CS 3113 – Intro to Operating Systems – Spring 2025

Syllabus

Instructor: Dr. Sridhar Radhakrishnan, sridhar@ou.edu

Office Location: Devon Energy Hall, DEH 244
Office Hours: 4:30 PM to 6:00 PM (Tuesday and Thursday).

Class Location: Nielsen Hall 170
Class Time: 9:00 AM to 10:15 AM (Tuesday and Thursday)

Course Prerequisite: C S 2413 or C S 2414, and C S 2813 or MATH 2513, and C S 2614 or ECE 3223.

Textbook: Abraham Silberschatz, Peter Baer Galvin, Greg Gagne: Operating System Concepts, 10th Edition. Wiley 2018, ISBN 978-1-118-06333-0

Learning Management System/Website: https://canvas.ou.edu/courses/388236

Course Catalog Description: An introduction to the major concepts and techniques of designing and implementing operating systems, including memory management, process management, information management, and computer security. Principles of performance evaluation. Class projects require the design and implementation of software systems. A UNIX family operating system will be used.

Course Format: This course will be in a traditional in-class format. There may be some remote component from time to time where video recordings of lectures will be made available on the course canvas page.

Course Activities, Assignments, and Assigning Grades: Students are required to take two exams and a final. There will be no makeup exams except in cases of emergencies. Failure to take the final exam will result in an automatic F as the overall course grade. There will be a set of five programming projects that each student should individually complete, and all programming projects must be written in C/C++ and will be specified. The course letter grade will be assigned based on the overall percentage: >= 90 (A), >=80 and < 90 (B), >=70 and < 80 (C), >=60 and < 70 (D), and <60 (F). The allocation of percentages is given below:

	Percentages
Attendance (class roll will be taken in each class via Top Hat)	15%
Exam 1	20%
Exam 2	20%
Final	25%
Programming Projects	30%

ABET Outcomes of Instruction in CS 3113:

> Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline.

Belonging Statement: The University of Oklahoma fosters an inclusive culture of respect and civility, belonging, and access, which are essential to our collective pursuit of excellence and our determination to change lives. The unique talents, perspectives, and experiences of our community enrich the learning, and working environment at OU, inspiring us to harness our innovation, creativity, and collaboration for the advancement of people everywhere.

CS 3113 is a learning community where curiosity and interaction are highly valued. As your instructor, I strongly believe that asking questions and engaging in discussions are essential for mastering operating systems. Students are encouraged to actively participate, challenge concepts, and collaborate, as this subject is best learned through dialogue and shared exploration.

Course Expectations: Students are expected to engage actively in learning by attending all lectures, participating in discussions, and asking questions to deepen their understanding. A curious and open-minded approach will help in exploring complex operating system concepts. Programming assignments must be completed independently, requiring students to develop a strong foundation in C/C++ and apply it to implement OS concepts. Proper coding standards, documentation, and adherence to project specifications are essential, and students must uphold academic integrity by avoiding code sharing.

To perform well in exams and assessments, students should stay current with readings, review lecture notes, practice problem-solving, and seek clarification on difficult topics. Effective time management is crucial—meeting deadlines, allocating sufficient time for debugging, and avoiding last-minute work will contribute to success in programming projects. Constructive collaboration is encouraged; students should discuss concepts and problem-solving approaches with peers while maintaining individual accountability and respecting diverse viewpoints.

Students must adhere to all course policies, including submission guidelines, academic integrity rules, and classroom conduct expectations. Seeking assistance from the instructor or TAs is encouraged, but students are ultimately responsible for their own learning. Lastly, critical thinking and problem-solving skills are essential for this course. Students should analyze OS challenges, evaluate trade-offs, and apply theoretical knowledge to real-world scenarios through hands-on programming exercises. Meeting these expectations will help students gain a deep understanding of operating systems and develop skills that are valuable in both academics and industry.

Course Goals: The goals of this course are to:

1. Provide a Strong Theoretical Foundation in Operating Systems
   - Introduce students to the core principles, design, and implementation of modern operating systems.
   - Cover key concepts such as process management, memory management, file systems, security, and system performance.
2. Develop Practical Programming Skills for OS Concepts
   - Enable students to write and analyze C/C++ programs that interact with various OS components.

    - Reinforce theoretical concepts through hands-on programming assignments.
3. Enhance Understanding of Process and Resource Management
    - Teach students how operating systems handle multiple processes, scheduling, synchronization, and inter-process communication.
    - Provide insights into concurrency challenges and solutions.
4. Build Competency in Memory and Storage Management
    - Help students understand how operating systems allocate, manage, and protect memory using techniques like paging and segmentation.
    - Explore file system structures, disk management, and I/O handling.
5. Expose Students to OS Security and Protection Mechanisms
    - Introduce key security principles, including access control, authentication, and protection mechanisms within an OS.
    - Discuss real-world threats and mitigation strategies.
6. Develop Critical Thinking for System Design and Optimization
    - Encourage students to evaluate trade-offs in OS design choices and analyze system performance.
    - Foster problem-solving skills for optimizing operating system performance.
7. Prepare Students for Advanced Courses and Industry Applications
    - Equip students with the knowledge needed for advanced coursework in systems, networking, security, and distributed computing.
    - Provide skills relevant to real-world software development, system administration, and research in computing.

Learning Objectives: Upon successful completion of this course, students will be able to:

1. Understand Core Operating System Concepts
    - Explain the fundamental principles of operating systems, including process management, memory management, file systems, and security.
2. Process and Thread Management
    - Describe and implement concepts related to processes, threads, and concurrency, including context switching, scheduling algorithms, and synchronization mechanisms.
3. CPU Scheduling and Synchronization
    - Analyze different CPU scheduling algorithms and evaluate their efficiency under various workload conditions.
    - Apply synchronization techniques such as locks, semaphores, and monitors to solve concurrency problems.
4. Memory Management and Virtual Memory
    - Explain memory allocation strategies, paging, segmentation, and virtual memory concepts.
    - Implement basic memory management techniques in C/C++.
5. File Systems and Storage Management
    - Describe and compare different file system architectures and access methods.
    - Implement file I/O operations and understand disk scheduling techniques.
6. Security and Protection Mechanisms
    - Understand fundamental security concerns in operating systems, including access control, authentication, and system protection techniques.
7. Programming for OS Concepts in C/C++

   - Write C/C++ programs to implement and experiment with various operating system components, including process creation, inter-process communication (IPC), threading, memory management, and file system operations.

8. Performance Analysis and Optimization
   - Apply principles of performance evaluation to analyze OS algorithms and optimize system performance.
9. Develop Critical Thinking and Problem-Solving Skills
   - Identify and solve real-world operating system-related challenges through programming assignments and projects.

Course Reflection Survey: You'll receive a Course Reflection Survey at the end of each semester for each course that you are enrolled in. I strongly encourage you to complete this survey. Your feedback can help me adjust my class for future semesters to help other students be successful. Your feedback is confidential and I will only receive it after final grades are due. Course Reflection Survey results may also factor into teaching evaluations and annual performance reviews and are shared with department and program chairs.

Copyright Statement: Sessions of this course may be recorded or live-streamed. These recordings are the intellectual property of the individual faculty member and may not be shared or reproduced without the explicit, written consent of the faculty member. In addition, privacy rights of others such as students, guest lecturers, and providers of copyrighted material displayed in the recording may be of concern. Students may not share any course recordings with individuals not enrolled in the class or upload them to any other online environment.

Tentative Course Schedule—Note that some lectures will be recorded video lectures, and we will announce this in advance.

Date	Topics	Projects
January 14, 2025	Introductions; Chapter 1: OS Overview;	Project 1 Assigned
January 16, 2025	Chapter 1 (contd); Chapter 2: OS Structures	
January 21, 2025	Chapter 2: OS Structures	
January 23, 2025	Chapter 3: Processes	
January 28, 2025	Chapter 3: Processes	
January 30, 2025	Chapter 4: Threads and Concurrency	
February 4, 2025	Chapter 4: Threads and Concurrency	Project 1 Due; Project 2 Assigned
February 6, 2025	Chapter 5: CPU Scheduling	
February 11, 2025	Chapter 5: CPU Scheduling	
February 13, 2025	Chapter 6: Synchronization Tools	
February 18, 2025	Chapter 6: Synchronization Tools	
February 20, 2025	Chapter 7: Synchronization Examples	
February 25, 2025	**Exam – 1**	Project 2 Due; Project 3 Assigned
February 27, 2025	Chapter 8: Deadlocks	
March 4, 2025	Chapter 8: Deadlocks	
March 6, 2025	Chapter 9: Main Memory	
March 11, 2025	Chapter 9: Main Memory	
March 13, 2025	Chapter 9: Main Memory	Project 3 Due; Project 4 Assigned
March 18, 2025	**Spring Break**	
March 20, 2025	**Spring Break**	
March 25, 2025	Chapter 10: Virtual Memory	
March 27, 2025	Chapter 10: Virtual Memory	
April 1, 2025	**Exam – 2**	
April 3, 2025	Chapter 11: Mass Storage Structure	Project 4 Due; Project 5 Assigned
April 8, 2025	Chapter 11: Mass Storage Structure	
April 10, 2025	Chapter 12: I/O Systems	
April 15, 2025	Chapter 13: File-System Interface	
April 17, 2025	Chapter 14: File-System Implementation	
April 22, 2025	Chapter 15: File-System Internals	
April 24, 2025	Chapter 16: Security	Project 5 Due
April 29, 2025	Chapter 17: Protection	
May 1, 2025	Chapter 18: Virtual Machines	
May 9, 2025	**Final Examination; 8:00 AM – 10:00 AM (Friday) Nielson Hall 170**	

Course Policies

Programming Projects:

Projects must be coded in C or C++. We will use ANSI C or C++, so if your program compiles with any G++ or GCC compiler, you are set to go.

1. You will also use the Gradescope facility to submit the source program.
2. For every 24 hours late, you will be deducted 10% of the grade of the programming project. Any project that is more than 5 days late will not be evaluated.
3. A programming project that does not meet the specifications will receive an automatic 50% grade deduction.
4. You are better off submitting a working project on the fifth day rather than one that does not work on the due date.
5. Programs have to be documented clearly. Programs that lack or are weak in documentation will receive a deduction of up to 30% of the grade. Follow the documentation methods used in programs in your data structures book.
6. You will demo your project to the grader during the grader-assigned special office hours **if the grader so wishes**. Graders are not responsible for debugging your programs.
7. The project specifications presented by the instructor may not contain all the implementation details. It is your responsibility to understand the specifications thoroughly. Please ensure that all relevant questions regarding the project are asked during class time.
8. **Copying programs or consulting others for coding is strictly prohibited and will be treated as plagiarism. Additionally, copying programs from the Internet is also strictly prohibited. All projects are individual projects; hence, you are required to work independently without help from others.**
9. In addition to the above general evaluation policies, each programming project will have a set of specifications that must be met.

Exams: All exams will be conducted on paper in class. All exams will be closed books, notes, laptops, or electronic devices. Failure to take the final exam will result in an automatic F as the course grade.

Incompletes: The grade of "I" is intended for the rare circumstance when a student who has been successful in a class has an unexpected event shortly before the class's end. We will not consider giving a student a grade of "I" unless the following three conditions have been met:

1. It is within two weeks of the end of the semester.
2. The student has a grade of C or better in the class.
3. The reason that the student cannot complete the class is properly documented and compelling.

Classroom Conduct: Because cell phones and laptops can distract substantially from the classroom experience, students are asked not to use either during class except when required as part of a classroom exercise. Disruptions of the class will also not be permitted. In the case of disruptive behavior, we may ask that you leave the classroom and charge you with violating the Student Code of Responsibilities and Conduct. Examples of disruptive behavior include:

- Allowing a cell phone or pager to repeatedly beep audibly.
- Playing music or computer games during class in such a way that they are visible or audible to other class members.
- Exhibiting erratic or irrational behavior.
- Behavior that distracts the class from the subject matter or discussion.
- Making physical or verbal threats to a faculty member, teaching assistant, or class member.
- Refusal to comply with faculty or teaching assistant direction.

Proper Academic Conduct: Feel free to discuss all assignments with the instructors or the TAs. Code (projects and assignment): you may discuss code solutions with other students. However:

1. You may not look at or share code with others;
2. If you discuss a solution with anyone, you must document their names in your assignment.
3. You must document this in your code if you use external resources (e.g., Piazza, StackOverflow.com, ChatGPT, or other LLMs).

Make sure that your computer account is properly protected. Use an appropriate password, and do not give your friends access to your account or computer system. Do not leave printouts, computers, or thumb drives around a laboratory where others might access them. Programming projects will be checked by software designed to detect collaboration. This software is extremely effective and has withstood repeated reviews by the campus judicial processes.

Upon the first documented occurrence of inappropriate collaborative work or of taking a solution from a network resource, the instructors will report the academic misconduct to the Campus Judicial Coordinator. The procedure is documented in the University of Oklahoma Academic Misconduct Code (http://integrity.ou.edu). The provider and receiver of a solution will be treated equally in the misconduct process.

Generative AI Policy: Students are permitted to use Generative AI tools, such as ChatGPT and Copilot, as aids in understanding operating system concepts, debugging code, and generating ideas for programming assignments. However, all work submitted must reflect the student's own understanding and effort. If Generative AI is used to assist in coding, problem-solving, or explanation, students must explicitly acknowledge its use by adding comments in their code or submissions detailing what aspects were AI-assisted. Directly copying AI-generated solutions without comprehension or modification is not acceptable and may be considered academic misconduct. This course aims to develop independent problem-solving skills, so while AI can be a helpful tool, it should complement learning rather than replace critical thinking and hands-on programming experience. Students should consult the instructor if they have questions about appropriate AI usage.

Attendance Policy: Regular attendance is essential for success in this course, as active participation in lectures and discussions is crucial to understanding operating system concepts. Attendance will be recorded in each class using Top Hat, and it will account for 15% of the overall course grade. Students are responsible for ensuring their attendance is accurately recorded. Absences due to emergencies or university-approved events must be communicated in advance when possible, and proper documentation may be required for consideration. Missing class without a valid excuse may negatively impact the final grade. While lecture recordings may be available for some sessions, they are not a substitute for in-class participation. Students are encouraged to engage actively in discussions and ask questions, as interaction is a key component of learning in this course.

University Policies

Mental Health Support Services

Support is available for any student experiencing mental health issues that are impacting their academic success. Students can either been seen at the University Counseling Center (UCC) located on the second floor of Goddard Health Center or receive 24/7/365 crisis support from a licensed mental health provider through TELUS Health. To schedule an appointment or receive more information about mental health resources at OU please call the UCC at 405-325-2911 or visit University Counseling Center. The UCC is located at 620 Elm Ave., Room 201, Norman, OK 73019.

Title IX Resources and Reporting Requirement

The University of Oklahoma faculty are committed to creating a safe learning environment for all members of our community, free from gender and sex-based discrimination, including sexual harassment, domestic and dating violence, sexual assault, and stalking, in accordance with Title IX. There are resources available to those impacted, including: speaking with someone confidentially about your options, medical attention, counseling, reporting, academic support, and safety plans. If you have (or someone you know has) experienced any form of sex or gender-based discrimination or violence and wish to speak with someone confidentially, please contact OU Advocates (available 24/7 at 405-615-0013) or University Counseling Center (M-F 8 a.m. to 5 p.m. at 405-325-2911).

Because the University of Oklahoma is committed to the safety of you and other students, and because of our Title IX obligations, I, as well as other faculty, Graduate Assistants, and Teaching Assistants, are mandatory reporters. This means that we are obligated to report gender-based violence that has been disclosed to us to the Institutional Equity Office. This means that we are obligated to report gender-based violence that has been disclosed to us to the Institutional Equity Office. This includes disclosures that occur in: class discussion, writing assignments, discussion boards, emails and during Student/Office Hours. You may also choose to report directly to the Institutional Equity Office. After a report is filed, the Title IX Coordinator will reach out to provide resources, support, and information and the reported information will remain private. For more information regarding the University's Title IX Grievance procedures, reporting, or support measures, please visit Institutional Equity Office at 405-325-3546.

Reasonable Accommodation Policy

The University of Oklahoma (OU) is committed to the goal of achieving equal educational opportunity and full educational participation for students with disabilities. If you have already established reasonable accommodations with the Accessibility and Disability Resource Center (ADRC), please submit your semester accommodation request through the ADRC as soon as possible and contact me privately, so that we have adequate time to arrange your approved academic accommodations.

If you have not yet established services through ADRC, but have a documented disability and require accommodations, please complete ADRC's pre-registration form to begin the registration process. ADRC facilitates the interactive process that establishes reasonable accommodations for students at OU. For more information on ADRC registration procedures, please review their Register with the ADRC web page. You may also contact them at (405)325-3852 or adrc@ou.edu, or visit www.ou.edu/adrc for more information.

Note: disabilities may include, but are not limited to, mental health, chronic health, physical, vision, hearing, learning and attention disabilities, pregnancy-related. ADRC can also support students experiencing temporary medical conditions.

Religious Observance

It is the policy of the University to excuse the absences of students that result from religious observances and to reschedule examinations and additional required classwork that may fall on religious holidays, without penalty. [See Faculty Handbook 3.15.2]

Adjustments for Pregnancy and Related Issues

Should you need modifications or adjustments to your course requirements because of pregnancy or a pregnancy-related condition, please request modifications via the Institutional Equity Office website or call the Institutional Equity Office at 405/325-3546 as soon as possible. Also, see the Institutional Equity Office FAQ on Pregnant and Parenting Students' Rights for answers to commonly asked questions.

Final Exam Preparation Period

Pre-finals week will be defined as the seven calendar days before the first day of finals. Faculty may cover new course material throughout this week. For specific provisions of the policy please refer to OU's Final Exam Preparation Period policy.

Emergency Protocol

During an emergency, there are official university procedures that will maximize your safety.

Severe Weather: If you receive an OU Alert to seek refuge or hear a tornado siren that signals severe weather.

1. Look for severe weather refuge location maps located inside most OU buildings near the entrances.
2. Seek refuge inside a building. Do not leave one building to seek shelter in another building that you deem safer. If outside, get into the nearest building.
3. Go to the building's severe weather refuge location. If you do not know where that is, go to the lowest level possible and seek refuge in an innermost room. Avoid outside doors and windows.
4. Get in, Get Down, Cover Up
5. Wait for official notice to resume normal activities.

Additional Weather Safety Information is available through the Department of Campus Safety.

The University of Oklahoma Active Threat Guidance

The University of Oklahoma embraces a Run, Hide, Fight strategy for active threats on campus. This strategy is well known, widely accepted, and proven to save lives. To receive emergency campus alerts, be sure to update your contact information and preferences in the account settings section at one.ou.edu.

RUN: Running away from the threat is usually the best option. If it is safe to run, run as far away from the threat as possible. Call 911 when you are in a safe location and let them know from which OU campus you're calling from and location of active threat.

HIDE: If running is not practical, the next best option is to hide. Lock and barricade all doors; turn of all lights; turn down your phone's volume; search for improvised weapons; hide behind solid objects and walls; and hide yourself completely and stay quiet. Remain in place until law enforcement arrives. Be patient and remain hidden.

FIGHT: If you are unable to run or hide, the last best option is to fight. Have one or more improvised weapons with you and be prepared to attack. Attack them when they are least expecting it and hit them where it hurts most: the face (specifically eyes, nose, and ears), the throat, the diaphragm (solar plexus), and the groin.

Please save OUPD's contact information in your phone.

NORMAN campus: *For non-emergencies call (405) 325-1717. For emergencies call (405) 325-1911 or dial 911.*

TULSA campus: *For non-emergencies call (918) 660-3900. For emergencies call (918) 660-3333 or dial 911.*

Fire Alarm/General Emergency

If you receive an OU Alert that there is danger inside or near the building, or the fire alarm inside the building activates:

1. *LEAVE* the building. Do not use the elevators.
2. *KNOW* at least two building exits
3. *ASSIST* those that may need help
4. *PROCEED* to the emergency assembly area
5. *ONCE safely outside, NOTIFY first responders of anyone that may still be inside building due to mobility issues.*
6. *WAIT* for official notice before attempting to re-enter the building.

OU Fire Safety on Campus