

EECS 376: Foundations of Computer Science Syllabus

The University of Michigan, Spring 2024

An introduction to Computer Science theory, with applications. Design and analysis of algorithms, including paradigms such as divide-and-conquer and dynamic programming. Fundamentals of computability and complexity — learn to identify what problems a computer cannot solve at all, what problems are unlikely to be efficiently solvable, and how to apply approximations to such problems. Introduction to randomness in computation, including how algorithms can benefit from randomness and how to analyze randomized algorithms. Applications of computational hardness to cryptography, including specific algorithms that are essential to the internet.

Instructors

Ali Movaghar <movaghar@umich.edu>

Mode of Instruction

EECS 376 lectures and discussions sections will be in person.

All dates and times in this course are with respect to Ann Arbor (Eastern) time.

Lectures are offered in person, and lecture slides will be posted the same day as the lecture. We do not take attendance. Lectures will be recorded and made available shortly after delivery.

Discussions will be held in person on Mondays and Wednesdays. You may attend any discussion section, subject to space availability. We do not take attendance in discussion.

Homework assignments should be turned in electronically to Gradescope.

Exams will be held in person. Refer to the [course schedule](#) for exam dates and times.

Office hours will be held primarily in-person. Some office hours will be available on Zoom.

Computer and Network Recommendations

Make sure you have a laptop consistent with [CAEN recommendations](#).

Test your internet connection with the [U-M Custom Speedtest website](#) and make sure it meets the [minimum requirements for any UM service](#). You'll need more bandwidth if there will be multiple simultaneous users in your household.

Resources for help with computing equipment:

- Information and Technology Services (ITS) [Laptop loaner program](#)
- College of Engineering (CoE) [Office of Student Affairs](#), email requests to coe-studentaffairs@umich.edu

Overview

For every complex problem there is a solution that is clear, simple and wrong. — H. L. Mencken

Welcome to EECS 376: Foundations of Computer Science! This course covers foundational aspects of Computer Science that will help you reason about any computing task. In particular, we are concerned with the following:

- What are common, effective approaches for designing algorithms?
- How do we reason about whether an algorithm is correct, and how efficient it is?
- Are there limits to what problems we can solve with computers, and how do we identify whether a particular problem is solvable?
- What problems are efficiently solvable, and how do we determine whether a particular problem is?
- What techniques can we use to approximate solutions to problems that we don't know how to solve efficiently?
- Can randomness help us in solving problems?
- How can we exploit difficult problems to build algorithms for cryptography and security?

The only way to answer these questions is to construct a *model* and apply a *proof-based* (i.e. *mathematical*) methodology. Thus, this course will feel much like a math course, but we apply the techniques to widely applicable problems in Computer Science.

As an example, how can we demonstrate that there is no general algorithm for determining whether or not two programs have the same functionality? A simple but incorrect approach would be to try all possible algorithms to see if they work. However, there are infinitely many possible algorithms, so we have no hope of this approach working. Instead, we need to construct a model that captures the notion of what is computable by any algorithm and use that to demonstrate that no such algorithm exists.

The main purpose of this course is to give you the tools to approach computational problems you've never seen before. Rather than being given a particular algorithm or data structure to implement, approaching a new problem requires reasoning about whether the problem is solvable, how to relate it to problems that you've seen before, what algorithmic techniques are applicable, whether the algorithm you come up with is correct, and how efficient the resulting algorithm is. These are all steps that must be taken prior to actually writing code to implement a solution, and these steps are independent of your choice of programming language or framework. If we required you to implement

all the algorithms you design in this course, the workload would be far greater, and it would only replicate the coding practice you get in your programming courses. Instead, we focus on the aspects of problem solving that you have not had much experience in thus far.

The training we give you in this course will make you a better programmer, as algorithmic design is crucial to effective programming. This course also provides a solid framework for further exploration of theoretical Computer Science should you wish to pursue that path. However, you will find the material here useful regardless of which subfields of Computer Science you decide to study.

Please refer to the [course schedule](#) for a full list of topics that we plan to explore.

Prerequisites

Passing EECS 203 and EECS 280 are all you need to be prepared for this course. Everyone who has made it through EECS 203 and 280 is able to complete EECS 376 successfully.

Textbook

We use a set of [course notes](#) developed specifically for this class. Required reading assignments are listed on the [schedule](#). We encourage you to read them in advance of lecture, but it's fine if you read them afterwards instead.

As additional resources, we recommend *The Nature of Computation* by Cris Moore and Stephan Mertens, and *Introduction to the Theory of Computation* by Michael Sipser (2nd or 3rd edition). The former textbook is [available from the university library](#).

Communication

The [course website](#) links to all course resources. Please check the website regularly for updates.

We use [Piazza](#) as a collaborative discussion forum for the course, and it is the best place for technical questions, assignment updates, and policy questions. Please do not publicly post homework solutions before their due dates; if you have a specific question about your solution, make a private post instead. Please **search the forum before posting**; duplicate questions create extra work for everyone, and they make it harder for others to find answers to their questions.

Please use the [admin request form](#) for administrative requests, including registration issues, extensions, and accommodations.

For confidential matters, please contact us using our [individual email addresses](#).

We sometimes publish important announcements via [Canvas](#). Please ensure that you can receive Canvas announcements.

Exams

We will have one midterm exam and one final exam. Exam dates are on the [course schedule](#). Make sure to verify that you can attend both exams before registering.

Alternate Exams

We provide alternate exams only for documented conflicts due to required attendance for other courses, a religious holiday, or university-affiliated athletics.

We will also work with you if you have [SSD accommodations](#).

We will post forms for requesting alternate exams and letting us know about SSD accommodations in the first few weeks of the semester. To ensure that we can accommodate requests, we will announce deadlines for submitting the forms.

Discussions

In addition to lectures, we will hold discussion sections each week. While the main goal of lectures is to introduce new concepts, discussion sections are intended to give you deeper insight into the concepts as well as some practice with exercises. Though attendance is **not** required, you do need to be familiar with the material covered in discussion.

Homework Assignments

Homework assignments enable you to further your understanding of the material and to put the concepts into practice. We will have five weekly assignments. Assignments are due at 8pm Eastern Time on Thursdays, unless stated otherwise. After this, we will accept late submissions until 9:59pm on the due date **with a 5% penalty**. We post solutions shortly after this as a form of immediate feedback, allowing you to learn from them and use what you learned on the next assignment. As such, **we will not accept submissions posted after 9:59pm**.

We understand that you may be unable to finish a homework assignment in time due to medical procedures, unplanned sickness, or other expected events. For this reason, your **single lowest homework score** will be dropped and will not influence your final grade. If you are unable to submit work on time for an extended period, contact the [course administration](#) as soon as possible.

Assignment Grading

To obtain full credit on a problem, your response must be *clear*, *concise*, and *correct*. It is not sufficient that a correct argument appear *somewhere* in the answer, if it is also accompanied by incorrect or unclear reasoning.

Assignments consist of several questions intended to give you practice on applying the concepts covered in lecture, as well as to build and evaluate your understanding of the material.

Collaboration

Outside of exams, we encourage collaboration in EECS 376, especially on concepts, strategies, and solution ideas. We encourage you to form study groups to support each other and work together according to the rules outlined below.

Unless otherwise specified in an assignment or this document, all work you submit must be **your own, original work**. Collaboration must not result in solutions that are identifiably similar to other solutions, past or present. If you are referencing others' work, put it in quotes! If you are directly quoting, or building on others' writing, provide a citation. See the [Rackham Graduate policy on Academic and Professional Integrity](#) for the definition of plagiarism and associated consequences. Violations of the Honor Code will be taken seriously; please see [details about the Honor Code here](#).

The core rules for collaboration in this course are:

- **Your solution write-up must be done individually and entirely on your own, and you may not look at anyone else's write-up, including your collaborators'.**
- When turning in work that benefited from a collaboration, **you must acknowledge your collaborator(s)** (as your answer to question 0 on the assignment).

Further details about what constitutes acceptable collaboration are below:

Encouraged Collaboration	Unacceptable Collaboration
Brainstorming solution ideas, e.g., what kind of reduction to use for a problem and what a suitable language for the reduction might be	Sharing solution sketches or drafts, or otherwise directly giving away your solution to someone, in writing or verbally
Helping others understand the problem statement and nuances of the problem	Providing your solution as a reference
Sharing hints provided by the course staff	Copying solutions in whole or in part (including from the Internet, an AI bot, etc.), even if the solution is modified Writing a solution for someone else, or having someone else write your solution

Encouraged Collaboration	Unacceptable Collaboration
Looking at someone else's solution after the deadline (plus any grace period) to learn from it and/or provide feedback	Sharing your solution in a way that could be copied, e.g., sending answers over email or taking a picture of a solution

If you are unsure about what constitutes an Honor Code violation, please contact the course staff with questions.

Grades [↗](#)

Your final grade is computed from a weighted average of homework assignment grades, exam grades, and returning a receipt that you submitted course evaluations.

Assignment	Weight
Homework assignments	40%
Midterm exam	29%
Final exam	30-31%
Course evaluations	0-1%
Total	100%

We encourage you to complete the course evaluations, as they help us improve the course and CSE as a whole. For each evaluation, you will receive full credit if you fill it out and submit it before the deadline. Otherwise, your final exam score will count as your score for that survey. For example, if you submit no course evaluation receipt, your final exam will count for 31% of your final grade, rather than 30%.

There are no letter grades for individual assignments or exams. The final course letter grade is based on the total weighted score earned. Passing EECS 376 requires meeting the following criteria:

- Your weighted-average exam score must be at least 45% of the available exam points, weighted as above.
- Your weighted total for the course as a whole must be a passing score.

Your letter grade will be determined by your total weighted score. The exact thresholds between letter grades will be set at the end of the semester. We may adjust any of the thresholds, but only in your favor.

Total Weighted Score \geq	Letter Grade \geq
40%	D
48%	C-
55%	C
64%	C+
70%	B-
76%	B
82%	B+
88%	A-
93%	A

In particular, if you score at least 55% overall, and your weighted exam average is at least 45%, then you are guaranteed to pass the course with a C or better.

There is no guaranteed threshold for an A+. A grade of A+ is awarded only for exceptional work and participation (e.g., providing many high-quality answers on Piazza, doing many extra-credit questions, etc.), at the discretion of the instructors.

Please note:

- We do not round scores when computing grades.
- We do not offer the opportunity for “make-up” work.
- We do not reconsider final grades after they have been released. We can only submit grade changes to correct errors in recording or calculating scores.

Regrades [↗](#)

Some components of assignments and exams are graded by hand. Request a regrade via the mechanism announced for the assignment; you must **explain why you believe the rubric was applied incorrectly, and say what score you believe you should receive according to the rubric**. We accept requests only to correct grading errors, not disagreement with the rubric. Your score on the regraded problem may increase or decrease.

We do not accept regrade requests for automatically graded components of assignments or exams.

In all cases, regrade requests are due no later than 7 days after a grade is released unless a shorter deadline is specified.

Accommodations for Students with Disabilities

If you need, or think you might need, an accommodation for a disability, please let us know during the first three weeks of the semester. Some aspects of this course may be modified to facilitate your participation and progress. As soon as you make us aware of your needs, we can work with the Services for Students with Disabilities (SSD) office to help us determine appropriate academic accommodations. SSD (734-763-3000; <http://ssd.umich.edu>) typically recommends accommodations through a Verified Individualized Services and Accommodations (VISA) form. Any information you provide is private and confidential and will be treated as such.

Commitment to Equal Opportunity

As indicated in the [General Standards of Conduct for Engineering Students](#), we are committed to a policy of equal opportunity for all persons and do not discriminate on the basis of race, color, national origin, age, marital status, sex, sexual orientation, gender identity, gender expression, disability, religion, height, weight, or veteran status. Please feel free to contact us with any problem, concern, or suggestion.

Student Well-Being

Students may experience stressors that can impact both their academic experience and their personal well-being. These may include academic pressure and challenges associated with relationships, mental health, alcohol or other drugs, identities, finances, etc.

If you are experiencing concerns, seeking help is a courageous thing to do for yourself and those who care about you. If the source of your stressors is academic, please contact the courses instructors so that we can find solutions together. For personal concerns, U-M offers many resources, some of which are listed at [Resources for Student Well-being](#) on the Well-being for U-M Students website. You can also search for additional resources on that website.

Revisions

While we try to do our best to plan ahead, unfortunately, sometimes circumstances do arise that necessitate a policy change. When this happens, the change will be announced, and this document will be updated with the new policy.
