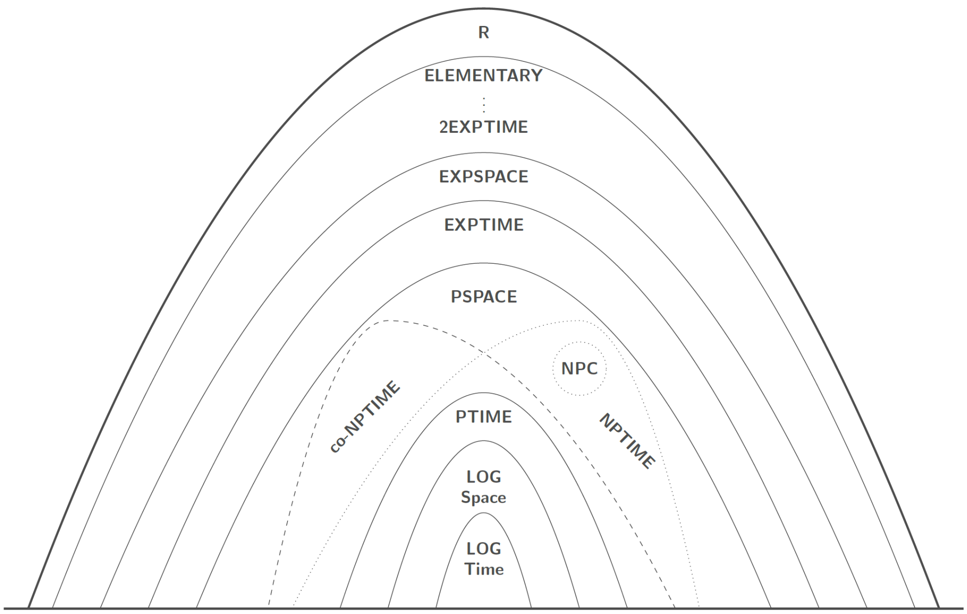
Computational Complexity Theory

CS 313 L1
Fall Semester 2024

"*P* versus *NP* — a gift to mathematics from computer science" – *Steve Smale*.



Course Information

Start Date	End Date	Class Location	Meeting Time
Sep 02 2024	Dec 03 2024	GF-C109	Mon Fri (11:30 AM-12:45 PM)
Aug 23 2024	Aug 23 2024	Fri (11:30 AM-01:25 PM)	Fri (11:30 AM-01:25 PM)

This course is a CS Elective and can be counted toward a CS minor. For other students, it can be counted as either a Free Elective, University Elective, or CS Elective.

Instructor Information

Instructor: Muhammad Abdullah Zafar
Office Location: E-112
Email: abdullah.zafar@sse.habib.edu.pk
Office Hours: Mon / Wed, 11:30 - 1:00 pm

Course Description

Some mathematical problems are harder to solve than others. Take for example the problem of finding a route on a map that visits all towns exactly once. So far, no computer can accomplish this task for a modestly sized map, even if it ran for a 100 years! Such results are the outcome of thinking computationally about mathematics; in specific, thinking about the resources expended in calculating something. This methodology has had far reaching effects: Internet banking, e-commerce, digital currencies, to name a few, all rely fundamentally on efforts of this kind.

The course will familiarize students with the concept of efficient computation, building upon the conceptual foundations laid in CS 212: Nature of Computation. It will take a Turing-machine-centric approach to studying the complexity classes that are most relevant to efficient computation. These include P, NP, PSPACE, LOG, NLOG, BPP, and a few others. We will also learn to use methodologies such as reduction, diagonalization, and relativization to study classes such as NPI and the Polynomial Hierarchy. We will also study Boolean circuits and introduce Parameterized Complexity theory and its corresponding complexity classes. If time permits, we will also introduce some applications to cryptography and randomness.

Course Aims

CS 313, Computational Complexity Theory (CCT), is about the study of feasible computation: computation that is physically realizable (as opposed to the historically earlier general theory of computation). Ideally one would like to determine the kinds of computations that can be carried out in the real world on current digital computers. Conversely, one requires knowledge of what can't be computed efficiently; a theory of intractability. This project is closely connected to the design and analysis of algorithms, as explored in CS 412, but here we are more interested in studying the hardness of problems themselves, not in the development of fast algorithmic solutions to them.

At the end of the course, you ought to have acquired the following skills:
• The ability to articulate the fundamental concepts in Computability and Complexity Theory.
• The ability to analyze computational and complexity properties of combinatorial problems of various levels of difficulty.
• The ability to guide the development of real world algorithms by relevant results from Complexity Theory.

Course Learning Outcomes (CLOs)

By the end of the course, students will be able to:

CLO	Description	Learning-Domain Level
1. Computation	Compare different models of computation such as Turing Machines, its oracular variants, and Boolean Circuits.	Cog-2
2. Complexity	Analyze and be able to develop solutions based on relevant results from Complexity Theory.	Cog-6
3. Proof	Write relatively formal proofs of the results of Complexity Theory covered in class.	Cog-6
4. Collaboration	Collaborate fruitfully on researching and presenting homework problems.	Aff-3

Format and Procedures

We will use CANVAS for all official course information.

Expectations

So that you succeed in this course, we expect that you follow the good academic practices listed below.

- You will *read the book* and stay abreast of it as the course proceeds.
- You will be *fully present* in the class whenever you choose to do so and you will help maintain the class decorum.
- You will *observe the 2-week contest period*. Bring up any queries about your score with the course staff within 2 weeks of its release. Otherwise, we will have moved on to the next assessment and will not be able to help you.

Seniors

If you are taking this course as a senior, chances are that you might need to pass it in order to graduate on time. Please be alert regarding your standing in the course and utilize the course staff's consultation hours *early on* if you start to lag behind the class, and not when it is too late!

Time Commitment

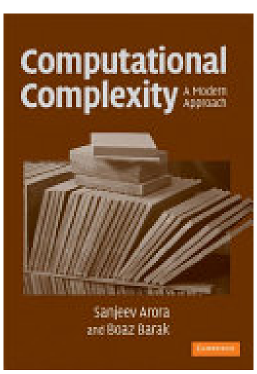
This is a 3+0 course, i.e. 3 lecture credit hours. Each credit hour formally equates to a certain number of hours (at least 2) that you are expected to work outside class. The exact number varies, and you may need more or less than others based on your comfort with mathematics and capacity to absorb and apply abstract ideas.

Mode of Instruction

To every extent possible, this course will take place in person.

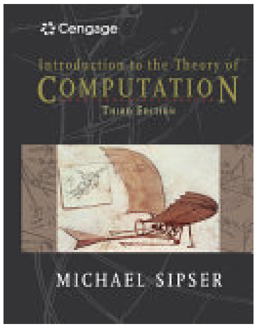
In the unfortunate circumstance where we need to go online, relevant instructions will be shared accordingly.

Required Texts and Materials



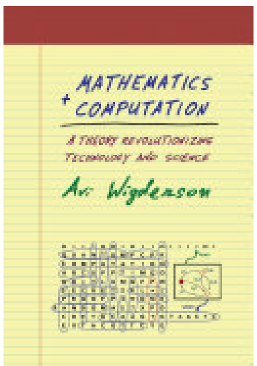
Computational Complexity - A Modern Approach
ISBN: 9780521424264
Authors: Sanjeev Arora, Boaz Barak
Publisher: Cambridge University Press
Publication Date: 2009-04-20

Introduction to the Theory of Computation

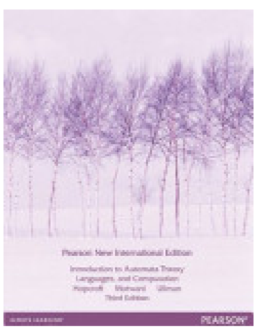


ISBN: 9781133187790
Edition: 3rd
Authors: Michael Sipser
Publisher: Cengage Learning
Publication Date: 2012-06-27

Optional Materials



Mathematics and Computation
ISBN: 9780691189130
Authors: Avi Wigderson
Publisher: Princeton University Press
Publication Date: 2019-10-29



Introduction to Automata Theory, Languages, and Computation
ISBN: 9781292039053
Authors: John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman
Publication Date: 2014-01-01

Assessments

Assessment type	%	Remarks
Homework Assignments (3)	20	Assignments will be discussed in pairs, written individually and graded through pair-wise viva as well as individual checking.
Quizzes (5)	15	Quizzes will be attempted individually in-class.
Mid 1	15	This is an individual written exam.
Mid 2	15	This is an individual written exam.
Final	25	This is an individual written exam.

Assessment type	%	Remarks
Class Part	10	To be assigned by the instructor at their own discretion.

The homework grade will be divided evenly between the individual written submission and pair-wise viva.

Grading Scale

Letter Grade	GPA Points	Percentage
A+	4.00	[95-100]
A	4.00	[90-95)
A-	3.67	[85-90)
B+	3.33	[80-85)
B	3.00	[75-80)
B-	2.67	[70-75)
C+	2.33	[67-70)
C	2.00	[63-67)
C-	1.67	[60-63)
F	0.00	[0, 60)

Note: [a, b) is a range of numbers from a to b where a is included in the range and b is not.

Late Submission Policy

Please observe the deadline requirements prescribed for each assessment.

Homeworks will incur a 10% penalty for each late day, i.e., a submission that is one day late will be graded out of 90%, a submission that is two days late out of 80% and so on.

Quizzes and exams will not be accepted late.

Please be mindful of deadlines and discuss with your instructor *beforehand* if you foresee any issues.

Week-Wise Schedule (Tentative)

The schedule may change in view of class progress as the semester proceeds. See the Live Syllabus for an updated version.

Week	Topic	Notes
1	**Preliminaries** Sets, strings, and languages Logic and proofs Asymptotic notations	
2	**Basic Models of Computation** Finite automata and Turing machines Efficiency and running time Computable vs uncomputable	CP (1/7)
3	**Basic Complexity Classes** The class P The class NP	Quiz 1
4	**Reducibility and NP-completeness** Polynomial time reductions NP-completeness The Cook-Levin theorem	HW 1 due CP (2/7)
5	**More on NP-completeness** Examples of NP-complete problems Proving problems NP-complete Optimization vs decision problems NP-hard problems	Quiz 2
6	**Exponential Time Classes** The classes EXP & NEXP **Diagonalization** Time hierarchy theorem for Deterministic TMs	Midterm 1 CP (3/7)
7	**Space Complexity** The class PSPACE The class NL	
8	**Space Complexity II** NL-completeness	HW 2 due Quiz 3 CP (4/7)
9	**The Polynomial Hierarchy** The class {Sigma}p2 The polynomial hierarchy Alternating Turing machines	

Week	Topic	Notes
10	**Boolean Circuits** Boolean algebra (and logic) Boolean circuits and P/poly	Quiz 4 CP (5/7)
11	**Boolean Circuits II** Boolean algebra (and logic) Boolean circuits and P/poly	Midterm 2
12	**Randomized Computation I** Probabilistic Turing machines Probabilistic complexity classes (RP, coRP, ZPP)	HW 3 due CP (6/7)
13	**Randomized Computation** Probabilistic Turing machines Probabilistic complexity classes (RP, coRP, ZPP)	Quiz 5
14	**Advanced Topics** Parameterized Complexity	CP (7/7)
15	**Advanced Topics** Relativization of P vs NP	
16	**Final Exam Week**	Final

Academic Integrity

Each student in this course is expected to abide by the Habib University Student Honor Code of Academic Integrity. Any work submitted by a student in this course for academic credit will be the student's own work.

Scholastic dishonesty shall be considered a serious violation of these rules and regulations and is subject to strict disciplinary action as prescribed by Habib University regulations and policies. Scholastic dishonesty includes, but is not limited to, cheating on exams, plagiarism on assignments, and collusion.

a. Plagiarism: Plagiarism is the act of taking the work created by another person or entity and presenting it as one’s own for the purpose of personal gain or of obtaining academic credit. As per University policy, plagiarism includes the submission of or incorporation of the work of others without acknowledging its provenance or giving due credit according to established academic practices. This includes the submission of material that has been appropriated,// bought, received as a gift, downloaded, or obtained by any other means. Students must not, unless they have been granted permission from all faculty members concerned, submit the same assignment or project for academic credit for different courses.

b. Cheating: The term cheating shall refer to the use of or obtaining of unauthorized information in order to obtain personal benefit or academic credit.
c. Collusion: Collusion is the act of providing unauthorized assistance to one or more person or of not taking the appropriate precautions against doing so.

All violations of academic integrity will also be immediately reported to the Student Conduct Office.

You are encouraged to study together and to discuss information and concepts covered in lecture and the sections with other students. You can give "consulting" help to or receive "consulting" help from such students. However, this permissible cooperation should never involve one student having possession of a copy of all or part of work done by someone else, in the form of an e-mail, an e-mail attachment file, a diskette, or a hard copy.

Should copying occur, the student who copied work from another student and the student who gave material to be copied will both be in violation of the Student Code of Conduct.

If you wish to use generative-AI tools to complete any of your assessments, you must first obtain permission from your course instructor. AI generated work will not be accepted in all classes or even all assessments. The instructor's permission is required. If the permission is granted, you should declare its use and properly cite the source of the generated content. Failing to identify AI written or assisted work is academic dishonesty and will be treated as any case of plagiarism by the university.

The principle for academic integrity is that your submissions must be substantially your own work and that any work that is not originally your thought must be identified and credited. If the use of AI tools is prohibited in the course, respect the rules and do not use these tools for assessments. The fundamental purpose of assessment is to learn, synthesize information and explain new connections and interpretations that arise from your secondary research. Be aware that unauthorized use of AI tools for assessments can result in a conduct case being filed. This can have serious consequences for your academic standing and future career opportunities.

During examinations, you must do your own work. Talking or discussion is not permitted during the examinations, nor may you compare papers, copy from others, or collaborate in any way. Any collaborative behavior during the examinations will result in failure of the exam, and may lead to failure of the course and University disciplinary action.

Penalty for violation of this Code can also be extended to include failure of the course and University disciplinary action.

Program Learning Outcomes (For Administrative Review)

Upon graduation, students will have the following abilities:

- PLO 1: Theoretical Computer Science: recall and apply foundational principles of computer science.
- PLO 2: : Application Development: build software systems of varying complexity in light of fundamental computer science principles and any other constraints.
- PLO 3: Analysis and Design: perform technical analysis and design using core computing and mathematical knowledge.
- PLO 4: Systems: apply the knowledge of computing systems.
- PLO 5: : Research and Exploration: develop expertise in and contribute to a given sub-field of computing by drawing upon a strong foundation in the fundamentals of computer science and mathematics to solve real-life problems.
- PLO 6: Problem Solving: identify and analyze problems and propose effective computing-based solutions.
- PLO 7: Practical Exposure: make effective use of current tools, technologies, and good industry practices.
- PLO 8: Responsible Citizenship: conduct their computing practice in a manner that is ethical and socially responsible and corresponds to their distinct sense of identity and service to the community.
- PLO 9: Self-Learning: continuously adapt their skills to the changes taking place around them.
- PLO 10: Design Thinking: apply design thinking principles to the design of a solution.
- PLO 11: Multi-disciplinarity: incorporate knowledge and input from multiple disciplines.
- PLO 12: Communication and Teamwork: communicate and function effectively as a member or a leader of a variety of teams.

Program Learning Outcomes (PLOs) mapped to Course Learning Outcomes (CLOs)	
	CLOs of the course are designed to cater following PLOs:

	PLO 1: Analysis PLO 2: Design PLO 6: Self-learning PLO 8: Communication & Teamwork			
	Distribution of CLO weightages for each PLO			
	CLO 1	**CLO 2**	**CLO 3**	**CLO 4**
PLO 1	20%	30%	30%	20%
PLO 2	10%	30%	30%	30%
PLO 6		30%	50%	20%
PLO 8				100%

Mapping of Assessments to CLOs

Assignments	**CLO #01**	**CLO #02**	**CLO #03**	**CLO #04**
HW 1				
HW 2				
HW 3				
Quiz 1				
Quiz 2				
Quiz 3				
Quiz 4				
Quiz 5				
Midterm 1				
Midterm 2				
Final				

Recording Policy

Only asynchronous and synchronous online sessions will be conducted and recorded via MS Teams. Link to the recordings will be available to all students on Canvas Learning Management System.

For this course, collaboration is allowed in the following instances: homework assignments.

Accommodations for Students with Disabilities

In compliance with the Habib University policy and equal access laws, I am available to discuss appropriate academic accommodations that may be required for student with disabilities. Requests for academic accommodations are to be made during the first two weeks of the semester, except for unusual circumstances, so arrangements can be made. Students are encouraged to register with the Office of Academic Performance to verify their eligibility for appropriate accommodations.

Inclusivity Statement

We understand that our members represent a rich variety of backgrounds and perspectives. Habib University is committed to providing an atmosphere for learning that respects diversity. While working together to build this community we ask all members to:

- share their unique experiences, values and beliefs
- be open to the views of others
- honor the uniqueness of their colleagues
- appreciate the opportunity that we have to learn from each other in this community
- value each other's opinions and communicate in a respectful manner
- keep confidential discussions that the community has of a personal (or professional) nature
- use this opportunity together to discuss ways in which we can create an inclusive environment in this course and across the Habib community

Attendance Policy

You are expected to attend and participate in all lectures Under extenuating circumstances, you may miss up to 05 synchronous sessions. Excessive absences will lead to an automatic withdrawal from the course.

Office Hours Policy

Every student enrolled in this course must meet individually with the course instructor during course office hours at least once during the semester. The first meeting should happen within the first five weeks of the semester but must occur before midterms. Any student who does not meet with the instructor may face a grade reduction or other penalties at the discretion of the instructor and will have an academic hold placed by the Registrar's Office.