CS 4410
Operating Systems

Introduction & Logistics

Mike George (in for Elisavet Kozyri)
Summer 2013
Cornell University
Welcome!

Today:

• Logistics
• Introduction
  • Motivation
  • What is an OS?
  • Issues in OS Design
  • Why Learn Operating Systems?
• Draft schedule
Logistics

- **Class dates:**
  - May 20 - June 28, 2013
- **Days/times:**
  - M-F 8:30 AM - 9:45 AM
  - Upson Hall 207
- **Instructor:**
  - Elisavet Kozyril (with Mike George filling in for week 1)
  - Office Hours: MWF 10:00 AM - 11:00 AM, 331 Upson Hall
  - Email: ekozyri or mdgeorge at cs.cornell.edu
- **Webpage:**
  - www.cs.cornell.edu/CS4410/2013su/
Logistics

- Prior Knowledge
  - CS 3410, CS 3420
  - Programming experience
  - Computer architecture

- Course Readings
  - Operating System Concepts (8th Edition) Silberschatz, Galvin and Gagne

- Lectures
  - Intensive
  - Interactive
  - Please attend!
Logistics

- Homework
  - weekly assigned
  - independent work
  - theoretical + practical + review
  - synchronized with the lectures

- Mini-test
  - fifteen-minute in-class test
  - related to the previous assignment
  - easy, short, preparation time = 0

- Final Exam
  - June 28, 8:30 am
  - Upson Hall 207
Logistics

• Grading
  • from A+ to F
  • non-curved
    - 5% at instructor's discretion
      (participation, mini-presentation, etc)
    - 10% mini-tests
    - 40% assignments
    - 50% final exam
  • Remember: The target is the knowledge, not the grade!

• Academic Integrity
Just built a processor...want to write “Hello World”.

- Step 1: look up specs for the display
- Step 2: write subroutine to send commands to printer
- Step 3: Buy a new display
- Step 4: Rewrite your program

Welcome to 1950

- Libraries for I/O
Now I want to run two programs

- Only one set of:
  - Registers (pc)
  - Memory
  - Devices
- Multi-tasking
  - Run one program at a time
  - Switch between them
Enter the Operating System

- Abstraction
  - Provide uniform interface
- Virtualization
  - Make one device seem like many
- Isolation
  - Prevent programs from stomping on each other
- Access Control
  - Applications still need to control system
What is an Operating System?

Users
- Mary
- John

Applications
- Web-browser
- Word Processor
- Video Game

Operating System
- Scheduler
- Monitor Driver
- Network Driver
- Memory manager
- Disk manager

Hardware
- CPU
- Memory
- Disk
- Network card
- Monitor
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?

- As a user (programmer)
  - Abstractions aren't perfect
- As a developer
  - Lots of new OSs
- A new world
  - No “cloud OS”
  - Back to the 1950s
Lectures Schedule

- Hardware
- Processes
- Threads
- CPU scheduling
- Synchronization
- Deadlocks
- Memory management
- Virtual memory
- Disks
- File systems
- Network
- Security