CS 4410 / CS4411 Operating Systems

Prof. Sirer & Prof. George

Fall 2014

Cornell University

What’s This Course About?

- Ostensibly, it’s about operating systems
  - architecting complex software
  - identifying needs and priorities
  - separating concerns
  - implementing artifacts with desired properties
- It’s really about software design principles
  - It just so happens that OSes illustrate organizational principles and design patterns

What Kind of a Course Is This?

- Constructive, top-down
  - We start from first principles and re-derive the design of every component of a complex system
- Deconstructive, bottom-up
  - We dissect existing systems, learn what tradeoffs they make, what patterns they use

Course Objective

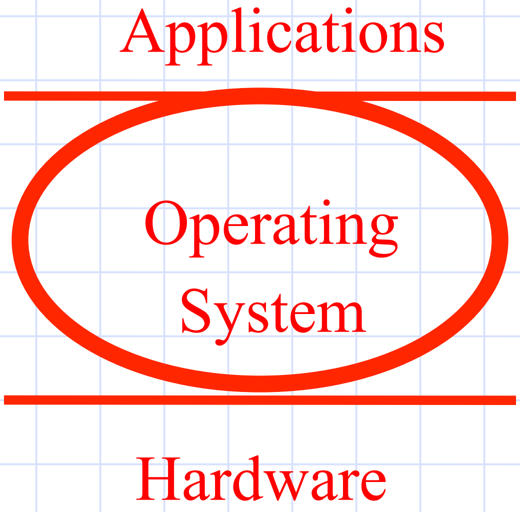
Establish a foundation for building complex programs

- Architect systems
- Identify desirable properties
- Build large systems
- Debug complex systems

Understand all layers of the software stack between hardware and applications

What is an Operating System?

- An operating system (OS) provides a virtual execution environment on top of hardware that is more convenient than the raw hardware interface
  - "All of the code you did not write"
  - Simpler
  - More reliable
  - More secure
  - More portable
  - More efficient



What do OSes do?

- Manage physical resources
- Provide virtual resources
- Implement mechanisms and enforce policies to arbitrate access to resources
- Mediate the interaction of mutually distrusting applications
- Provide an extensible, general-purpose platform for a variety of applications

What Resources Do OSes Control?

- Physical Resources
  - CPU, memory, disks, networks, I/O devices, …
- Virtual Resources
  - Files, directories, sockets, names, …

Issues in OS Design

- Structure: how is an OS organized?
- Concurrency: how are parallel activities created and controlled?
- Sharing: how are resources shared?
- Naming: how are resources named by users?
- Protection: how are distrusting parties protected from each other?
- Security: how to authenticate, authorize and ensure privacy?
- Performance: why is it so slow?

More Issues

- Reliability: how do we deal with failures?
- Extensibility: how do we add new features?
- Communication: how do we exchange information?
- Scale: what happens as demands increase?
- Persistence: how do we make information outlast the processes that created it?
- Accounting: who pays the bills and how do we control resource usage?

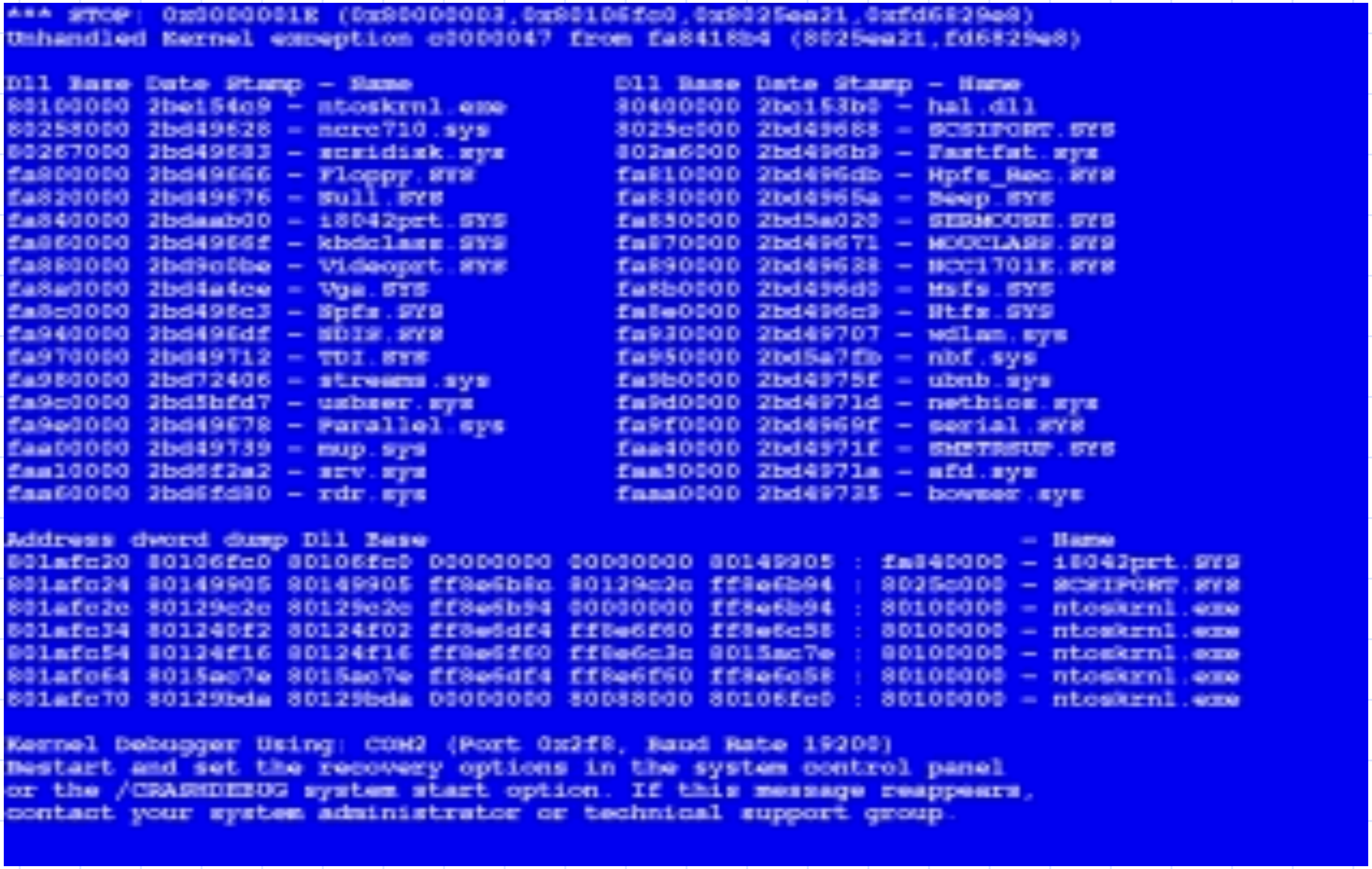
Why Learn Operating Systems?

- At most 1% of software developers will work on OS code
- The material taught in this course is critical not just for building Oses, but for building correct, high-performance applications (clients of Oses), as well as building reusable platforms for others
- This course will go far beyond OS design to cover all aspects of computer organization, including concurrency, synchronization, input/output, filesystems, networking, routing, distributed systems and so forth
- Engineering pride alone requires full understanding

Fact

- There has never been as exciting a time to work on systems hardware and software as now!!!
- The world is increasingly dependent on computer systems
  - Connected, networked, interlinked
- People just do not know how to build robust systems

BSOD



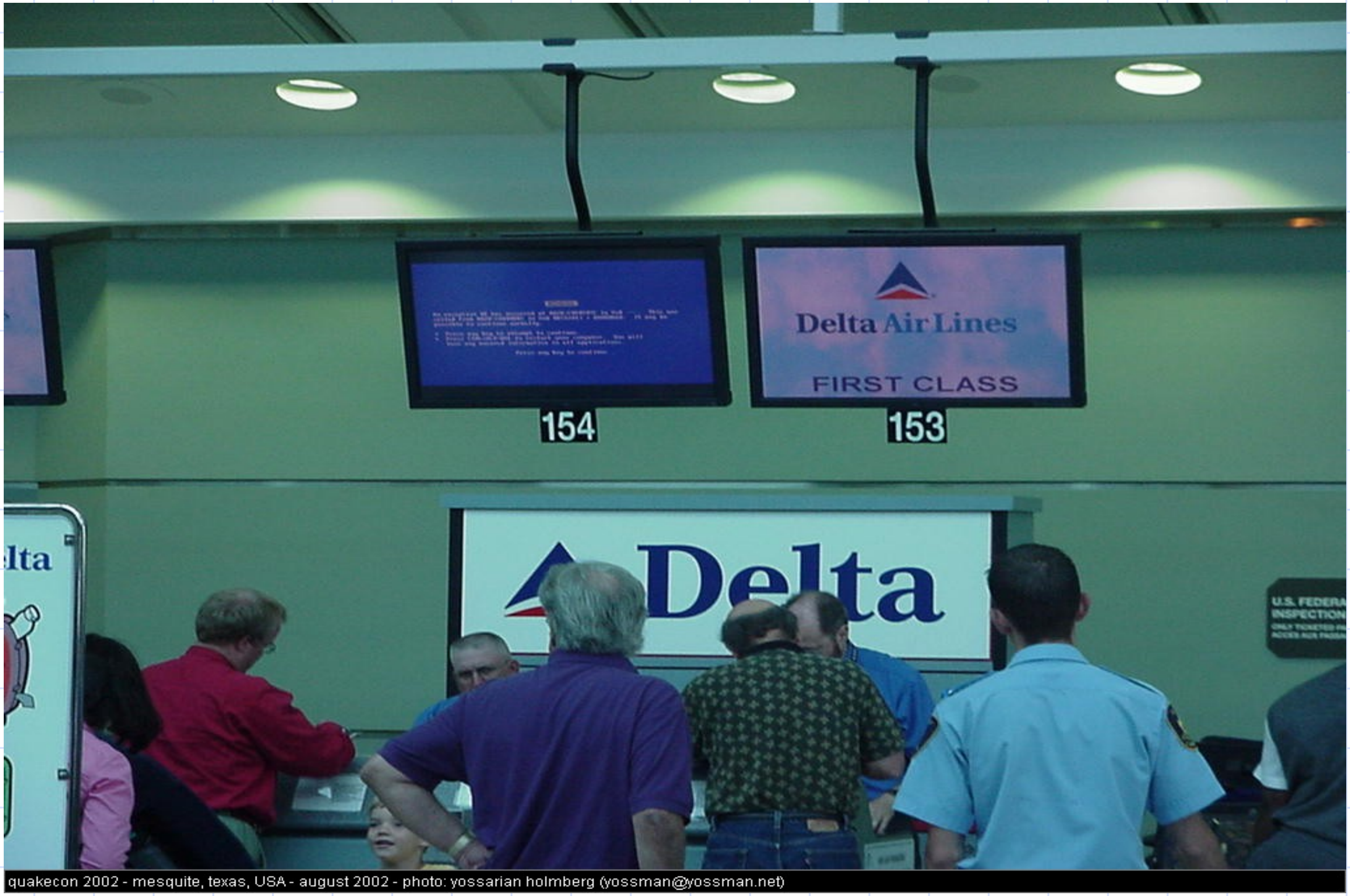
```
*** STOP: 0x0000001E (0x80000003,0x80106fc0,0x8025ea21,0xfd6829e8)
Unhandled Kernel exception c0000047 from fa8418b4 (8025ea21,fd6829e8)

Dll Base Date Stamp - Name             Dll Base Date Stamp - Name
80100000 2be154c9 - ntoskrnl.exe       80400000 2bc153b0 - hal.dll
80258000 2bd49628 - ncrc710.sys        8025c000 2bd49688 - SCSIPORT.SYS
80267000 2bd49683 - scsidisk.sys       802a6000 2bd496b9 - Fastfat.sys
fa800000 2bd49666 - Floppy.SYS         fa810000 2bd496db - Hpfs_Rec.SYS
fa820000 2bd49676 - Null.SYS           fa830000 2bd4965a - Beep.SYS
fa840000 2bdaab00 - i8042prt.SYS       fa850000 2bd5a020 - SERMOUSE.SYS
fa860000 2bd4966f - kbdclass.SYS       fa870000 2bd49671 - MOUCLASS.SYS
fa880000 2bd9c0be - Videoprt.SYS       fa890000 2bd49628 - NCC1701E.SYS
fa8a0000 2bd4a4ce - Vga.SYS            fa8b0000 2bd496d0 - Msfs.SYS
fa8c0000 2bd496c3 - Npfs.SYS           fa8e0000 2bd496c9 - Ntfs.SYS
fa940000 2bd496df - NDIS.SYS           fa930000 2bd49707 - wdlan.sys
fa970000 2bd49712 - TDI.SYS            fa950000 2bd5a7fb - nbf.sys
fa980000 2bd72406 - streams.sys        fa9b0000 2bd4975f - ubnb.sys
fa9c0000 2bd5bfd7 - usbser.sys         fa9d0000 2bd4971d - netbios.sys
fa9e0000 2bd49678 - Parallel.sys       fa9f0000 2bd4969f - serial.SYS
faa00000 2bd49739 - mup.sys            faa40000 2bd4971f - SMBTRSUP.SYS
faa10000 2bd6f2a2 - srv.sys            faa50000 2bd4971a - afd.sys
faa60000 2bd6fd80 - rdr.sys            faaa0000 2bd49735 - bowser.sys

Address dword dump Dll Base                                      - Name
801afc20 80106fc0 80106fc0 00000000 00000000 80149905 : fa840000 - i8042prt.SYS
801afc24 80149905 80149905 ff8e6b8c 80129c2c ff8e6b94 : 8025c000 - SCSIPORT.SYS
801afc2c 80129c2c 80129c2c ff8e6b94 00000000 ff8e6b94 : 80100000 - ntoskrnl.exe
801afc34 801240f2 80124f02 ff8e6df4 ff8e6f60 ff8e6c58 : 80100000 - ntoskrnl.exe
801afc54 80124f16 80124f16 ff8e6f60 ff8e6c3c 8015ac7e : 80100000 - ntoskrnl.exe
801afc64 8015ac7e 8015ac7e ff8e6df4 ff8e6f60 ff8e6c58 : 80100000 - ntoskrnl.exe
801afc70 80129bda 80129bda 00000000 80088000 80106fc0 : 80100000 - ntoskrnl.exe

Kernel Debugger Using: COM2 (Port 0x2f8, Baud Rate 19200)
Restart and set the recovery options in the system control panel
or the /CRASHDEBUG system start option. If this message reappears,
contact your system administrator or technical support group.
```

BSOD

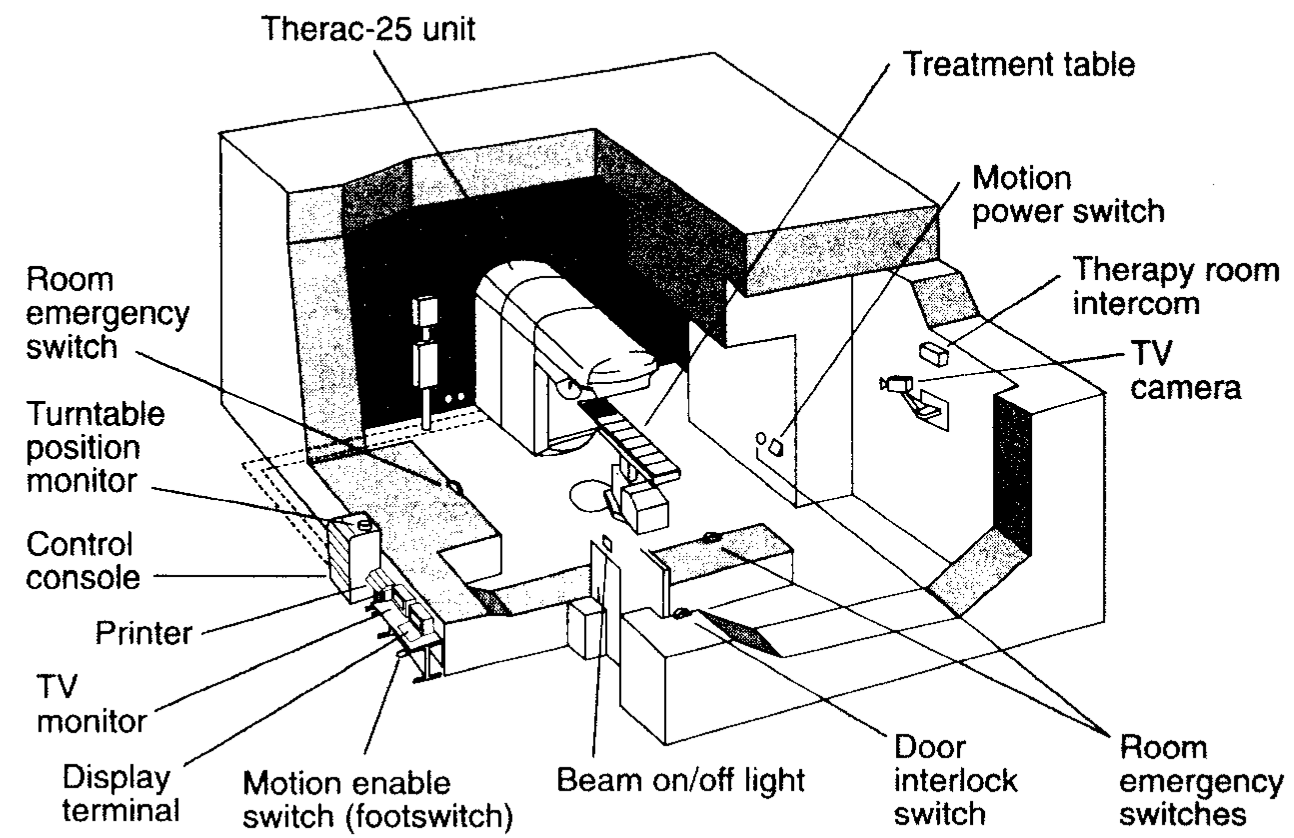


quakecon 2002 - mesquite, texas, USA - august 2002 - photo: yossarian holmberg (yossman@yossman.net)

BSOD

Photo courtesy of zem.

Therac-25



- A safety-critical system with software interlocks
- Beam controlled entirely through a custom OS

Therac-25

- Old system used a hardware interlock
    - A lever that could either be in the “zap” or “x-ray” position
- New system was computer controlled
- A synchronization failure was triggered when competent nurses used the back arrow to change the data on the screen “too quickly”

Therac-25 Outcome

- Beam killed one person directly, burned others, and may have given inadequate treatment to cancer patients
- Problem was very difficult to diagnose; initial fix involved removal of the back arrow key from the keyboard
- People died because a programmer could not write correct code for a concurrent system

Denver airport

- The most modern, most expensive airport in recent history
  - Cost overrun in excess of $2B

- Highly automated luggage handling system was supposed to deliver your luggage to you at arrival

- Lack of persistence caused luggage carts to “forget” their contents, sprinkling the luggage on the runway

USS Yorktown

- Modern guided missile cruiser off of Norfolk, VA in 1998
- Crew member entered 0 for a data value
- Cascade of failures led to a failure of the propulsion system
- Yorktown was dead in the water for a couple of hours

Bitcoin Banks

- There is a new crop of systems for holding data known as NoSQL databases

- We cautioned about the errors people were committing with NoSQL systems

- The simplest attack against banks, of simultaneous withdrawals, actually works! Attackers stole millions of dollars from Bitcoin banks
  - One bank lost a million and folded
  - Another lost $500K, took losses

Other Systems

- FAA air traffic control system
- IRS data management system
- IBM “Microkernel”
- Pentagon data security
- Many others, too numerous to list

Summary

- We do not have the necessary technologies and know-how to build robust computer systems
- There is huge demand for people who deeply understand and can build robust systems

Philosophy

- Not your grandfathe’s OS course
- We believe that the following topics are critical for every software engineer
  - Building concurrent systems
  - Architecting networked components
  - Building transactional systems
- Not widely shared, the course is unique

Logistics

- Lectures
  - Tuesday, Thursday 2:55-4:10pm, Ives 305
- Instructor
  - Prof. Sirer, egs at systems.cs.cornell.edu
  - Prof. George, mdgeorge at cs.cornell.edu
  - Office Hours
    - Prof. Sirer: Tuesday 4:10-5:10pm
    - Prof. George: Thursday 4:10-5:10pm
- TAs

Communication

- Course Web Page
  - Office hours, assignments, lectures, and other supplemental materials will be on the web site
- Email
  - cs4410staff at systems.cs.cornell.edu
  - The email alias goes to us and the TAs, not to the whole class

Administrative

- Course has three components
  - Lectures and Readings
  - Exams
  - Projects and MiniProjects
- Textbook
  - Anderson and Dahlin
  - XXX
- You are expected to keep up with all three

Grading

- CS4410 Breakdown
  - ~40% MiniProjects
  - ~12% Prelim 1
  - ~18% Prelim 2
  - ~25% Final
  - ~5% Flexgrade (participation, attitude, effort)
- CS4411 Breakdown
  - ~90% Projects
  - ~10% Flexgrade
- Grading will not be done on a curve
  - It is our goal to be able to give everyone an A+
  - Help us achieve this

Projects in CS4411

- Projects will be done in two-person teams
  - You may indicate a desired partner
  - If they also indicate you, we will pair you up
  - If you don't have a preferred partner, we'll pair you up with someone suitable

- Working in groups
  - Start early, time management is key
  - Manage the team effort
  - Part of what you are supposed to learn is how to manage to get work done in a small team

MiniProjects in 4410

- There will be approximately 4-6 miniprojects
- MiniProjects will be done individually
- Working individually
  - Start early, time management is key

Academic Integrity and Honor Code

- All submitted work must be your own
  - All homeworks must be your own independent work
  - OK to study together
  - Cannot share solutions, ever
- Project groups submit joint work
  - All group assignments must represent solely the work of the two people in that group
  - Cannot be in possession of someone else's solution
- Violations will be prosecuted to the fullest extent
- Closed-book exams, no calculators

Our Expectations

- Code of Silence
  - Absolute quiet during lectures
  - If you have a question, please speak up
    - Chances are 100% that someone else has the same question
- No electronics, Luddite zone
  - Scientific studies show that such classrooms are far more effective

Enrollment SNAFU

- There is great demand for CS courses in general, this course in particular
- The department has decided that the only way to manage the demand is to award the limited slots according to need
  - I was not consulted during the capping process
  - I did not implement the caps
  - Having said that, I don't know of any other way to manage the demand
- There was an error where the caps were set wrong, as result, some people who enrolled in April had to be unenrolled
  - The instructors had nothing to do with this!

Enrollments

- If you are enrolled now, congrats!
  - If you decide to drop the course, please do so promptly so others can step in!
- If you’re not yet enrolled yet, you need to place your name on a department-managed waitlist
- The department, not the instructors, will triage the waitlist and send enrollment PINs

Prerequisite

- CS3410 or equivalent required. Or permission of instructor.
- Required means required.
- If you did not take CS3410 or equivalent, you must contact course staff, explain your situation and request permission.

Draft Syllabus

- Introduction
- Architectural Support for Operating Systems
- Processes and Threads
- Scheduling
- Synchronization, Mutual Exclusion, Spin Locks, Semaphores, Condition Variables
- Deadlocks, Detection and Avoidance
- Memory Management
- Networking, LANs, WANs, Ethernet, ARP, IP, UDP, TCP
- Disks and RAID
- Filesystems, UFS, LFS
- Security

Draft Project Plan for CS4411

- Threads and Concurrency
- Scheduling
- Basic Datagram Networking
- Reliable Streaming Protocols
- Routing
- Filesystems

About Prof. Sirer

- Interests: OSes, distributed systems, self-organizing and peer-to-peer systems
- Bitcoin, flaws within, fixes to
- Sailing, wind, sea, woodworking (aka body modification)

About Prof. George

- Cornell grad student through 2013, lecturer since 2013
- Took 4410 a decade ago!
- Interested in designing and building systems and reasoning about code.
- Just taught 4410 over the summer

Questions?

- And demographics…