



CS 569: Special Topics in Software Engineering: Program Analysis & Evaluation



Instructor: Manish Motwani
Email: motwanim@oregonstate.edu
Personal website: mmotwani.com
Office: KEC 3043
Office hours: TBD
Term: Winter 2024 (1/8/24–3/15/24)
Time: Monday and Wednesday 12–1:50 PM
Room: **BAT 250 - Batcheller Hall 250 (GP)**
Credit hours: 4

Course Description: Software engineering involves a wide range of activities, including understanding customer requirements, planning the system to be designed, collaborating in teams to develop the software, performing maintenance, and conducting various analysis and evaluation tasks like testing, verification, and validation. In CS 569, we will explore cutting-edge research techniques that simplify the analysis and evaluation of software.

Throughout the term, we will cover the following topics: static and dynamic program analyses; software testing strategies and automated test generation; software quality metrics; verification and validation techniques for software correctness.

A significant portion of this course will involve students becoming acquainted with and participating in cutting-edge research. There are three key research components in this course:

1. Lectures and assignments will incorporate the latest research findings in software engineering.
2. Students, either individually or in pairs, will engage in reviewing and presenting existing research from recent top-tier conferences and journals to the class. They will summarize the research in both written and oral presentations and lead class discussions.
3. Students, either individually or in groups of up to three, will undertake a term-long research project that will provide additional hands-on experience in using state-of-the-art techniques.

Mandatory Prerequisites: None

Recommended Prerequisites:

- In preparation for taking this course, we recommend that you visit our [Advantageous Skill Assessment](#) to determine which skills you may wish to refresh or acquire prior to the start of classes.
- CS 561

Student Learning Outcomes: Apply modern static and dynamic program analysis techniques to identify issues in code. Automatically generate tests using state-of-the-art test generation techniques to verify software correctness. Develop skills to analyze and identify the key contributions and limitations of a research paper and present those insights to the peer audience. Develop a group project that will provide hands-on experience in using state-of-the-art techniques.

Reading: There is no required textbook for the course. Reading assignments will come from published research papers. Students who wish to read established textbooks beyond the assigned reading should consider:

- The Mythical Man-Month: Essays on Software Engineering by Frederick P. Brooks Jr.; Addison Wesley
- Software Architecture: Foundations, Theory, and Practice by Richard N. Taylor, Nenad Medvidovic, and Eric Dashofy; John Wiley and Sons
- Fundamentals of Software Engineering by Carlo Ghezzi, Mehdi Jazayeri, Dino Mandrioli; Prentice Hall
- Software Testing and Analysis: Process, Principles, and Techniques, by Mauro Pezzè and Michal Young, John Wiley and Sons

Academic Calendar: All students are subject to the registration and refund deadlines as stated in the Academic Calendar: <https://registrar.oregonstate.edu/osu-academic-calendar>.

Statement Regarding Students with Disabilities: Accommodations for students with disabilities are determined and approved by Disability Access Services (DAS). If you, as a student, believe you are eligible for accommodations but have not obtained approval please contact DAS immediately at 541-737-4098 or at <http://ds.oregonstate.edu>. DAS notifies students and faculty members of approved academic accommodations and coordinates implementation of those accommodations. While not required, students and faculty members are encouraged to discuss details of the implementation of individual accommodations.

Student Conduct Expectations link: <https://beav.es/codeofconductLinks>

Student Bill of Rights: OSU has twelve established student rights. They include due process in all university disciplinary processes, an equal opportunity to learn, and grading in accordance with the course syllabus: <https://asosu.oregonstate.edu/advocacy/rights>.

Reach Out for Success: University students encounter setbacks from time to time. If you encounter difficulties and need assistance, it's important to reach out. Consider discussing the situation with an instructor or academic advisor. Learn about resources that assist with wellness and academic success at <https://oregonstate.edu/ReachOut>. If you are in immediate crisis, please contact the Crisis Text Line by texting OREGON to 741-741 or call the National Suicide Prevention Lifeline at 1-800-273-TALK (8255).

Nondiscrimination policy: Software engineering is at its nature a collaborative activity and it benefits greatly from diversity. This course includes and welcomes all students regardless of age, background, citizenship, disability, sex, education, ethnicity, family status, gender, gender identity, geographical origin, language, military experience, political views, race, religion, sexual orientation, socioeconomic status, and work experience. Our discussions and learning will benefit from these and other diverse points of view. Any kind of language or action displaying bias against or discriminating against members of any group, or making members of any group uncomfortable are against the mission of this course and will not be tolerated. The instructor welcomes discussion of this policy, and encourages anyone experiencing concerns to speak with him.

Academic integrity: Students are allowed to work together on all aspects of this class. However, for the homework assignments, each student must submit his or her own write up, clearly stating the collaborators. Your submission must be your own. When in doubt, contact the instructors about whether a potential action would be considered plagiarism. If you discuss material with anyone besides the class staff, acknowledge your collaborators in your write-up. If you obtain a key insight with help (e.g., through library work or a friend), acknowledge your source and write up the summary on your own. It is the student's responsibility to remove any possibility of someone else's work from being misconstrued as the student's. Never misrepresent someone else's work as your own. It must be absolutely clear what material is your original work. Plagiarism and other anti-intellectual behavior will be dealt with severely. Note that facilitation of plagiarism (giving your work to someone else) is also considered to be plagiarism, and will carry the same repercussions. Students are encouraged to use the Internet, literature, and other publicly-available resources, except the homework solutions and test (including quizzes, midterms, finals, and other exams) solutions, from past terms' versions of this course and other academic courses, whether at OSU and at other institutions. To reiterate, the students are not allowed to view and use past homework and test solutions, unless explicitly distributed by the CS 569 staff as study material. Whenever students use Internet (including **ChatGPT**), literature, and other publicly-available resources, they must clearly reference the materials in their write ups, attributing proper credit. This cannot be emphasized enough: attribute proper credit to your sources. Failure to do so will result in a zero grade for the assignment and possibly a failing grade for the class, at the instructor's discretion. Copying directly from resources is not permitted, unless the copying is clearly identified as a quote from a source. Most use of references should be written in the words of the student, placing the related work in proper context and describing the relevant comparison.

Grading:

Class Participation	10%
Paper presentations	20%
Homeworks	30%
Project	40%

Tentative Course Outline:

The weekly coverage might change as it depends on the progress of the class. However, you must keep up with the reading assignments, homework and project.

Week	Day & Date	Content	Reading	Homework	Project
Week 1	Mon, 1/8/24 Wed, 1/10/24	Course introduction Program analysis overview			
Week 2	Mon, 1/15/24 Wed, 1/17/24	No class (MLK day) Static program analysis		Homework 1	Lit review/topic selection
Week 3	Mon, 1/22/24 Wed, 1/24/24	Dynamic program analysis Software testing strategies			
Week 4	Mon, 1/29/24 Wed, 1/31/24	Automated test generation Software verification and validation			
Week 5	Mon, 2/05/24 Wed, 2/07/24	Software quality assurance Project idea presentations		Homework 2	
Week 6	Mon, 2/12/24 Wed, 2/14/24	Paper presentations Paper presentations			Mid term project report
Week 7	Mon, 2/19/24 Wed, 2/21/24	Paper presentations Paper presentations			
Week 8	Mon, 2/26/24 Wed, 2/28/24	Mid project presentations Paper presentations		Homework 3	
Week 9	Mon, 3/04/24 Wed, 3/06/24	Paper presentations Guest lecture			Final project report
Week 10	Mon, 3/11/24 Wed, 3/13/24	Guest lecture Final project presentations		-	