

CSE 506

Graduate OS

Introduction

Don Porter

Why Grad OS?

- ✦ Primary Goal: Demystify how computers work

An example progression

- ✦ Undergrad OS:
 - ✦ High-level understanding of paging
 - ✦ Theoretical issues like fragmentation
- ✦ Grad OS (506): Build a pager
 - ✦ Solid understanding of how paging SW + HW work
- ✦ Advanced Grad OS (624): Read novel research papers
 - ✦ Do creative things with paging: virtualization, security, etc

506: Learn by doing

- ✦ You will write major chunks of your own OS
 - ✦ Memory management, context switching, scheduler, file system, IPC, network driver, shell, etc.
 - ✦ Linux scheduler:
 - ✦ Difficult to understand just by reading source
 - ✦ Small modifications require first understanding the code
 - ✦ Impossible to replace/reimplement
 - ✦ No substitute for building it yourself!

JOS

- ✦ Developed at MIT, used at several top schools
 - ✦ The “J” is for Josh Cates, not Java
- ✦ In C and Assembly, boots on real PC hardware
 - ✦ You get the skeleton code, fill in interesting pieces
- ✦ Build the right intuitions about real OSes
 - ✦ but with much simpler code

Labs, cont.

- ✦ This course is **coding intensive**
 - ✦ You should know C, or be prepared to remediate quickly
 - ✦ You will learn basic, inline x86 assembly
 - ✦ You must learn on your own/with lab partner
- ✦ The lab is difficult, but worthwhile
 - ✦ You will want to commemorate, with a T-shirt, tattoo, etc.

Last Lab

- ✦ Includes open ended project
 - ✦ Can add significant feature to JOS
 - ✦ Or do a research task on another system
- ✦ Plan ahead – proposals due 11/08

CSE 522

- ✦ This course can also count as your MS project course (CSE 522)
- ✦ Requirements: Same as 506, except:
 - ✦ You must do the labs alone
 - ✦ You must complete 1 challenge problem in each lab
- ✦ To enroll: you must first be in 506
 - ✦ Ask me and I will have you moved to 522

No Textbook

- ✦ You're welcome
- ✦ Several recommended texts
 - ✦ Several free on SBU safari online site
 - ✦ Others on reserve at library
 - ✦ Required readings will mainly be papers you can print out

Lectures

- ✦ Compare and contrast JOS with real-world OSes
 - ✦ Mostly Linux, some Windows
- ✦ Supplement background on hardware programming
 - ✦ Common educational gap between OS and architecture

Prerequisites

- ✦ Undergrad OS
 - ✦ In some cases, industry experience is ok
 - ✦ Worth brushing up if it has been a while
 - ✦ In-class quiz, due before you leave
 - ✦ *If you can't answer 50% of these questions, consider ugrad OS*
- ✦ C programming
- ✦ Basic Unix command-line proficiency
- ✦ See me if you have already done the JOS lab, or similar

Space in the class

- ✦ Wait list is currently full
- ✦ Grad students often over-enroll
 - ✦ Space likely to open up in first week
 - ✦ If you want in, keep showing up for a few lectures
- ✦ Worst case: Prof. Zadok teaching 506 in spring
 - ✦ Likely to be offered every semester going forward

Course email list

- ✦ Sign up at <http://lists.cs.stonybrook.edu/mailman/listinfo/csc506>
- ✦ This is the primary announcement medium
- ✦ And for discussions about course work
 - ✦ Do not post code here or other solutions
 - ✦ Goal: Everyone can learn from general questions
- ✦ Material discussed on the mailing list can be an exam question

Other administrative notes

- ✦ Read syllabus completely
- ✦ Subscribe to the class mailing list
- ✦ 2 exams cover: lectures, labs, mailing list
- ✦ Every student will get a VM for lab work
 - ✦ You may use your own computer, staff can't support it
- ✦ All staff email goes to cs506ta@cs.stonybrook.edu
 - ✦ Except private issues for instructor only

Lab partners

- ✦ Self-selected
 - ✦ Lab mailing list good for finding them
- ✦ Same for entire course
 - ✦ Changes only with instructor permission
- ✦ Can work alone, but better with help
- ✦ List partner in first lab submission

Academic Integrity

- ✦ I take cheating very seriously. It can end your career.
- ✦ In a gray area, it is your job to stay on right side of line
- ✦ Never show your code to anyone except your partner and course staff
- ✦ Never look at anyone else's code (incl. other universities)
- ✦ Do not discuss code; do not debug each other's code
- ✦ Acknowledge students that give you good ideas

Lateness

- ✦ Each group gets 72 late hours
 - ✦ List how many you use in slack.txt
 - ✦ Each day after these are gone costs a full letter grade on the assignment
- ✦ It is your responsibility to use these to manage:
 - ✦ Holidays, weddings, research deadlines, conference travel, Buffy marathons, release of the next Zelda game, etc.
- ✦ 3 Exceptions: illness (need doctor's note), death in immediate family, accommodation for disability

Lab 1 assigned

- ✦ Due Friday, 9/9 at 11:59 pm, eastern.
- ✦ Instructions on website
- ✦ Quick demo

Getting help

- ✦ TA's (TBD) will keep office hours
- ✦ Instructor keeps office hours
 - ✦ Note that "by appointment" means more time available on demand

Questions?