EE 282 Computer Systems Architecture

https://web.stanford.edu/class/ee282/

Lectures:
Monday & Wednesday, 3:00 - 4:20 PM, Shriram 104

Instructor: Caroline Trippel
Email: trippel@stanford.edu
Office Hours: Wednesdays (4:20 - 5:20 PM) in Gates 470

Instructor: Christos Kozyrakis
Email: christos@cs.stanford.edu
Office Hours: Mondays (4:20 - 5:20 PM) in Gates 444

Review Sessions:
Friday, 2:30PM - 3:20PM, in Skilling Auditorium and on Zoom

Teaching Assistant: Noah Anderson
Email: noaha@cs.stanford.edu
Office Hours: Thursdays 9:00 - 10:30 AM on Zoom
Wednesdays 12:00 - 2:00 PM in Huang Basement

Teaching Assistant: Kira Tran
Email: ktran20@stanford.edu
Office Hours: Mondays 12:00 - 2:00PM in Huang Basement
(Mondays 12:00 - 1:00PM on Zoom)

Teaching Assistant: Jane Nguyen
Email: janen9@stanford.edu
Office Hours: Tuesdays 1:00 - 3:00 PM in Huang Basement
Tuesdays 2:00 - 3:00 PM on Zoom

Course Overview

EE282 covers key hardware and software topics that are relevant for architecting advanced computer systems. Examples include: out-of-order processors, multi-level cache hierarchies, main memory systems, multithreading and vector techniques, GPUs and accelerators, virtualization, virtual memory, datacenter architecture, advanced IO systems, and non-volatile storage. Programming assignments include: simulation of multi-core processor designs to characterize performance, design and implementation of advanced processor optimizations, and formal verification of an open-source processor.

This class is offered for 3 units as either a letter grade or a CR/NC course.

Prerequisite: EE180 or an equivalent class. You are expected to understand the fundamentals of computer systems, pipelining and basic caching, assembly programming, and virtual memory.You are ***heavily recommended*** to take this prerequisite before taking EE282. If you have not filled this prerequisite, most likely, you will have a hard time catching up and will end up with a poor learning experience and bad grade. If it has been some time since you took the prerequisite, we suggest that review the following material from the class textbook: Appendix A.{1-8}, Appendix B.{1-3}, and Appendix C.{1-2, 4, 6}. Students are also expected to have programming experience with C, C++, or other similar languages. CS110 or CS140 or equivalent classes on operating systems are useful but not required.

EE282 is appropriate for undergraduate and graduate students specializing in the broad field of computer systems. It is also appropriate for other EE and CS students who want to understand, program, and make efficient use of modern computer systems of any scale in their day-to-day work. Through active engagement and completion of course activities, you will be able to:

- Identify core design paradigms that are foundational to architecting advanced computer systems and how/when they are used, including: pipelining, parallelism, out-of-order execution, prediction (or speculation), locality and caching, indirection, amortization, redundancy, specialization, focus on the common case (for efficiency), focus on the uncommon case (for security).
- Explain and relate together the basics of advanced processor design, including core and memory systems architecture.
- Describe the basics of specialized hardware design, including the architecture of GPUs, AI Processors, and other accelerators.
- Understand the basics of hardware support for datacenter workloads, including hardware support for virtualization and networking, and connect how these hardware features interact with low-level systems software.
- Investigate optimal specifications by conducting design space exploration for a multi-core processor implementation using an architecture simulator.
- Implement advanced processor optimizations and integrate into an existing design in SystemVerilog.
- Formally verify parts of an open-source processor implementation using commercial-grade model-checkers.

Contacts

We will use Ed Discussion (https://edstem.org/us/courses/96863) for Q&A on all class material. We will check the forum regularly, especially close to assignment deadlines and exams. When posting a question, make sure you use an appropriate subject. For example, if your question refers to the definition of memory latency in Problem 2 of the Problem Set, an appropriate subject would be "PS, Problem 2: Definition of memory latency".

Before posting a question, check to see if this question has already been asked/answered. Consolidating

follow-ups to the same basic question in the same thread will help us stay organized and reduce confusion on your end. You are welcome to discuss questions with your classmates using the forum as long as you do not violate the Stanford honor code.

For questions that are not appropriate for posting publicly to the forum, you can send a private message to the course staff on Ed or directly email them—we prefer the former as a first step, since we monitor Ed regularly. For any course-related emails, please prefix the subject with "[ee282]".

Course Materials

The class website (https://web.stanford.edu/class/ee282/) is the primary way to share links to important course materials, such as the up-to-date schedule, reading assignments, lecture notes, homework assignments, and solutions. Please **check the website frequently** since new information will be added regularly.

Ed Discussion (https://edstem.org/us/courses/96863) will be used both for online discussion and for announcements. We recommend keeping email notifications on for Ed Discussion (https://edstem.org/us/settings/) to avoid missing any time-sensitive announcements.

All course materials linked from the website will redirect you to Canvas (https://canvas.stanford.edu/courses/226130). Lecture recordings will also be available here, since this is a CGOE course.

We will use Google Forms with TinyURL to facilitate in-class discussions and quizzes.

You will need to have access to a PC to complete the programming components of homework assignments. You will also need internet connection so that you can access email, class website, Ed discussion, and Canvas. Students can borrow equipment and access other learning technology from the Lathrop Learning Hub.

This course has required readings from **Hennessy & Patterson, Computer Architecture: A Quantitative Approach, 7th edition**. We have reserved two copies of the textbook for your use at the Terman Engineering Library. The tentative schedule for the required readings are included at the end of this syllabus.

Coursework and Grading Scheme

This course is worth 3 units of credit, which means you may be asked to complete 9 hours of work per week, including class time. The class schedule in the last section of the syllabus highlights assignments and due dates so that you can plan your work for the quarter accordingly.

Participation

You are expected to attend the lectures in person. This is your opportunity to ask questions, contribute answers and insights, and affect the nature of the class. Also note that the lecture notes and the textbook are not guaranteed to capture 100% of every topic discussed aloud during lecture. From our experience, students who do not attend lectures find exam questions to be very difficult.

The lecture notes will be made available before class. During the lectures, there will also be in-class quizzes which you will be given time to work on. You will submit answers via

Google Forms. CGOE students are expected to complete the in-class quizzes on Gradescope within 1 week after the lecture recording is available.

Assignments: Assignments include a Problem Set and 3 Programming Assignments.

The Problem Set consists of 15 problems. Each problem within the set has a reference lecture and the corresponding due date is put up on the lecture schedule. Note that due dates are per question.

You will work on all assignments (i.e., all Problem Set questions and Programming Assignments) in groups of 2 students. All assignments are due by 11:59PM PT on the dates indicated on the assignment on GradeScope. **There will be no late days.** In the case of an unexpected or sudden emergency, please reach out to the teaching staff. Solutions to Problem Set questions will be available online approximately a day after their deadline.

Exams:

There will be one online midterm and one in-person final exam.

- The Midterm **will be given asynchronously on Tuesday, May 5** and will be designed to take 90-180 minutes. You may take the exam at any point in the 24-hour window from 12:00AM to 11:59PM PT on May 5th. The midterm will cover lectures 1 through 10.
- The Final will be designed to take 120-180 minutes and will be given **in person on Saturday, June 6**, 8:30 – 11:30 AM in Shriram 104. The final will cover lectures 1-19.

The CGOE staff will disseminate instructions for taking exams as a CGOE student.

Final grades will be *tentatively* computed as follows:

Category	Grade %	Notes
Attendance and Participation	5%	• 3 conditional absence "passes" (see Course Policies); • Arrival/departure >10 min late/early counts as an absence
Problem Set	5%	
3 Programming Assignments	15%	
Midterm	15%	
Final	60%	

Course Policies

Attendance: Non-CGOE students get **3 "passes"** to be physically absent from class without losing

participation points, provided that you:

- watch the recording; and
- complete and submit the in-class exercises within a week.

Arriving/departing >10 minutes late/early to/from a class counts as an absence.

We understand circumstances may arise that will interfere with your ability to attend class, such as feeling sick or needing to quarantine for the safety of everyone. In light of these considerations, we request that you:

- skip class if you are feeling sick, especially if you have any COVID-19 symptoms;
- watch the lecture recording posted on Canvas;
- complete the in-class quiz asynchronously;
- remember that you have 3 absence "passes," which are explicitly intended to cover these sorts of situations;
- meet with the course staff afterwards to ask questions and get caught up.

If you have already used up all late "passes" and are sick, please contact the course staff. Please do not come to class if you are ill! Our priority is to keep everyone healthy while maximizing your learning in this course.

Review Sessions: Review sessions will be held periodically throughout the quarter to provide additional guidance on Programming Assignments and help you prepare for the Final and Midterm. They will take place in-person and on Zoom and be recorded.

Regrade Requests: Requests for regrading must be submitted in writing within a week from the grade assignment date. Submit a note that describes the specific grading complaint and your assignment manuscript. However, regrading will affect the whole assignment/exam (not just one question) and can lead to a lower grade than the original one.

Collaboration: In general, collaboration is encouraged subject to the following guidelines:

- No more than 2 people can collaborate on the Problem Set or any Programming Assignment.
- Students working together should submit a single assignment for the pair.
- Any assistance received in the solution to the Problem Set question or a Programming Assignment should be acknowledged in writing with specific details.
- No sharing of code nor partial/complete solutions across groups is permitted.

See the honor code and collaboration for some general guidelines, which apply to both the Problem Set and Programming Assignments.

Honor Code: The Honor Code is taken seriously in this course and suspected violations will be referred to the Office of Judicial Affairs. Please refer to the honor code webpage at: https://communitystandards.stanford.edu/policies-guidance/honor-code.

You are encouraged to discuss assignments with other students. However, each student must independently write their own solution to homework assignments. Submitting solutions or code written by another person is a violation of the Honor Code.

Use of generative AI: Our class will follow guidance from the Board of Judicial Affairs regarding use of AI and the Stanford Honor Code, which notes that use of generative AI to "substantially complete" an assignment or exam by entering the prompt and submitting the output as one's own work

is not permitted. More concretely, this class permits the use of generative AI for small tasks, pending that you explicitly acknowledge whenever AI is used and for what, and using AI in a productive way is encouraged. Using AI to circumvent the course learning objectives is disallowed. If you are unsure about a particular use of AI or how to acknowledge its use, please reach out to the course staff.

Course Privacy: As noted in the university's recording and broadcasting courses policy, students may not audio or video record class meetings without permission from the instructor (and guest speakers, when applicable). If the instructor grants permission or if the teaching team posts videos themselves, students may keep recordings only for personal use and may not post recordings on the Internet, or otherwise distribute them. These policies protect the privacy rights of instructors and students, and the intellectual property and other rights of the university. Students who need lectures recorded for the purposes of an academic accommodation should contact the Office of Accessible Education.

COVID-19: Stanford University is no longer mandating the use of masks indoors in most circumstances. You can find the most current policies on campus masking requirements on the COVID-19 Health Alerts here.

Extended Absences: Despite our efforts to take precautions and protect ourselves and those around us, it is possible that one or more of us in the class will get sick, will need to give care to someone sick, or for other reasons will require an extended absence during the quarter.

- If a student requires an extended absence before more than 70% of coursework is completed, there might be opportunities to Withdraw from the course, or develop a schedule for making up and submitting coursework later in the quarter. Students in this situation should talk to a staff member at the Office of Accessible Education and to the instructor as soon as possible.
- If a student requires an extended absence after at least 70% of coursework is completed at a passing grade or higher, students may request an Incomplete. Incompletes do not award any credit and can drop students below the minimum required unit load. This could negatively impact academic progress, graduation, NCAA and Veteran's certifications, and financial aid. Students in this situation should talk to their section leader to discuss options.

Academic Accommodations

If you experience disability, please register with the Office of Accessible Education (OAE). Professional staff will evaluate your needs, support appropriate and reasonable accommodations, and prepare an Academic Accommodation Letter for faculty. To get started, or to re-initiate services, please visit oae.stanford.edu. The OAE is located at 563 Salvatierra Walk; phone: 650-723-1066. If you already have an Academic Accommodation Letter, we invite you to share your letter with us. Academic Accommodation Letters should be shared at the earliest possible opportunity so we may partner with you and OAE to identify any barriers to access and inclusion that might be encountered in your experience of this course.

Tentative Schedule: Lectures and Required Reading

[HP: Hennessy & Patterson, Computer Architecture: A Quantitative Approach, 7th edition; MC: Morgan Claypool Synthesis Lectures]

Date	[Lecture #] Topic	Reading Assignment	Reference Problem/s [Due date]
3/30	[1] Introduction, Review of prerequisites *Lecturer: Christos Kozyrakis*	[Prerequisite material] HP: Appendices A.{1-8}, B.{1-3}, C.{1-2, 4, 6} Sections 1.{1-13}	
4/1	[2] Advanced Caches I: Multi-level caches, Optimizations *Lecturer: Caroline Trippel*	HP: Sections 2.{1, 3, 6, 7}, Appendix B.{3, 6}	
4/6	[3] Advanced Caches II: Optimizations (continued), Prefetching *Lecturer: Caroline Trippel*	HP: Sections 2.5, 5.{1, 2}	Problems 1 & 2 [4/16]
4/8	[4] Coherence problem, Cache coherence *Lecturer: Caroline Trippel*	MC: A Primer on Memory Consistency and Cache Coherence, Second Edition: Chapters 2, 6 HP: Section 5.{3, 4}	Problem 3 [4/23] Prog. Assign. 1 [5/7]
4/13	[5] Synchronization *Lecturer: Caroline Trippel*	HP: Section 5.5	
4/15	[6] Memory Consistency *Lecturer: Caroline Trippel*	MC: A Primer on Memory Consistency and Cache Coherence, Second Edition: Chapters 3, 4, 5.{1-2} HP: Section 5.{6-7}	Problem 4 [4/23]
4/20	[7] Main Memory *Lecturer: Caroline Trippel*	HP: Section 2.2	Problem 5 [4/30]
4/22	[8] Out-of-Order Processors I: Dynamic Scheduling and Branch Prediction *Lecturer: Christos Kozyrakis*	HP: Sections 3.{1-5}	Problem 6 [4/30]
4/27	[9] Out-of-Order Processors II: Speculation and Superscalar *Lecturer: Christos Kozyrakis*	HP: Sections 3.{6-10}	Problem 7 [4/30] Prog. Assign. 2 [5/16]
4/29	[10] Multithreading *Lecturer: Christos Kozyrakis*	HP: Sections 3.{11}	Problem 8 [5/3]
5/4	[11] Vectors and SIMD *Lecturer: Christos Kozyrakis*	HP: Sections 4.{1-3, 5}, Appendix G.	Problem 9 [5/14]
5/5	Midterm Exam Time: 90-180 min, 12:00 AM - 11:59 PM	Lectures 1-10	
5/6	[12] GPUs *Lecturer: Michael Pellauer*	HP: Sections 4.{4, 6-8}	Problem 10 [5/21]
5/11	[13] AI Processors *Lecturer: Thierry Tambe*	Understanding Matrix Multiplication on a Weight-Stationary Systolic Architecture Roofline: an insightful visual performance model for multicore architectures Microscaling Data Formats for Deep Learning	Problem 11 [5/21]
5/13	[14] Virtualization *Lecturer: Christos Kozyrakis*	HP: Sections 6.{1-3} MC: HW & SW Support for Virtualization: Chapters 1-4 (optional)	Problem 12 [5/23]
5/18	[15] Advanced I/O & Networking *Lecturer: Christos Kozyrakis*	HP: Section 6.5 Appendices F.{1-2}; MC: HW & SW Support for Virtualization: Chapter 6 (optional)	Problem 13 [5/28] Prog. Assign. 3 [6/3]
5/20	[16] Advanced Memory Management *Lecturer: Caroline Trippel*	MC: HW & SW Support for Virtualization: Chapters 1, 2, 4, 5; HP: Appendix L	Problem 14 [5/28]
5/25	Memorial Day (No Classes)		
5/27	[17] Non-Volatile Memory	HP: Section 6.5, Appendix D.2; Flash storage memory	Problem 15 [6/4]

Date	[Lecture #] Topic	Reading Assignment	Reference Problem/s [Due date]
	Lecturer: Christos Kozyrakis		
6/1	[18] Datacenter Hardware/Management *Lecturer: Christos Kozyrakis*	HP Sections 6.{4, 6-9}	
6/3	[19] Microarchitectural Attacks and Defenses *Lecturer: Caroline Trippel*	Opening Pandora's Box: A Systematic Study of New Ways Microarchitecture Can Leak Private Data Spectre Attacks: Exploiting Speculative Execution	
6/6	Final Exam Time: 8:30AM - 11:30AM Venue: Shriram 104	Lectures 1-19	