UCONN
UNIVERSITY OF CONNECTICUT

CSE 3500
Algorithms and Complexity, Fall 2023
Computer Science and Engineering

Course and Instructor Information

Course Title: Algorithms and Complexity
Credits: 3
Prerequisites: CSE 2050 or 2100; and 2500

Professor: Lina Kloub
Email: lina.kloub@uconn.edu
Office location: ITE 302
Class times: Tuesday and Thursday at 12:30pm-1:45pm in MCHU 205
TAs, Office Hours and Availability: TBD by second week of class. The information will be posted in HuskyCT.

The course website is hosted on HuskyCT which is used for course material distribution, announcements, and out-of-class discussions. Changes to the course syllabus, schedule, or homeworks will be announced immediately in HuskyCT and in class.

Course Description

Algorithms and Complexity is a course focused on the design and analysis of efficient computer algorithms. The course covers various algorithm design techniques, including divide-and-conquer, dynamic programming, and greedy approaches. Additionally, it delves into graph algorithms, advanced data structures, and topics such as Asymptotic analysis and notation, reductions, and NP-completeness.

This course was created with the aim of acknowledging and embracing the unique thinking and learning styles of neurodiverse students. We strongly believe in fostering an inclusive and empowering learning environment that focuses on individual strengths. Throughout the course, we will introduce various innovative teaching methods, such as collaborative learning with peers, engaging project-based activities, and interactive games during review sessions. These approaches aim to enhance your understanding, encourage active participation, and provide opportunities to tackle complex problems effectively.

Course Objectives

The course aims to provide a comprehensive treatment of algorithm design and analysis. By the end of the course, you will be able to:

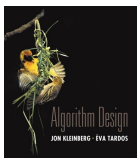
1. Demonstrate the basic concepts of algorithms.
2. Analyze and evaluate algorithm efficiency using asymptotic notations and understanding common running time complexities. This will enable you to assess and compare the efficiency of different algorithms.
3. Gain a comprehensive understanding of heaps, heapsort, and priority queues, allowing you to effectively utilize these structures in algorithm design and analysis.
4. Explore graph algorithms: You will learn key graph concepts and applications. This knowledge will empower you to solve graph-related problems efficiently.
5. Apply greedy algorithms: You will develop proficiency in applying greedy algorithms to solve problems. This will enable you to find optimal solutions for various scenarios.
6. Utilize advanced algorithmic techniques like divide-and-conquer strategies, and study the principles and applications of randomized algorithms. This will broaden your problem-solving skills and enhance your algorithmic toolkit.
7. Utilize dynamic programming techniques: You will learn and practice dynamic programming approaches through solving problems. This will enable you to effectively solve complex optimization problems.
8. Demonstrate complexity theory: You will explore the concepts of computational complexity, including classes P and NP, reduction techniques, and the concept of NP - completeness. This knowledge will provide insights into the limits of efficient algorithmic solutions.

By the end of the course, you will have acquired a comprehensive understanding of algorithm design and analysis. You will be able to apply various algorithmic techniques, analyze algorithm efficiency, and confidently tackle a wide range of computational problems.

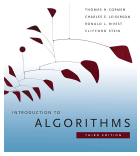
Schedule

Week	Date	Module	Book	HWs	Objective(s)
1	8/28	**Basic concepts of algorithms:** Stable matching problem, a motivating example. **Algorithm analysis:** Asymptotic notations.	Ch. 1 ,2	HW 1	1
2	9/4	**Algorithm analysis:** Asymptotic notations Common running time. **Data structures:** Understand and analyze heaps, heapsort and priority queues.	Ch. 2	HW 2	2, 3
3-4	9/11-9/18	**Graphs:** Basic definitions and applications. DFS and BFS applications, Bipartite graphs, strongly connected components, Topological Sort.	Ch. 3	HW 3	4
5	9/25	**Exam 1 (Thursday 9/28)**			
6-7	10/2-10/9	**Greedy algorithm:** Interval scheduling. Shortest path: Dijkstra. Minimum spanning tree: Kruskal, Prim. Huffman Code.	Ch. 4	HW 4, HW 5	5
8-9	10/16-10/23	**Divide-and-conquer:** Mergesort, Master theorem. Binary search, matrix multiplication (Strassen's algorithm), integer multiplication.	Ch. 5	HW 6	6
		Randomized algorithms: Quick sort and Quick select.	Ch. 5	HW 7	6
10	10/30	**Exam 2 (Thursday 11/2)**			
11-12	11/6-11/13	**Dynamic programming:** Fibonacci sequence, Longest common subsequence, Maximum sum subarray (Kadane's algorithm), and Bellman Ford.	Ch. 6	HW 8	7
13	11/20	**Thanksgiving Break**			
14	11/27	**Complexity theory:** P, NP, Reduction, and NP-complete	Ch. 8	HW 9	8
15	12/4	Individual Project selection and Group Project presentations			
	12/11	**Final Exams week**			

Suggested Course Materials



Algorithm Design by Kleinberg and Tardos, 2006. This book is our primary resource for the course. However, it is not mandatory to purchase the book, as all the material will be covered in class through slides. Additionally, you can access the official slides from the authors through this link, which will provide valuable support in comprehending the materials.



Introduction to Algorithms by CLRS, 2009. This book is recommended for additional reading and further exploration of the subject matter.

Course Requirements and Grading

The grading breakdown for the course is as follows:

Course Components	Weight
Participation and Activities	10%
Exams	35%
Homework	25%
Individual project	15%
Group project	15%

Classroom Participation and Activities (10%):

We want you to actively participate in lectures for 2 main reasons:

1. Effective communication and the ability to work with others are part of the Computer Science program objectives (full list here). These are great opportunities to build those skills.

2. The single greatest predictor of student performance is class attendance. From a meta-analysis of 52 published articles and 16 unpublished dissertations:

   These relationships make class attendance a better predictor of college grades than any other known predictor of academic performance, including scores on standardized admissions tests such as the SAT, high school GPA, study habits, and study skills.

Active participation is an essential component of this course. We will utilize various interactive learning strategies to enhance our understanding of the course material. One such strategy is the Think-Write-Pair-Share method. Think-Write-Pair-Share (TWPS) is an interactive learning strategy that we will utilize during our classroom sessions. TWPS promotes active participation, critical thinking, and collaborative learning. During designated class activities, you will be given in-class worksheets containing questions related to the topic being discussed. (1) think individually about a topic or answer to a question;(2) write down your thoughts, ideas, or answers in the provided space on the worksheet. (3) pair with a partner and discuss the topic or question; and (3) share ideas with the rest of the class.

By incorporating the TWPS strategy, we aim to create an interactive and engaging classroom atmosphere where everyone's ideas are valued. It provides an opportunity for you to actively participate, collaborate with your peers, and deepen your understanding of the course material.

Additionally, regular quizzes will be conducted at the end of each lecture, serving the purpose of capturing attendance and assessing your grasp of the covered material.

This participation policy aims to encourages you to build these skills while accommodating absences due to illness or other circumstances.

- You can earn 1 participation point per lecture, totaling approximately 26 possible points throughout the semester.
- Full credit for participation is awarded at 20 points, allowing for up to 6 absences without penalty.

Note that there is no extra credit for going above and beyond 20 points.

The lectures will be live streamed in case you cannot make it in person. The links for the live-stream can be found here.

Exams (35%):

There will be two in-class exams, each worth 10% of the final grade. Additionally, there will be a cumulative final exam worth 15%.

For each exam, you are allowed to bring one sheet of **handwritten** notes (8x11 size), and you can write on both sides.

Exam 1 and 2 will be scheduled on Thursday during the exam week. To help you prepare effectively, we have planned an interactive review session on Tuesday. During this session, we will incorporate engaging games and activities to assist you in revisiting and reinforcing the course materials. It will be a fun and interactive way to enhance your understanding and boost your confidence for the exams.

Homework (25%):

There will be a total of 9 homework assignments that consists of problem sets and/or programming assignment that designed to assess your mastery of each subject.

- We encourage high-level discussion of ideas and problems with peers, but each student must complete and submit their homework individually.
- homeworks should be handed into HuskyCT. We strongly encourage LATEX for typesetting work. www.overleaf.com is a great online LATEX editor that can be used for problem sets or programming assignment collaboration. See the online introduction to LATEX to get started.
- Everyone can have bad days, so, the lowest homework assignment score will be dropped.

Everyone will be allotted 5 no-questions-asked late day token to be used at any point unless explicitly prohibited.

- Each late day token allows you to submit an assignment up to 24 hours late. You may use 1 late day token for up to 24 hours late submission or 2 tokens for up to 48 hours late submission. Maximum of 2 tokens can be used per homework assignment.
- To use a late token, **just submit your assignment after the deadline**. We will manually deduct any late tokens used after the 48-hour late window closes for an assignment.
- You can see how many late day tokens you have left in HuskyCT under "My Grades".
- No late submission will be accepted. Homework submissions without an adequate number of late day tokens will not be accepted if they are submitted after the deadline.

Homework is typically due at 11:59 PM EST Wednesday.

Individual Project (15%):

The individual project involves selecting an algorithm not covered in class and recording a 5-8 minute video presentation.

- In the video, you will explain the selected algorithm and analyze its running time.
- You can use slides, a whiteboard, illustrations, or any creative approach to effectively present the algorithm.
- At the beginning of the video, state your name and netID.

Three presentations will be selected to be shown in class, and students will have the opportunity to vote for the best one. The student with the most votes will receive 2 additional points toward their course grade.

The project is due Friday 12/1/2023 at 11:59 PM EST.

Group Project (15%):

Students have the opportunity to form teams consisting of 3-4 members. The project involves:

- Choose an algorithm of your preference, different from the ones picked by your team members for the Individual Project and not covered in class.
- Conducting thorough research to gain a comprehensive understanding of it,
- Develop and submit a working implementation of the algorithm using a programming language of your choice.
- Writing a report spanning 5-10 pages. The report should include sections such as introduction, comparison with other algorithms, justification of its superiority, detailed explanation of the algorithm, implementation details, results, analysis of time complexity with proofs, and a list of references.
- teams are required to deliver a presentation during the final week of classes, with a time limit of 5-8 minutes.

Additionally, each team member will have the chance to evaluate their fellow team members' contributions. The grading will consider the overall delivery of the report, the presentation, and a peer review.

To ensure smooth coordination, it is essential for teams to submit a proposal outlining their selected project and team members by the end of the fourth week of the semester. However, **if any student does not have a team, they are encouraged to reach out to me before the end of the second week of the semester for assistance in finding a suitable team.**

The project is due Friday 12/1/2023 at 11:59 PM EST.

Thresholds

I will determine exact floors and half letter grades at the end of the semester according to the standards of performance below. Instructors reserve the right to modify the floors up or down to match the appropriate level of mastery.

Grade	Letter Grade	GPA
93-100	A	4.0
90-92	A-	3.7
87-89	B+	3.3
83-86	B	3.0
80-82	B-	2.7
77-79	C+	2.3
73-76	C	2.0
70-72	C-	1.7
67-69	D+	1.3
63-66	D	1.0
60-62	D-	0.7
<60	F	0.0

Academic Misconduct

Academic misconduct is dishonest or unethical academic behavior that includes, but is not limited to, misrepresenting mastery in an academic area (e.g., cheating), failing to properly credit information, research, or ideas to their rightful originators or representing such information, research, or ideas as your own (e.g., plagiarism), or submit homework you did not write, or discussing exams before all grades are posted, or Posting questions on forums like Reddit, Stack- Overflow, or Chegg.

The penalty for academic misconduct is an F in the course.

Student Responsibilities and Resources

As a member of the University of Connecticut student community, you are held to certain standards and academic policies. In addition, there are numerous resources available to help you succeed in your academic work. Review these important standards, policies and resources, which include:

- The Student Code
  - Academic Integrity

  - Resources on Avoiding Cheating and Plagiarism
- Copyrighted Materials
- Credit Hours and Workload
- Netiquette and Communication
- Adding or Dropping a Course
- Academic Calendar
- Policy Against Discrimination, Harassment and Inappropriate Romantic Relationships
- Sexual Assault Reporting Policy

Students with Disabilities

The University of Connecticut is committed to protecting the rights of individuals with disabilities and assuring that the learning environment is accessible. If you anticipate or experience physical or academic barriers based on disability or pregnancy, please let me know immediately so that we can discuss options. Students who require accommodations should contact the Center for Students with Disabilities, Wilbur Cross Building Room 204, (860) 486-2020 or http://csd.uconn.edu/.

Blackboard measures and evaluates accessibility using two sets of standards: the WCAG 2.0 standards issued by the World Wide Web Consortium (W3C) and Section 508 of the Rehabilitation Act issued in the United States federal government." (Retrieved March 24, 2013 from Blackboard's website.

Help

Technical and Academic Help provides a guide to technical and academic assistance. This course is completely facilitated online using the learning management platform, HuskyCT. If you have difficulty accessing HuskyCT, you have access to the in person/live person support options available during regular business hours through the Help Center. You also have 24x7 Course Support including access to live chat, phone, and support documents.

Evaluation of the Course

Students will be provided an opportunity to evaluate instruction in this course using the University's standard procedures, which are administered by Office of Institutional Research and Effectiveness (OIRE). Additional informal formative surveys may also be administered within the course as an optional evaluation tool.

Excluding materials for purchase, syllabus information may be subject to change. The most up-to-date syllabus is located within the course in HuskyCT.