

WAYNE STATE UNIVERSITY

COLLEGE OF ENGINEERING

Computer Science Department

CSC 4420 - Computer Operating Systems

Winter 2020

Course Description:

Operating system is an essential software layer to use various various computing devices. This course covers the basic components and design principles of modern operating systems, including process and thread, CPU scheduling, memory management, file system and others. This course also provides hand-on programming experiences of using Linux system calls, and design and implementation of a cloud-based file system.

Administrative Information:

Meeting Days: Mon /Wed
Meeting Time: 11:30 AM - 12:45 PM
Meeting Place: Alex Manoogian Hall Room: 0091

Course Web Site: <http://lihaoxu.eng.wayne.edu/Courses/CSC4420/>

Instructor: Lihao Xu
Address: 5057 Woodward, Floor 14, Suite 1401.1
Email: lihao@wayne.edu
Office Hours: Mon, Wed, 10:30 AM – 11:15 AM or by appointment.

Lab - Section 001:

GTA/Lab instructor: Azam Peyvandipour
Email: azam.peyvandipour@wayne.edu
Office Hours: Mon & Tue, 5:00pm – 6:00pm
Address: 5057 Woodward, Floor 3, Room 3210

Lab - Section 002:

GTA/Lab instructor: Rui Chen
Email: ruichen@wayne.edu
Office Hours: Wed, Thu, Fri, 1:30-2:00pm
Address: 5057 Woodward, Floor 2, Room 2212

Co-requisite:

CSC 4421, Computer Operating Systems Laboratory, is a co-requisite for CSC 4420. You **must** elect the lab in order to take CSC 4420.

Prerequisites:

CSC 3100/3101, Computer Architecture and Organization and CSC 2200/2201, Computer Science II. Exceptions will be granted by permission of the instructor only.

In addition, you should have experience designing and implementing software using a standard set of development tools, preferably in the Linux/UNIX environment (Linux is the target platform for the CSC 4420 assignments). Ideally, this experience includes familiarity with typical software development tools such as a compiler (gcc preferred), debugger (gdb preferred), and a Linux editor (such as vi or emacs). As for languages, you may use either C or C++. I will try to present examples as much as possible in C++. If you have experience only with C++, you will find it useful to gain some familiarity with the core C constructs and the C standard libraries (libc), since many books present examples using C.

Textbook:

- **Modern Operating Systems**, Andrew S. Tanenbaum, 4th Edition, Pearson, 2015

References:

Operating System Concepts, 8th Edition, Wiley, 2008.

Understanding the Linux Kernel, 3rd Edition, O'Reilly Media, 2005.

Advanced UNIX Programming, 2nd Edition, Pearson, 2004.

Advanced Programming in the UNIX Environment, Addison-Wesley, 1992.

Course Learning Outcomes:

The aim of the course is to enable students to understand a set of abstractions and constructs that are useful in many large-scale software systems, not just operating systems. Specifically, upon course completion, the students should be able to:

1. Understand the historical development of operating systems to control complex computing systems, and understand the impact of system software on individuals, organizations and the society at large.
2. Learn the structure and operations of an Operating System both theoretically and practically by studying widely used modern operating systems (e.g., Linux) techniques and code.
3. Develop a comprehensive understanding of different Operating Systems management primitives, such as thread, process, and resource management such as scheduling, paging;
4. Learn how to use OS through system calls
5. Exposed to protection and security from an Operating System point-of-view
6. Hands-on experiences through a cloud and file system project

Student Outcomes:

- a. An ability to apply knowledge of computing and mathematics appropriate to the discipline.
- b. An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution.
- c. An ability to design, implement and evaluate a realistic computer-based system, process, component, or program to meet desired needs.
- d. An understanding of professional, ethical, legal, security, and social issues and responsibilities.
- e. An ability to communicate effectively with a range of audiences.
- f. Recognition of the need for, and an ability to engage in, continuing professional development.
- g. An ability to use current techniques, skills, and tools necessary for computing practices.
- h. An ability to design and implement a practical system using existing tools

Brief list of topics to be covered:

- OS Basic
- Interfaces and System Calls
- Processes and Threads
- CPU Scheduling
- Process Synchronization
- Memory Management
- File System
- I/O
- Security
- Cloud

Assessment:

Assessment of how well outcomes are being achieved will be done by applying a rubric to a random sample of at least 25% of the students who have completed the work being used for assessment. Assessment tools are homeworks, lab assignments, and projects. For each outcome being assessed, each student in the sample will be judged to (a) Exemplary Competence, (b) Basic Competence (c) Approaching Competence, or (d) Beginning Competence to meet an objective standard designed to assess this outcome. The thresholds used are: Basic Competence (70%), Exemplary Competence (80%). We will say that this offering of the course

achieved the particular outcome if and only if 70% or more of the students sampled were assessed to be in categories (a) or (b):

Objectives	Outcomes	Assessment Method
1	a, d	Homework
2	b	Homework, Labs
3	g	Homework, Labs
4	c, g	Homeworks, Labs, Project
5	b, f	Project
6	e, h	Project

Assignments:

Homeworks

There will be four homeworks, each with theoretical and lab components, in addition to the lab session assignments.

Project

There will be one large scale design and implementation project:

Design and Implementation of A Secure Amazon S-3 Based File System

See details at the course web site.

Evaluation:

Homeworks: 50%

Project: 50%

Grading:

90% - 100% A

80% - 84% B+

70% - 74% B-

62% - 64% C

85% - 89% A-

75% - 79% B

65% - 69% C+

60% - 61% C-

Course Communication:

All lecture notes, slides and homeworks will be posted at the course web site. All homeworks are to be submitted via blackboard.

Policies:

Submitting Assignments:

Homework assignments, programming projects, and project must be completed by the scheduled date. Submissions will be through the blackboard drop box. Late homework assignments/projects will be penalized 20% per day (except in cases of illness or emergency), and will not be accepted 3 days after the due date. That is, you will have up to three late days which you may allocate as you wish to accommodate minor illnesses, religious holidays, work conflicts, computer / network failures, etc. Additional late days will be granted only for MAJOR illnesses/emergencies for which you have documentation, such as a doctor's note.

The following reasons are **not** considered major emergencies: The computers / network / printers were down, you had a job interview, you had job responsibilities, you lost all your files. Starting early and keeping backups of your work can avoid these problems.

Collaboration and Cheating:

I expect programming assignments to reflect your own effort. Acts of cheating and unacceptable collaboration will be dealt with using university policy. Cheating is when you copy, with or without modification, someone else's work that is not meant to be publicly accessible. Unacceptable collaboration is the knowing exposure of your own exam answers or project solutions, or the use of someone else's exam answers or project solutions made public.

At the same time, I encourage students to help each other learn the course material. As in most courses, there is a boundary separating these two situations. You may give or receive help on any of the concepts covered in lecture or discussion and on the specifics of C++. You are allowed to consult with other students about the conceptualization of a project. However, all written work, whether in scrap or final form, must be done by you. You are not allowed to work out the programming details of the problems with anyone or to collaborate to the extent that your programs are identifiably similar. You are not allowed to look at or in any way derive advantage from the existence of solutions prepared in prior terms, whether these solutions are copies of former students' work or solution sets handed out by course staff. I will be using an automated program to correlate projects against each other and past solution sets.

Using publicly available code, such as that posted in web tutorials or in textbooks without giving an attribution is also a violation of academic integrity. Therefore, if you use code from a textbook or the web, you must state what sources you consulted. If you use such code without telling me, you will receive a 0 for the project. If you use such code and you do tell me, your grade may still be reduced, since I expect you to do independent work. That is, the more independent your programming efforts are, the more points you will receive. Each programming assignment will come with guidelines about what I expect as far as consulting additional resources.

If you have any questions as to what constitutes unacceptable collaboration or exploitation of prior work, please talk to the instructor right away. You are expected to exercise reasonable precautions in protecting your own work. Don't let other students borrow your account or computer, don't leave your program in a

publicly accessible directory, and take care when discarding printouts. If your program is substantially the same as someone else's, I will find out about it and you will receive a grade of 0 for the programming assignment. **Collaboration with other students in the class when permitted will be communicated.** Needless to say, exams are an individual effort!

Incompletes:

I will hold to university policy on incompletes. This policy states that incompletes are only to be given for emergencies and when there is only a minor amount of work remaining at the end of the semester. Accordingly, I will not give incompletes for the following reasons: You are not happy with your performance in the class and want to work on projects more (See me early if you need help). You have a medical problem early in the semester (This situation is addressed by a medical withdrawal. You can re-take the course next year). You have job conflicts (Sorry. This does not qualify as an emergency).

Because of the extraordinary variety of religious affiliations of the University student body and staff, the Academic Calendar makes no provisions for religious holidays. However, it is University policy to respect the faith and religious obligations of the individual. Students with classes or examinations that conflict with their religious observances are expected to notify their instructors well in advance so that mutually agreeable alternatives may be worked out.

Student Disabilities Services

If you have a documented disability that requires accommodations, you will need to register with Student Disability Services for coordination of your academic accommodations. The Student Disability Services (SDS) office is located in the Adamany Undergraduate Library. The SDS telephone number is 313-577-1851 or 313-202-4216 (Videophone use only). Once your accommodation is in place, someone can meet with you privately to discuss your special needs. Student Disability Services' mission is to assist the university in creating an accessible community where students with disabilities have an equal opportunity to fully participate in their educational experience at Wayne State University.

Students who are registered with Student Disability Services and who are eligible for alternate testing accommodations such as extended test time and/or a distraction-reduced environment should present the required test permit to the professor at least one week in advance of the exam. Federal law requires that a student registered with SDS is entitled to the reasonable accommodations specified in the student's accommodation letter, which might include allowing the student to take the final exam on a day different than the rest of the class.

Schedule and Topics Covered

See the course web site for details.