CS4010: Introduction to Theoretical Computer Science

Fall 2026

Course Syllabus

1 Course Information

1.1 General Information

Instructor: Aditi Dudeja – aditidudeja@cuhk.edu.cn – `https://aditidudeja.github.io`

Prerequisites: Mathematical maturity and CSC3001 (or equivalent).

Co-requisites: CSC4120 (or equivalent).

We assume a knowledge of basic mathematics, e.g., elementary calculus, proof by induction, linearity of expectation, permutations, logarithms, and asymptotic notation (i.e. "big-Oh", "big-Omega").

Textbook: Introduction to the Theory of Computing, 3rd Edition by Michael Sipser. ISBN-13: 9780357670583. The book is on reserve in the CUHK-Shenzhen Library.

Homeworks and exams will be solely based on the materials presented in the class – this includes the parts that may not necessarily be directly covered in the textbook.

Website: `https://aditidudeja.github.io/courses/csc4010-f26/` The webpage will contain updated syllabus information as the semester progresses and a calendar. You can also find course announcements, and slides there. Course announcements will be made via **Blackboard** and email. Grades can be checked on **Blackboard**.

Email policy: Please start the subject of all your emails to course staff with "[CSC4010]" so that they are not lost in our inboxes.

1.2 Lectures and Recitations

Class	Time	Place
Lectures	Monday, Wednesday 8:30-9:50	Administration Building E206
Tutorial	Tuesday 19:00-19:50	Teaching D Room 307

Participation in any of the lectures or recitations is **not mandatory**. However, if you decide to participate in the class, I ask you to please be respectful of your fellow students and do not use any electronic devices (cellphones, laptops, tablets) during the lectures, unless you are using a laptop or tablet for note-taking.

1.3 Course Staff

Instructor: Aditi Dudeja aditidudeja@cuhk.edu.cn

TA: ZHU Tianran tianranzhu@link.cuhk.edu.cn

USTF: SUN Youran youransun@link.cuhk.edu.cn

1.4 Office Hours

The Instructor's office hours are:

Time	**Place**
Mon, Tues: 4pm - 5pm	DY 504b
Wed, Thurs: 4pm - 5pm	on Zoom

If you have something to discuss directly with me (e.g., a concern, a question, or a comment), send an email with the subject "[CSC4010] ?" (replace ? with what you like to discuss) to me and we will set up a time for a meeting.

The TA's office hours are:

Time	**Place**
Mon: 4pm - 5pm	ZX 408-49
Wed: 4pm - 5pm	on Zoom

The USTF's office hours are:

Time	**Place**
Mon: 1pm - 2pm	on Zoom
Thurs: 1pm - 2pm	on Zoom

1.5 Course Topics

The question of which tasks can be performed efficiently is central to human experience, since time and other resources are always in shortage. The field of theoretical computer science studies the effects of limited resources: How does restricting the time, memory, and other resources available to algorithms affect their ability to perform tasks? This question is formulated and studied as generally as possible and with mathematical rigor, using well-defined abstract notions of algorithms, tasks, and resources.

This course teaches how to mathematically model general computation and how to analyse models. The main conceptual focus of the course is impossibility results (i.e. lower bounds), which rigorously prove limits for models of computation. A secondary focus is utilizing impossibility results for productive purposes, i.e. building algorithms and protocols from lower bounds. The key points students can take from the course are:

1. We can abstract computation into mathematical models, and prove impossibility results for these models.

2. This abstraction is relevant to real-world scenarios.

3. Impossibility results can be leveraged to build useful algorithms.

The course will have four (planned) modules.

(a) **Computability Theory:** In this module, we will learn how to mathematically model computational problems and models. We will study Turing machines, which are arguably, the right model to capture computation. We will ask which problems are computable by a Turing Machine given an arbitrarily large (but finite) number of resources. We will show certain problems are uncomputable. An example of such a problem is the Halting Problem.

(b) **Complexity Theory:** The distinction between "computable" and "uncomputable" problems is crude, and provides only limited information as to which problems can actually be computed, because it does not take into account resource limitations. Specifically, even computable problems may be beyond reach of any real-world computational procedure, whose resources are inherently limited. In this model, we will obtain more information about realistically solvable problems by analyzing resource-bounded models. The resources we will focus on are time and memory. We will show (among other things) that given more time, a Turing machine can solve more problems.

(c) **Fine-grained Complexity:** This area of research attempts to discover the precise time complexity of solving specific interesting problems. For example, given a set of vectors, can we find a pair of orthogonal vectors faster than brute-force?

(d) **Cryptographic Applications of Lower Bounds:** In this module, we will learn that the lower bounds we have proved so far are useful in practice and can be leveraged to build algorithms and cryptographic protocols for safe computation.

2 Assignments and Grading

2.1 Grading

- 40% In-class Tests
- 30% Midterm exam
- 30% Final exam

2.2 In-Class Test

There will be 4 during the semester. Each test will contain several multiple-choice (or similar types of) questions on the topics directly covered in the lectures. You should think of these tests as a reminder to follow the course lectures; they are all at the level that if you even briefly review the topics taught in that week, you should be able to answer the quiz in less than 10 minutes. The tests are also a way for me to monitor the class progress and engagement as a whole.

- **Timing:** The tests will be held in-class during the first twenty minutes of tutorial hours on (tentatively) 29/9/26, 27/10/26, 24/11/26, 15/12/26.
- A missed/late quiz draws zero credit. Exceptions will be made only for exceptional circumstances and you will need to provide a university-issued written verification. However, we will be dropping the grade of your lowest-graded quiz.

2.3 Letter Grades

The tentative letter grade thresholds are as follows.

1. 40% is the minimum numeric grade that translates to D.
2. 45% is the minimum numeric grade that translates to D+.
3. 50% is the minimum numeric grade that translates to C-.
4. 55% is the minimum numeric grade that translates to C.
5. 60% is the minimum numeric grade that translates to C+.
6. 65% is the minimum numeric grade that translates to B-.
7. 70% is the minimum numeric grade that translates to B.
8. 75% is the minimum numeric grade that translates to B+.
9. 80% is the minimum numeric grade that translates to A-.
10. 85% is the minimum numeric grade that translates to A.

At the end of the semester, after I've had a chance to review all the grades, I might make the thresholds a bit more generous. I won't make them harsher.

2.4 Homeworks

During the course you will get about 4-5 assignments. They will help you in developing and sharpening the skills you acquire during the lectures. They will involve a small number of questions for you to solve, as well as mathematical statements to prove. You should think of assignments as the main way of exercising your skills in this course beyond what is directly taught in the lectures.

These will not be graded, but we will discuss them during tutorials and **I strongly encourage you to discuss them in office hours and/or submit them for feedback**. You should make an honest attempt to try solving the questions. In the office hours and tutorials, describe your attempt and where you got stuck and we will give you further hints on how to solve the problem.

Importance of HW for grading: Even though HW is not graded it is actually much more important than that. The in-class tests will be based on the HW. Approximately 30-40% of each exam (final/midterm) will be essentially identical to HW problems; sometimes the problems will be literally identical to a HW problem, other times they will be the same except for a few tiny tweaks. If you understand how to do the HW problem, the corresponding exam problem will be easy. On the flip side, if you didn't do the HW problem, then solving these problems in the short time-frame of the exam will be very difficult. So you can basically think of 30-40% of each exam as testing that you understood the HW.

2.5 Mid-Term Exam

There will be one mid-term exam in this course. You should think of this exam as a mix between quizzes and homeworks and a way of both testing your understanding of the course as well as further exercising your thinking skills.

- The midterm will be held tentatively in-class on 2/11/2026
- Midterm will be based on lectures 1-14. Any changes to these plans will be announced well in advance.
- Mid-term exam is closed-book: no textbooks, electronic devices, or cheat sheets are allowed.
- A missed exam draws zero credit. However, emergencies will be considered upon submitting a university-issued written verification to the Instructor.

2.6 Final Exam

- The final exam will be held at the end of the semester according to the University schedule. Further details on the timing and format of the final exam will be released later in the semester.
- The final exam is closed-book: no textbooks, electronic devices, or cheat sheets are allowed.

3 Academic Integrity

The students are expected to follow CUHK-Shenzhen, SDS, and Graduate School academic integrity policies for all their work in this course. Please familiarize yourself with these policies if you have not done so yet.