputers

- Berkeley 1980 – 199?
  - Tools for UNIX VESION 5 ON
    - VI, csh, ...
  - Improved kernel
  - Ported to PC
  - Eventually ends up in OSX
- Linux (early 1990s)
  - Written from scratch
  - ...

- Lots of undiscussed paths
  - Lisp Machines
  - Plan 9
  - NT
  - Dartmouth
  - ...

PROCESSES

```
int r = fork();   ← Here lies magic
if (r==0) {
  ...
} else {
```