CAS CS 330 - Summer 2021 - Introduction to Analysis of Algorithms – Syllabus

Official Course Description

Examines the basic principles of algorithm design and analysis; graph algorithms; greedy algorithms; dynamic programming; network flows; polynomial- time reductions; NP-hard and NP-complete problems; approximation algorithms; randomized algorithms This course fulfills a single unit in each of the following BU Hub areas: Quantitative Reasoning II, Critical Thinking.

Prerequisites

The class assumes **working knowledge of CS 112 and CS 131** (or MA 293). CS majors need to complete at least one of their Group B coursework (any two of CS 132/MA242, CS235/MA294 and CS237/MA581) before taking CS 330. If you don't have the prerequisites, please talk to an instructor before deciding to continue with this class.

TODO asap:

Sign up to the course Piazza and Gradescope. We'll use these for communication and homework. Details can be found later in this syllabus.

Piazza (Q&A, discussion, as well as distribution of lecture notes, homework, all links): http://piazza.com/bu/summer2021/cascs330

Gradescope (Homework submission) : https://www.gradescope.com/courses/268565 (Entry code: 4P488K)

Instructors and Teaching Fellows

Name	Office Hours via Zoom (all times are in ET)	Email@bu.edu *
Prof. Dora Erdos	Mon, Wed, Fri 12:15-1:15 pm	edori
Hannah Catabia	Mon 3:30-4:30 pm, Thur 10-11 am	catabia

***** Messaging via Piazza is preferable to email (and will get a faster response).

Textbook

Algorithm Design, by Kleinberg and Tardos. ISBN 0-321-29535-8.

Useful additional resources:

- Cormen, Leiserson, Rivest, and Stein. *Introduction to Algorithms*, 3rd ed. MIT Press.
- 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.)

Course Structure, Communication and LfA:

Structure:
The lecture will be taught by Professors Erdos. The TF, Hannah Catabia will lead the discussion sessions. The lecture will cover the theory of algorithms, while the discussion sessions will be problem-solving style. With the goal to reinforce the concepts covered in the lectures and to provide guidance on the homework assignments.

Communication:
We will be using **Piazza for all discussion** pertaining 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.

Lfa – Learn from Anywhere (a.k.a. remote and in-person learning):

This course is provided in the LfA format. You'll be able to come to the lecture hall in person or tune in via Zoom. Lectures (and discussions) will be recorded and posted so that you can view them later. Links to Zoom and the recordings can be found on the course Piazza. All office hours will be held via Zoom. We encourage participation from all students online or in person!

Lectures
Mon, Wed, Fri 9 :30 am -12 pm ET , EPC 208

Discussion Labs
Tue, Thu 9-10 am ET, EPC 207

Coursework and Grading

Grading:

The course grade will break down as follows:

30% weekly homework assignments, due Mondays, first due date May 31st, 2021).
30% in-class midterm exam (in-class, planned for **Monday, June 14th, 2021**).
40% comprehensive final (in-class, **Friday, July 2nd, 2021**).

Last day to **drop without a "W": Fri, May 28, 2021**. **With a "W" Wed, June 16, 2021**. Incompletes for this class will be granted based on CAS Policy (mostly only for last minute emergencies).

Exams: Both exams will consist of problem solving and short questions about the material. The duration of the exam is 2 hours. The exam will be during class time, but remote. Details on the logistics will be communicated via Piazza.

No collaboration whatsoever is permitted on exams.

Attendance: We will not take formal attendance in this course. However, 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. We understand that some students will reside in different time zones during the semester. Hence, all classes will be recorded and posted in a class repository so that you can watch it later. 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 as you need. Chances are that your question and answer will be as helpful to your classmates as to you.

Homework:

Homework problems: Homework problem sets will be written problems, assigned weekly. Assignments will always be posted Mondays and due the next Monday. They will allow you to practice (a) solving problems using the ideas from class, often in a new way, (b) communicating your ideas using technical language (precise descriptions, pseudocode, formal claims, proofs). Be aware that this latter is just as important as the former.

The lowest grade on your homework assignments during the semester will be **dropped**.

Content: The homework is meant as a tool to help you follow the material —take it seriously, allow yourself time to do it, (try to) enjoy the challenge! In this course it is very easy to fall behind. For this reason, the problem set due will follow the material of the current week, not the one

before. In practice, Homework assignments will be released on Mondays. The questions in the set will cover material up until the lecture Wednesday of that week.

Homework Submission: Assignments will be due **Mondays by 11:59PM ET**, electronically via **Gradescope**. Solutions should be typeset (preferred) or neatly hand-written and scanned.

Late Policy: Late assignments will NOT be accepted as we intend to post solutions the next morning. You can use your dropped grade to cover for one late assignment. 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 your solution, 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.

Workload: Be forewarned -- the workload in this course will be **heavy**. There is a problem set every week. As you likely already know, assignments requiring substantial creativity can take more time than you expect, so plan to finish a day early.

Citation policy: You are off course allowed and encouraged to further your knowledge by finding related material online or in books. However, if you use any resource other than material distributed in class **you have to cite it**. Your citation may be a url, the title and chapter of a book, paper reference.

Searching explicitly for answers to problems on the Web or from other outside sources (these include anyone not enrolled in the class) is strictly 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 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. 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. You can use Piazza posts to find team and you can meet over Zoom or in person.

Academic Conduct:

Academic standards and the code of academic conduct are taken very seriously by our University, by the College of Arts and Sciences, and by the Department of Computer Science. Course participants must adhere to the CAS Academic Conduct Code -- please take the time to review this document if you are unfamiliar with its contents.

If in doubt, our department has an extensive compilation of **examples** with regard to Academic Conduct and permissible collaboration.

Violations of this policy will be dealt with according to University regulations.

Topics:

We will mostly follow the order and content in the textbook. Topics are subject to change.

- Asymptotics, data structures, how to describe an algorithm (pseudocode)
- Graphs – data structures, graph traversals, connectivity, DAGs
- Greedy algorithms – scheduling, shortest paths, minimum spanning trees
- Divide-and-conquer – variations on MergeSort, integer multiplication
- Dynamic programming – interval scheduling, sequence alignment, knapsack
- Network flow – Ford-Fulkerson, MFMC theorem, applications
- Polynomial time reductions, NP