ICSI 412 Operating Systems

University at Albany - Computer Science
Chongqing University of Posts and Telecommunications
Computer Science, International College

Instructor: Dr. Jackson Marques de Carvalho
Email: Jmarquesdecarvalho@albany.edu
Term: Spring 2025.
This syllabus is subject to change at the instructor's discretion.

Co-instructors: Dr. Chang
changgh@cqupt.edu.cn
Dr. Yin Li
yinli@cqupt.edu.cn

COURSE/CATALOG DESCRIPTION

[3 hours] (Formerly I CEN/I CSI 400.) Design and implementation of operating systems. Topics include historical overview of operating systems and their principles, systems programming, file system organization and I/O, resource allocation, memory management and virtual memory, virtual machines, process scheduling and synchronization of concurrent processes, deadlock detection and prevention, and security. Only one of I CEN/I CSI/I ECE 400, I CSI 412 may be taken for credit. Prerequisite(s): I CSI 404 or I ECE 233 (formerly I ECE 333). Prerequisite(s): ICSI/I CEN 333.

COURSE OVERVIEW

This course addresses the basic concepts of operating systems, common to most computer systems, which interface the machine with the programmer/user. This class will introduce the processing unit, process management, synchronization, memory management, and I/O management. The course will also cover an introduction to the policies for scheduling, deadlocks, memory management, synchronization, and file systems.

OBJECTIVES

The successful student will be able to understand the basic components of a computer operating system, and the interactions among the various components.

Student Learning Objectives:

The students will be able to:

1. Describe the basic components of an operating system and their role in implementation for general purpose applications.
2. Define the concepts of processes, threads, asynchronous signals and competitive system resource allocation.
3. Show how a shell works.
4. Explain what multi-tasking is and outline standard scheduling algorithms for multi-tasking.
5. Discuss mutual exclusion principles and their use in concurrent programming, including semaphore construction and resource allocation.
6. Give an overview of system memory management.
7. Explain how file systems are implemented.
8. Discuss the features and strengths of various contemporary operating systems.

TEACHING STRATEGIES

Lecture goal: The instructor will deliver lectures where students are encouraged to actively participate in discussions as part of the learning activities. Some of the materials regarding lectures will be available. They are provided as a courtesy by the course instructor. Students are cautioned that only studying the course notes for the course is not a suitable substitute for attending class.

REQUIRED TEXTS

Silberschatz, Galvin, and Gagne, *Operating System Concepts*, 10th edition, Wiley, 2018
ISBN: 978-1-119-32091-3

RECOMMENDED TEXTS

Stallings, *Operating Systems*, 8th edition, Prentice-Hall, 2014.
Tanenbaum and Bos, *Modern Operating Systems* "MOS4", 4th edition, Pearson, 2014.
Kernighan and Ritchie, *The C Programming Language*, 2nd edition, Prentice-Hall, 1988.

POLICIES

Academic Integrity

It is every student's responsibility to become familiar with the standards of academic integrity at both CQUPT and University at Albany. Claims of ignorance, of unintentional error, or of academic or personal pressures are not sufficient reasons for violations of academic integrity. Any incident of academic dishonesty can result in *(1)* no credit for the affected assignment, project, or exam; *(2)* report to the appropriate University authorities (e.g., Dean of Undergraduate Education or Graduate Studies), and/or *(3)* a failing grade (E) for the course.

For all assignments and projects, make sure to do your own work, except where collaboration is explicitly permitted or required. Also, make sure that you properly cite any resource from which you borrow ideas and that you clearly distinguish them from your contributions.

Academic Dishonesty

Issues involving dishonesty are taken very seriously and are dealt with according to College and Department policy. Academic dishonesty includes:

1) Improper access to evaluation material or records.
2) Submission of material which is not the student's own work.
3) Conduct which interferes with the work or evaluation of other students.

Specifically, for this course, dishonesty involves:

1) Copying from another person, book, magazine, or other electronic or printed media.
2) Obtaining another person's exam answer or answers.
3) Assisting another student in submitting work that is not the student's own.

4) Any activity that falls under the general University at Albany definition.

Any act of academic dishonesty will result in a minimum penalty of a grade of zero (0) for that item for the first occurrence. An automatic E in the course will result for the second offense. This policy holds for homework assignments and programs, projects as well as for exams. In essence this policy applies to all work submitted by the student. Penalties will be applied to all parties involved.

Attendance Policies

Students are expected to attend every class meeting of courses in which they are registered. Only in specific, unavoidable situations absences from class will be excused. Unavoidable situations include illness, death in the family, religious observances, participation in university- sponsored activities, government-required activities, and any other absence which the professor approves.

COURSE EXPECTATIONS

1. Students are expected to attend classes in adherence to the University's Missed Class Policy.
2. Assignments are expected to be submitted on the date and according to the specific instructions included in the description of the assignment. On-time assignments must be submitted electronically according to instructions included in the assignment and by the due date provided in the assignment. **No assignment will be accepted after its due date**.
3. All Exams will be announced about one week in advance. Students are expected to be present for all exams. A make-up exam will only be given in the event of an emergency, and **only if the instructor or the co-instructor is informed in advance.** In case you cannot contact the instructor or co-instructor directly, you are to contact the instructor or co-instructor through email or call the Department/College and leave a message for the instructor or co-instructor. Failure to notify either the instructor or the co-instructor prior to missing an exam will result in a zero for the exam. Whether an absence is "excusable" is determined by the instructor. A makeup exam may have the grading weights adjusted.
4. If a student is unable to appear for an exam due to serious and unavoidable circumstances, he/she must immediately contact the instructor or co-instructor and submit a written letter with supporting documents at the earliest opportunity.

GRADING

There are five basic components that will be used to determine a student's grade in the course:

- Labs/Quizzes 15%
- Class participation 5%
- Programming Projects 15%
- Exam 1 25%
- Exam 2 40%

COMMUNICATION GUIDELINES

1. Emails will be answered within reasonable time frames during the daytime on weekdays.

LECTURE TOPICS

Each week covers one or more topic(s). Lectures will cover fundamentals, not details. Students must read the text, class notes and other reference material thoroughly to learn details. The list of topics may be updated/altered as the term progresses. Also, the sequence by which topics will be covered may not necessarily be as presented here. The course will address as many of the following topics as time will allow.

Chapter 1 - Introduction

Chapter 2 – Operating-Systems Structures

Chapter 3 – Processes
- 3.1 Process Concept
- 3.2 Process Scheduling
- 3.3 Operations on Processes
- 3.4 Interprocess Communication
- 3.5 IPC in Shared-Memory Systems
- 3.7 Examples of IPC Systems
  - 3.7.1 POSIX Shared Memory
  - 3.7.4 Pipes
  - 3.7.4.1 Ordinary Pipes
- 3.9 Summary

Chapter 5 – CPU Scheduling
- 5.1 Basic Concepts
- 5.2 Scheduling Criteria
- 5.3 Scheduling Algorithms
- 5.7 Operating-Systems Examples
  - 5.7.1 Example: Linux Scheduling
- 5.8 Algorithm Evaluation
  - 5.8.1 Deterministic Modeling
- 5.9 Summary
- 5.9 Summary

Chapter 4 – Threads & Concurrency
4.1 Overview
4.3 Multithreading Models
4.4 Thread Libraries
4.4.1 Pthreads

Chapter 6 – Synchronization Tools
- 6.1 Background
- 6.2 Critical-Section Problem
- 6.3 Peterson's Solution
- 6.4 Hardware Support for Synchronization
- 6.4.2 Hardware Instructions
- 6.6 Semaphores
- 6.8 Liveness
- 6.8.1 Deadlock
- 6.8.2 Priority Inversion

Chapter 7 – Synchronization
- 7.1 Classic Problems of Synchronization
- 7.1.1 The Bounded-Buffer Problem
- 7.3 POSIX Synchronization
- 7.3.2 POSIX Semaphores

Chapter 8 – Deadlocks
- 8.1 System Model
- 8.2 Deadlock in Multithreaded Applications
- 8.3 Deadlock Characterization
- 8.3.1 Necessary Conditions
- 8.3.2 Resource-Allocation Graph
- 8.4 Methods for Handling Deadlocks
- 8.5 Deadlock Prevention
- 8.6 Deadlock Avoidance

Chapter 9 – Main Memory
- 9.1 Background
- 9.1.1 Basic Hardware
- 9.1.2 Address Binding
- 9.2 Contiguous Memory Allocation
- 9.3 Paging
- 9.4 Structure of the Page Table
- 9.5 Swapping
- 9.6 Example: Intel 32- and 64-bit Architectures

Chapter 10 – Virtual Memory
- 10.1 Background
- 10.2 Demand Paging
- 10.4 Page Replacement
- 10.6 Thrashing
- 10.9 Other Considerations
- 10.9.3 TLB Reach
- 10.9.4 Inverted Page Tables
- 10.9.5 Program Structure

Chapter 11 – Mass-Storage Structure
- 11.1 Overview of Mass-Storage Structure
- 11.1.1 Hard Disk Drives
- 11.2 HDD Scheduling
- 11.2.1 FCFS Scheduling
- 11.2.2 SCAN Scheduling
- 11.2.4 Selection of a Disk-Scheduling Algorithm

Chapter 12 - I/O Systems
- 12.1 Overview

12.2 I/O Hardware
12.2.2 Polling
12.2.3 Interrupts
12.2.4 Direct Memory Access
12.2.5 I/O Hardware Summary

Chapter 13 – File-System Interface
13.1 File Concept
13.3 Directory Structure

Chapter 14 File-System Implementation
14.4 Allocation Methods

Appendix C
C.7 File Systems