

Lecture 0

Course Introduction

Prof. Yinqian Zhang

Spring 2024

General Course Info

- Course Name: Operating Systems
- Term: Spring 2024 (Feb. 19 ~ Jun. 8, 2024)
- Time: Tue. 14:00–15:50
- Lecturer: Prof. Yinqian ZHANG (yinqianz@acm.org)
- Lab instructor: Ms. Yun SHEN (sheny@mail.sustech.edu.cn)

Self Introduction

Open Office Hour:
Tues. 9:00 ~ 11:00

- A professor
 - of Computer Science and Engineering at SUSTech
 - Leads TEECERT Labs
 - Previously at Ohio State University in USA
- A security researcher
 - Study the security of computer architectures, operating systems, and software applications
 - Do research on clouds, blockchains, IoTs
 - Find vulnerabilities in CPUs and software
 - Build systems to defend against various attacks

Content of This Course

- We will study the theory of operating systems
 - Virtualization, concurrency, persistence
 - Management of CPU, memory, I/O and storage
- We will learn the implementation of operating systems
 - Build an operating system kernel by yourself (uCore)
 - Run your OS on (emulated) RISC-V CPU (via QEMU)

Goals of This Course

- Be competent with process concepts and CPU scheduling.
- Be competent with memory hierarchy and memory management.
- Be familiar with process control blocks, system calls, context switching, interrupts, and exception control flows.
- Be familiar with process synchronization, inter-process communication, and threads.
- Be familiar with multi-threaded programming.
- Be familiar with file systems, disk scheduling algorithms and I/O.
- Be exposed to security

Prerequisites

- C Programming
 - All course projects will use C (or assembly)
 - OS kernel is written in C
- Data structure and algorithm analysis
 - It helps you to understand OS concepts, e.g., LRU
 - It assists you to design efficient OS components
- Computer Organization
 - How do computers work?
 - Hardware and software coordination

Reference Books

- **Operating System Concepts**, 9th Edition, Abraham Silberschatz et. al. (a.k.a. **the Dinosaur Book**)
- **Operating Systems: Three Easy Pieces**, Remzi H. Arpaci-Dusseau and Andrea C. Arpaci-Dusseau
 - <https://pages.cs.wisc.edu/~remzi/OSTEP/>

Course Structure

- Lectures (2 hours)
 - Get the main ideas and concepts (mostly in English, may repeat in Chinese if needed)
 - lectures will not follow textbooks (but read textbooks will help you understand better)
 - Download slides (before class, and check for updates after)
 - Take notes and ask questions
- Lab (2 hours)
 - Tutorials on kernel code and lab assignments
 - Do the lab exercises / projects
- Lectures and labs are integrated
 - Content are mostly sync-ed
 - All assignments (even written ones) submitted through labs

Gradings

- Lecture participation: 10%
 - Attendance or/and quiz
 - Get 100 points in quiz
- Lab participation: 15%
 - In-class assignments
- Assignment: 30%
 - Written and coding
- Mid-term exam: 20%
- Final exam: 25%

Academic Misconduct

- If an undergraduate assignment is found to be plagiarized, the first time the score of the assignment will be 0.
- The second time the score of the course will be 0.
- If a student does not sign the Assignment Declaration Form or cheats in the course, including regular assignments, midterms, final exams, etc., in addition to the grade penalty, the student will not be allowed to enroll in the two CS majors through 1+3, and **cannot receive any recommendation for postgraduate admission exam exemption** and all other academic awards.

Keys to Success

- Come to lectures/lab tutorials
 - Announcement may be made in class without further notice
 - Make the lectures interactive
 - Take notes, ask questions, take attendance
 - Content in exams \subseteq **what is taught in class** \supseteq content on slides $\not\subseteq$ textbook
- Visit Blackboard
 - Announcement
 - Lecture slides (before classes, may update afterwards)
 - Tutorial / lab exercises
 - Solutions
- Prepare to code
 - You will build your own OS kernel!!!
 - Seek help in time!!!

Thank you!

