CAS CS 630 - Fall 2024 - Graduate Algorithms
Syllabus

Official Course Description

Examines advanced algorithmic topics and methods for CS graduate students, including matrix decomposition techniques and applications, linear programming, fundamental discrete and continuous optimization methods, probabilistic algorithms, NP-hard problems and approximation techniques, and algorithms for very large data sets.

Informal Description

This semester we are going to build your skills further in developing and evaluating efficient algorithms. We will study exciting topics on algorithmic techniques that are often used in practical applications. Most of these methods will make trade-offs between provable guarantees of optimality and time and space complexity. This course builds on your prior knowledge of various algorithms design paradigms as well as proving their correctness and estimate their running time. Some topics we will cover are network flow, NP, approximation and randomized algorithms, local methods.

Note that this course is intended for **MS** and advanced **BA** students. PhD students should take CS530 instead.

Prerequisites

Undergraduate prerequisite: **CS330** Introduction to Analysis of Algorithms

Graduate prerequisite: an **algorithms course at the level of CS330**. If you're not sure whether you have the background, you *must* talk to the instructor.

Topics that you should be familiar with from your previous studies:

- Proof techniques (e.g. direct proof, proof by contradiction, induction)
- Data structures (e.g. lists, queues, heaps, hash tables, trees, graph adjacency list)
- Asymptotic analysis of running time (i.e. big-Oh)
- Algorithm design paradigms, such as greedy, divide and conquer, dynamic programming, various graph algorithms

Sign up ASAP:

Piazza (Q&A, discussion, as well as distribution of lecture notes, homework, additional material, course logistics info):

https://piazza.com/bu/fall2024/cs630

Gradescope (Homework submission) :

https://www.gradescope.com/courses/847056 Entry Code: **G378R5**

TopHat: A platform for in-class interaction and questions. You have to purchase a membership, there are semester and year-long options. If you use TopHat in multiple courses, you only have to pay the fee once. *If you do not have an account and the fee presents a financial hardship, please let the instructor know via a private Piazza post*.

https://app.tophat.com/ course code 019862

Instructors and Teaching Fellows (reachable on Piazza) Office hour info is posted on Piazza

Name	@ bu.edu	office
Prof. Jeffrey Considine	jconsidi	CDS1625
Prof. Dora Erdos	edori	CDS910
TF Spyros Dragazis	dragazis	
TF Garry Kuwanto	gkuwanto@bu.edu	

Communication related to class material should be primarily conducted through Piazza, for personal questions you can use the “private” post option.

Textbook

There is no one textbook that covers every topic in this course. Hence, instead we will post the reading material to our class Google Drive prior to each lecture. There are also many resources - such as lecture notes, book chapters, tutorials and videos - publicly available. We encourage you to seek out those resources too for additional learning.

Some useful textbooks:

- *Algorithm Design*, by Kleinberg and Tardos. ISBN 0-321-29535-8.
- Cormen, Leiserson, Rivest, and Stein. *Introduction to Algorithms*, 3rd ed. MIT Press.
- Dasgupta, Papadimitriou, Vazirani. *Algorithms*.
- V. Vazirani. *Approximation Algorithms*.
- Williamson and Shmoy. *The Design of Approximation Algorithms*.
- Motwani, Raghavan. *Randomized Algorithms*.
- Harvey. *A First Course in Randomized Algorithms*.

For prerequisite review:

- Tim Roughgarden. *Algorithms Illuminated*. (the 4th part of the book also contains material on NP)
- J. Erickson. *Algorithms,* 2019. Available from http://algorithms.wtf/
  See also the extensive exercises on the website.
- Mathematics for Computer Science by Eric Lehman, Tom Leighton, and Albert Meyer. (Useful background on discrete mathematics.)

Structure:

Lectures: This course has two instructors - professors Considine and Erdos - who will run the course together. On any given day one of the two instructors will give the lecture. The class consists of two 75 minute lectures (Tues, Thur) and one 50 minute discussion (Wed) each week. Attendance in lectures is mandatory and will be tracked through TopHat participation (see later section for details). The TFs will lead the discussion sessions. The objective is to reinforce the concepts covered in the lectures through problem-solving, and to provide clarification and guidance on the homework assignments.

Discussion Labs: Labs will be an invaluable part of the course involving interactive problem-solving sessions, tips on homework questions, and supplemental material not covered in lecture. We will post lab notes on Piazza in advance -- please read before coming to lab. Solutions will be posted after all labs conclude.

Office hours: We will hold multiple office hours throughout the week, the purpose of these is to answer specific questions about the material and help with assignments. Students are welcome to attend office hours with any of the instructors and TFs. The exact office hour schedule will be communicated on Piazza.

Communication:

We will be using **Piazza for all discussion** pertaining to the class. You should post your questions about the material, lectures, homework or course logistics here. Piazza is highly preferred over sending the course staff emails. Most often your question and the answer will be just as useful to your fellow students as yourself. We encourage you to respond to questions. The course staff will be monitoring Piazza and chime in as needed. You are encouraged to ask questions about the homework assignments, e.g. clarifications, related material, but **do not post solutions** to the problems. If you think your question is too specific or would reveal too much, then please ask in a private post. See the pinned "Ethics and Etiquette" post on Piazza for further guidance.

Course atmosphere, diversity and inclusion: We intend to provide a positive and inclusive atmosphere in class and on the associated virtual platforms. Students from a wide range of backgrounds and with a diverse set of perspectives are welcome. We ask that students treat each other with thoughtfulness and respect, and do their part to make all their peers feel welcome. Your suggestions are encouraged and appreciated. Please let us know ways to improve the effectiveness of the course for you personally or for other students or student groups.

Grading

The course grade will break down as follows:

5% class participation (based on TopHat)
30% weekly homework assignments (due Wednesdays, first hw due **Wed, Sept. 18, 2024**).
30% in-class midterm exam (in-class, planned for **Thur, Oct. 10, 2024**).
35% comprehensive final (during finals week).

Students **must pass both exams** with at least **40% of the grade** to pass the course. This will be strictly enforced.

Last day to **drop without a "W": Oct 8, 2024** . **With a "W": Nov 12, 2024**. Incompletes for this class will be granted based on BU policy as well as CS departmental policy (mostly only for last minute emergencies).

Exams: Both exams will consist of problem solving and short questions about the material. The midterm will be during class time and takes 75 minutes. The final is during the University-assigned final exam slot. The content of the final is cumulative.

No collaboration whatsoever is permitted on exams, any violations will be reported to the College.

Accommodations: All are welcome in the course. If you require particular accommodations for exams or coursework, please contact the instructor (and forward any relevant documentation from Disability and Access Services) in a timely manner. If you are facing unusual circumstances during the semester, please reach out to us early on so that we can find a good arrangement.

Attendance and participation: Your participation grade depends on answering TopHat questions, which requires your presence in class. While our textbook will be very helpful, it is an imperfect substitute for in-class learning, which is the fastest (and easiest) way to learn the material. Some material covered in lecture and lab may not be in our textbooks. You are in all cases responsible to be up to date on the material. Class participation and questions are very much encouraged. *Please ask as many questions in class, labs and on Piazza as you need.* Chances are that your question and answer will be as helpful to your classmates as to you.

TopHat is a web-based platform for interactive questions during class. Our goal in using it is to make lectures more interactive, get you thinking actively about the material, and get some feedback on what you are learning. TopHat questions are generally multiple choice. Most of the points (80%) are for participation. The remaining 20% is for correctness. You will get the full 5% of the course grade if you get at least 80% of the possible TopHat points for the semester.

Homework:

Homework problems: There will be weekly homework assignments, 9 total for the semester. Problems will be posted **Thursdays** and due the following **Wednesday** evening. Homework problem sets will consist of written problems. They will allow you to practice (a) solving problems using the ideas from class, often in a new way, (b) to assess and analyze the correctness of your solution and (c) communicate your ideas using technical language (precise descriptions, pseudocode, formal claims, proofs). Be aware that this latter is just as important as the two former and we will place high emphasis on it.

The two lowest grades on your homework assignments will be **dropped**.

Content: The homework is designed to aid you in understanding of the material. The problems will usually ask you to apply an algorithm we study in some clever way or design a slight modification of it. Lab assignments will often be helpful to solving these problems. Solving problems is probably the most useful learning tool in this course—take it seriously, allow yourself time to do it, and have fun! Alumni often describe this course's homework as critical to their success in job interviews.

Limited collaboration on homework is permitted; see below.

Homework Submission: Assignments will be due **Wednesdays by 11:59PM**, electronically via Gradescope. Solutions to written problems should be *typeset*. Hand-written solutions will not be accepted. We'll provide you with resources on how to get started if you're not familiar yet with typesetting mathematical-algorithmic text.

Late Policy: Late assignments will **not** be accepted as we intend to post solutions the next morning. You can use your dropped grades to cover for up to two late assignments. Also, be mindful that sometimes it's ok to submit partial results if you weren't able to fully finish your assignment, don't miss the deadline because of last minute work.

Regrade Policy: If, after reviewing the solutions and your answer, you still believe a portion of your homework was graded in error, you may request a regrade, **via Gradescope**, *NOT* through email. One of the staff will consider your request and adjust your grade if appropriate. Note that when we regrade a problem, your score may go up or down. Regrade requests can be submitted up to one week (7 days) after grades for that assignment have been posted.

Workload: CS 630 is a substantial amount of work. There is a problem set every week as well as two exams to study for. As you likely already know, assignments requiring substantial creativity can take more time than you expect, so plan to finish a day early.

Personal difficulties: Unfortunately, it may happen that you find yourself under circumstances that are affecting your ability to perform well in this class. We are here to support you and find the best way to help you in this course. Please reach out, so that we can help.

Collaboration, Citation, and Academic Honesty

Citation policy: You can reference anything from the textbook, lecture and discussion notes, information given by the course staff without having to cite it. However, if you make use of any other information, you have to include **proper citation**. If you omit to do this you are committing plagiarism.
You are allowed and encouraged to further your knowledge by finding related material online or in books, but you have to cite it. Your citation may be a url, the title and chapter of a book, or a paper reference.

Searching explicitly for answers to problems on the Web or from persons not enrolled in the class this current semester is strictly forbidden. (This includes students who took the course in previous semesters, posted solutions, Chegg, GitHub, etc.) Using AI tools, e.g. ChatGPT, for solving problems is forbidden.

Collaboration Policy: Collaboration on homework is permitted and even encouraged. If you choose to collaborate on some problems, you are allowed to discuss each problem with **at most 3** other students currently enrolled in the class. Before working with others on a problem, you should think about it by yourself for at least 45 minutes.

You must write up each problem solution by yourself (using your own words) without assistance, even if you collaborate with others to solve the problem. Identical worded answers, including identical pseudocode, will receive no grade. You must also identify your collaborators clearly on the first page of your assignment. If you did not work with anyone, you should write ``Collaborators: none.'' It is a violation of this policy to submit a problem solution that you cannot orally explain to the instructors. You may get help on Piazza or in office hours from the instructors for the class for specific problems. You don't need to list them as collaborators .

No collaboration whatsoever is permitted on exams!

Collaboration strategies: If you do collaborate, use it as an opportunity to practice group work skills: give everyone a chance to speak, listen carefully, acknowledge good suggestions. If you have a tendency to be shy, speak up! If you have the tendency to dominate conversations, make sure to give others the floor. We strongly encourage you to find a small group of classmates that you regularly discuss and review material with. Feel free to post on Piazza to find a study mate.

Academic Conduct:
Academic standards and the code of academic conduct are taken very seriously by the University, the Graduate School of Arts and Sciences, and the Department of Computer Science. Course participants must adhere to the university Academic Conduct Code. Please take the time to review this document if you are unfamiliar with its contents.

Any case of academic misconduct, including but not limited to plagiarism and submission of work that was not solved by you – be that a tutor, friend, web resource, AI, etc., will be reported to the College and will also carry a grading penalty.

Tentative schedule:

This schedule is for informational purposes only and is subject, and is likely, to change as we progress through the semester.

week	topic
1	course intro, max flow, Ford-Fulkerson
2	Max flow: MFMC, flow applications, intro to poly-time reductions
3	Poly-time reductions, app: closest vector with cosine sim, NN. NP, NP-C.
4	Approximation algorithms, and how to prove approximation factors, mon submodular objective functions
5	Approx algos continued. (set cover variants, load balancing, make span, bin packing, etc.)
6	Approx algos continued, midterm
7	Randomized load balancing, probability review
8	Probability continued, tail bounds, overflow, worst vs expected runtime, etc.
9	Load balancing: multiple variants. Cuckoo hashing
10	Bloom filters, counting sketches
11	Quick sort in depth analysis, entropy, Huffman codes
12	Splay trees, KKT for randomized minimum spanning trees
13	Random walks on graphs (Page Rank, stationary distr etc, Metropolis-Hastings)
14	Random walks on graphs (Page Rank, stationary distr etc, Metropolis-Hastings)
15	review