CS 450: Operating Systems Lecture 1: Introduction

Spring 2014, J. Sasaki Dept of Computer Science Illinois Institute of Technology

1/13: Updated office hours

Introduction

- Dr. Jim Sasaki
- sasaki@iit.edu
- SB 110
- Office hours Spring 2014
 - Mon, Tue, Thu 1:30 pm 2:30 pm
 - Mon & Thu 12:45 1:45; Wed 1:45 2:45
 - Other times available

Today

- Prerequisites
- Resources
- Coursework
- Course Topics
- Many thanks to Michael Lee for letting me steal borrow some of his notes...

Prerequisites

Operating Systems Concepts

- Process
- Concurrency, exceptional control flow
- Memory hierarchy
- Virtual memory
- Files and I/O (file descriptors)

System-Level APIs

- Syscalls for
 - Process management
 - Exceptional control flow
 - Input/Output
 - Interprocess Communication

Programming

- C and/or Python
- Some kind of assembly
 - Familiarity, not expert-level
- Runtime stack

Resources

Websites

- Course: http://cs.iit.edu/~cs450
- Piazza: http://piazza.com
- Blackboard: http://blackboard.iit.edu

Textbook



 Note: 9th edition, not 8th edition

Coursework

Assignments

- 6 or so
- Written paper
- Quantitative analysis
- Simulation
- Coding

Exams

- Midterm and Final Exams
- Will Final exam be comprehensive?
 - Probably not
- One sheet of notes (both sides)
 - Closed book, no other notes

End-of-Semester Score

- Assignments: 50%
 - Not all assignments may be worth same amount
- Midterm and Final Exams: 25% each
- "Curving" (instructor option) add positive constant to everyone's EOS score so that class average is ≥ middle C

Letter Grade Cutoffs

- A: ≥ 90; B: 80–89; C: 70–79; D: 60–69%, E: < 60%
- To pass course you must also
 - Earn \geq 50% of EOS assignment points
 - Earn \geq 50% on each exam

Course Topics

Theory / Implementation

- Way too much material to be able to cover both theory and implementation.
- Even a relatively simple operating system like version 6 Unix took (less than) 10,000 lines of code
- cf. > 10 M lines for modern OS kernels

https://en.wikipedia.org/wiki/ File:Unix_history-simple.svg



18

OS Topics

- User view?
- Internal Responsibilities?

OS Services

OS as Providing Services to Users

- User interface Most visible part of OS
 - Old: Batch interface
 - Command-line interface
 - Unix "shell"
 - GUI interface
 - Mouse, keyboard, touch

OS as Providing Services to Programs

- Run programs: Compile, link, execute
- I/O operations, files, databases
- Data communication: networks, messages

OS as Resource Allocator & Guardian

- Services as resources
 - CPU, devices, files, databases, networks
 - Track usage (accounting)
 - Protect resources internal consistency
 - Error detection/avoidance, repair
 - Secure resources tampering, interference

OS Relies on Hardware

- The OS is software but the hardware has to support it.
- Some things can't be done at all without hardware support.
 - Interrupts, timers, test-and-set
- Some things can't be done quickly enough without hardware support
 - Virtual memory, MMU, graphics processors, direct memory access