Introduction

Operating Systems Practicum
CS 4411
Cornell University
Lectures

• Friday 2:30-3:20pm
• Hollister Hall B14

• Either Lecture or Help Session
• Today (8/26): Lecture
• Next week (9/2): Lecture, Introducing 4411-P1
• Third week (9/9): Help Session

• Online schedule coming soon!
Enrollment

- No auditing
- Anyone already enrolled in 4410 should be able to enroll in 4411. If this is not the case, let us know.

- For those of you who have a conflict with the databases class, we have decided to let you enroll in 4411 AT YOUR OWN RISK.
  - Trying to arrange (but not guaranteeing) a duplicate session
  - ultimately you are responsible for being up to speed
Office Hours

• There is only one Google Calendar for 4410/4411 Office Hours

• We will mark some Office Hours as exclusively 4411 office hours.

• TAs that list their hours as “4410 OH” have not taken 4411. Do not ask them about the projects.

• Staff Page will also differentiate between TAs.
Piazza

- We have our own Piazza
  - Please don’t post 4411 questions to 4410
  - Please don’t post 4410 questions to 4411
  - Don’t post code!
Projects

- Enrolled in 4410 & 4411: only required to do 4410’s Project 1 & 2. Both individual assignments.
- 4411: something due approximately every 2 weeks
- (If you are *only* enrolled in 4411, you need to do *only* the 4411 projects)
- You have 6 Projects in 4411

- Project are done in pairs. No partner?
  - Find one here
  - search on piazza
  - fill out the google form to be automatically assigned a partner (linked on piazza)
PortOS

- Most projects in PortOS
- Runs on Linux
- There will be a course VM
  - Work on your own machine at your own risk and definitely test on the VM before submitting

- Begins with some code base (not a blank file)
- Projects Build
  - Project n+1 will include your Project n code
- Tools like gdb, valgrind, purify are not going to play nicely with your assignments...
A word about C

• All 6 projects are in C
• Project #1 will be released next Thursday
  ▪ If your C is rusty → brush up NOW
  ▪ You should know:
    • Pointer arithmetic
    • Malloc, realloc, free
    • Typedefs, Type casting
    • void *, void **
    • Function pointers
    • Basic data structures (linked lists, etc.)
Partner Search

• Want to find a partner in person?
• Come to the front of class now.