Design and Analysis of Algorithms

Fall 2025

Instructor: Jasper Lee, `jasperlee@ucdavis.edu`

Graduate TA: Suraj Mathashery, `smathashery@ucdavis.edu`

(For all course admin) Course staff contact: `ecs122a-lee-fall25-staff@ucdavis.edu`

Undergraduate Tutors (Volunteers): Alan Buckser, Reina Itakura, Avery Li, Nivita Reddy

Lecture Time and Location: Tuesdays and Thursdays, 18:10–19:30 at Storer Hall 1322

Collaboration Hours/Discussions (Optional): See course webpage for the up-to-date schedule.

- Mondays 10:00-10:50, Wellman Hall 212
- Tuesdays 09:00-09:50, Olson Hall 205
- Thursdays 09:00-09:50, Hoagland Hall 113
- More TBA

Final Exam: 10 December 2025, 20:30-22:30 at Storer Hall 1322 (The time is set by the university and completely outside of the course staff's control)

Course Webpage: `https://jasperchlee.github.io/courses/122A/F25/index.html`

Piazza: `https://piazza.com/ucdavis/fall2025/ecs122aa/home` (Accessible via Canvas)

Textbook: The course will not follow any particular textbooks. However, the following text is recommended if you want a reference text for the concepts in the course:

- *Algorithms Illuminated: Omnibus Edition* by Tim Roughgarden, published by the Cambridge University Press

Contact the course staff if you have issues acquiring the text at reasonable/no cost.

Description

This course covers basic techniques and paradigms for the design and analysis of algorithms. The list of topics include:

- Divide and Conquer
- Dynamic Programming
- Greedy Algorithms
- Data structures
- NP-hardness

Prerequisites

This is an upper division undergraduate level course.

The prerequisites are

1. ECS 20 (Discrete Mathematics) or equivalent, and
2. ECS 60 or ECS 32B or ECS 36C or equivalent (Introduction to programming and basic algorithms and data structures)

As mandated by College of Engineering policy, prerequisites will be checked and enforced at the beginning of the quarter by the undergraduate advisors at the departmental level. The instructor has no control over this process.

Familiarity with basic linear algebra is useful and recommended for understanding some examples/applications in class, but not essential to do well in the course.

Learning Goals

Algorithms form the backbone of computer science. The main goal of this course is to introduce students to the design and rigorous analysis of algorithms. The overarching question addressed by the course content is "What are the right ways to *think* about algorithms?"

Design of algorithms Students will learn three important paradigms of algorithms design: divide and conquer, dynamic programming and greedy algorithms. The course will also touch on data structures, as well as the notion of NP-hardness for students to develop a "smell test" for hardness of computational problems.

Analysis of algorithms Students will learn to rigorously analyze algorithms, as a framework to guide the design of algorithms. Beyond analyzing algorithm runtime, the course will place particular emphasis on proving *correctness* of algorithms which is frequently surprisingly tricky. While most students will not need to write proofs in their future careers, the goal is to instill and develop broad intuition, "smell tests", and informal reasoning strategies that will be useful at large when designing algorithms.

Technical communication As a complementary goal, students will also learn to communicate technical algorithmic and mathematical ideas effectively. Students will learn these skills through collaboration with classmates, feedback on written assignments, as well as interactions with the instructor and TA/tutors in class and in collaboration hours.

Course Outline

The following is a rough course outline. See the course webpage for the most up-to-date lecture plan and schedule.

Introduction to algorithm analysis (1 lecture)

Divide and Conquer (3 lectures)

Dynamic Programming (4 lectures)

Greedy Algorithms (3 lectures)

Data Structures (2 lectures)

NP-hardness (2 lectures)

Miscellaneous topic(s) (2 lectures)

Grading

Each grade component will be curved by the instructor before a final grade is calculated. The percentages below are for reference, and the instructor will carefully examine all cases on grade boundaries before assigning the final grade.

Homeworks	40%
Midterm	25%
Final Exam	35%
Bonus: Extra Credit problems on homeworks	Maximum of 10%
Bonus: Showing up to Collaboration Hours once for HW1	5%

See the section on Collaboration Hours for details on the bonus 5%.

Homework Assignments

There are 5 written homework assignments (HW0, plus HW1-4) and 4 coding assignments (CA0, plus CA1-3). The homeworks are roughly weekly, with instructions on submission and deadlines.

With the exception of Homework 0 which should be submitted but graded only for completion, all homework submissions contribute to the homework component of the final grade as detailed in the previous section.

For the written assignments (except for HW0), students will team up in groups of 3 to write their homework submissions. Details about grouping will be posted on Piazza. HW0 is the exception: they should be submitted individually, although collaboration is allowed and encouraged.

Coding assignments will be completed and submitted individually. Discussion and collaboration with classmates are again both allowed and encouraged, but students must write their own code.

See the Collaboration Policy in the next section for further details.

There will also be extra-credit homework problems, which students are encouraged to solve, for further mastery of the material. Please refer to the instructions on each homework.

Assignments should be typeset neatly, in LaTeX or Word or otherwise (LaTeX highly recommended, with a template on the course webpage), or written and scanned in clearly legible handwriting. Unreadable submissions may not be graded.

Collaboration Hours and Discussion Sections

In place of traditional (and typically not very effective) office hours, this course will host a number of *collaboration hours* (collab hours for short) throughout the week. Discussion sections will also be run as collab hours. **Students should feel free to attend any collab hour/discussion section, regardless of which section they registered for.**

In collab hours, students come (ideally with their homework partner(s)) and collaborate with each other on solving homework problems, under course staff guidance. Homework problems, especially the extra credit problems, can be challenging. Collab hours help you through the problem solving process and make meaningful progress on homework problems, while ensuring that you do not just skip the problem solving by getting answers from your peers or the internet. If hours are not too busy, you are also welcome to come with algorithmic problems outside of course homeworks. **If you are stuck on a homework, come get help in collab hours!**

Collab hours (and discussion sections) are optional, but you are highly encouraged to attend to make progress on homework problems. In the instructor's experience, collab hours are a critical component to student learning given the hands-on problem solving nature of the sessions.

To encourage you to try out collab hours, the instructor will give a 5% bonus to any student who attends **any** collab hour between October 3 2025 and October 10 2025 (HW1 duration). To qualify for the bonus, you should have a meaningful interaction with a TA/tutor or the instructor in a collab hour, and take a selfie with that course staff member (who will judge whether the interaction has been sufficiently meaningful, erring on the side of being generous). Upload your selfie on Canvas to claim the bonus. If you feel that you do not need any help with the homework, to claim the bonus, you can also

- Demonstrate understanding of the homework problem by describing your solution to the course staff. (Much) more likely than not, the course staff member will have feedback on your presentation, and sometimes also on the correctness.
- Come to the collab hour with an interesting algorithmic problem you want help with, to discuss with the course staff! The course staff (including the instructor) may not necessarily be able to solve the problem on the spot, but we can also have a technical discussion brainstorming ideas akin to collaborative problem solving.

This is NOT an invitation to harass your TA/tutors to pose for selfies with you.

Collaboration Policy and Academic Honesty

Collaboration on homework sets (with others in addition to your partner(s)) is not only permitted, but also encouraged. To maximize learning, we suggest you first try solving the problems on your own, before exchanging and brainstorming ideas with your classmates. You must however write

out all solutions only with your homework partner(s). On each homework, please state who you discussed the problems with.

You are also allowed to consult other sources, for example resources on the Internet, for alternative explanations of concepts and results covered in the course, as well as related reading materials. Inevitably, some of you will (intentionally or unintentionally) stumble upon solutions to homework problems. In order to have an enforceable collaboration policy, the bottom line is that you must write out all the solutions in your own words, demonstrating that you at least understand the solution you have written. Under this policy, whilst you *are* allowed to just search for problem solutions online (if the problem is standard enough), and rewrite the solutions in your own words, it is of course heavily discouraged for the sake of *your* education.

If you do happen to base your answer (even if only partially) on outside sources, please also cite them. This is for the instructor's (and the class') benefit, to see what online sources may be useful.

For the purposes of this collaboration policy, treat Large Language Model (LLM) generated outputs as the same as other online sources. To cite an LLM output for its content, give the model name/version as well as the prompt used. If you used LLM to polish your writing (although also highly discouraged because LLM technical writing is generally terrible), you should cite only the model name/version.

Failure to comply with this lax collaboration policy, that is if you submit something that you plagiarized without demonstrating any understanding, results in a bad course grade and/or a report to OSSJA regardless of performance in the rest of the course.

In case of any doubt from the course staff on a student's understanding of their homework submissions, the instructor reserves the right to ask them for a meeting to determine if the collaboration policy has been violated. The scope of the meeting (e.g. which homework(s) the meeting covers) will always be given with ample notice prior to the meeting.

HW0 includes a task to sign an acknowledgement of this Collaboration Policy on a Google Form. You will only get credit for homework submitted on or after your signing date of the Collaboration Policy.

Late Policy

Except with the prior approval of the instructor, any submission that is late for no more than 48 hours will get a 20% reduction on that homework's grades, and any submission more than 48 hours late will not be graded.

Permission for unpenalized late submissions must be requested **at least 24 hours in advance** (except for emergency situations outside of the student's control), and will only be granted in exceptional/extenuating circumstances or for religious observances. Contact the staff mailing list for such requests.

Midterm and Final Exam

There is one midterm happening in class on 13 November 2025. It will be a 1-hour test, covering Divide and Conquer, Dynamic Programming and Greedy Algorithms.

The final exam is on 10 December 2025 from 20:30 to 22:30, at Storer Hall 1322 (our usual lecture hall). Please note that this late evening time is set by the university, and the instructor has absolutely no control over it. The final exam covers all concepts introduced in class, except those explicitly announced as non-examinable by the instructor.

Accommodations (SDC and Religious Observance)

If you have any disabilities (of any form and type), or any existing or new medical conditions that could affect your learning and ability to complete the coursework, please contact SDC or a dean to discuss. Let us know as soon as possible if you require any accommodations, attaching a relevant Dean's note or SDC email. The staff will support you as best as we can.

Students with religious observance conflicting with the course schedule should endeavour to inform the instructor within the first 2 weeks of the quarter, and no later than 1 week in advance, in order for us to make suitable arrangements.

Mental Health

Being a student can be very stressful. If you feel you are under too much pressure or there are psychological issues that are keeping you from performing well at UC Davis, we encourage you to contact Student Health and Counseling Services. They provide confidential counselling.

`https://shcs.ucdavis.edu/`

Coping with Unforeseen or Difficult Circumstances

If there are unforeseen or difficult circumstances affecting your ability to do well in the course, we are happy to take them into account with respect to our late policy. Please feel free to talk to the instructor about this. Additionally, the Office of Student Support and Judicial Affairs (`https://ossja.ucdavis.edu/`) can be a helpful resource for discussing current concerns and academic and personal plans.

Contacting the Staff

For any sensitive matters, you should feel free to contact the instructor or departmental staff directly, ignoring the instructions below.

- For technical questions about course content: post on Piazza, or better yet, go to collab hours
- For SDC accommodations: ask SDC to contact the instructor
- For requests for deadline extensions: email the course staff mailing list
- For grading concerns: email the course staff mailing list

- For concerns about the final grade: email the instructor
- For diversity and related concerns: email the instructor, or Professors Dipak Ghosal, Kurt Eiselt