Boston University, CAS CS 330 Spring 2026
Introduction to Analysis of Algorithms

Course Description

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

Prerequisites

Students must complete **CS 112 and CS 131** (or MA 293) prior to taking CS330. In addition CS majors need to complete **at least one** of their Group B coursework (any of CS 132/MA242, CS235/MA294 and CS237/MA581); students from other majors must have completed at least one of these courses or their equivalents. If you don't have the prerequisites, you **must** talk to an instructor before continuing with this class.

Resources (sign up ASAP):

- **Google Drive:** https://tinyurl.com/cs330-sp2026
- **Google Calendar:** https://tinyurl.com/cs330-sp2026-calendar
- **Piazza** (Q&A, discussion, course materials): https://piazza.com/bu/spring2026/cascs330
- **Gradescope**: https://www.gradescope.com/courses/1215382 entry code: **5DRKYN**
- **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 (join code): **808367**

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

Name	Email @ bu.edu (piazza preferred)	Office
Prof. Dora Erdos	edori	CDS 910
Prof. Adam Smith	ads22	CDS 1038
TF Connor Wagaman	wagaman	CDS 1039
TF Ephraim Linder	ejlinder	CDS 1039
TF Duy Le	duyle	CDS 1025

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.)

Structure

The class consists of two 75 minute lectures (Tues, Thur) and one 50 minute discussion (Fri) each week. Attendance in lectures is mandatory and will be tracked through TopHat participation (see later section for details). The two sections of the course, A1 and A2, will be treated as one class. The content of the two lectures is identical[1]: assignments, course staff and all resources will be shared, students can mix-and-match A1 and A2 lecture and discussion sections.

A1 section : Tuesdays and Thursday, 11:00 AM -12:15 PM in WED 130
A2 section : Tuesdays and Thursday, 12:30-1:45 pm in WED 130

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.

There will be multiple office hours throughout the week. The purpose of these is to answer questions about the material and help with assignments. Students are welcome to attend office hours with any of the course staff. The office hour schedule will be posted 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.

[1] If on a given day you can't attend your own lecture section you are welcome to attend the other lecture. No need to inform the instructor.

Topics

We will mostly follow the order and content in the textbook. Topics are subject to change. A detailed semester schedule can be found at the end of this syllabus.

- 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 - basic D&C algorithms, recurrences, integer and matrix multiplication
- Dynamic programming – interval scheduling, subsetSum, knapsack, sequence alignment, Bellman-Ford
- Network flow – Ford-Fulkerson, MFMC theorem, applications
- Polynomial time reductions, NP, NP-Completeness

Course atmosphere 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:

10% Participation (based on TopHat and lab submissions)
10% Weekly homework assignments (due Wednesdays by midnight, first hw due **Wed, Jan. 28**).
25% Midterm exam 1 (Tuesday, **March 3**, 2026, 6:00-8:00 PM, location TBA).
25% Midterm exam 2 (Tuesday, **April 7**, 2026, 6:00-8:00 PM, location TBA).
30% Comprehensive final exam (during finals week).

To pass the course students must earn at least 50% on two of the three exams, and get at least 50% average on the three exams.

Last day to **drop without a "W": February 24, 2026. With a "W": April 3, 2026**. Incompletes for this class will be granted based on CAS Policy as well as CS departmental policy (mostly only for last minute emergencies).

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 and work during lab sections, which requires your presence both in lecture and labs. 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 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 full TopHat participation points if you get **at least 80%** of the possible TopHat points for the semester.

Labs (discussion sections) are 50 minute long problem solving sessions (on Fridays) led by the TFs. The problems are designed to help you get a deeper understanding of the concepts discussed in lecture; you will also find them helpful in solving your homework assignments. During each lab session you will write on a worksheet that we hand out. **At the end of each lab you have to scan and upload your work to Gradescope.** You will receive 0, 1 or 2 points for each worksheet: 0 points for not submitting anything (meaning you were absent), 1 point for showing minimal effort, 2 points for showing diligent work. You will not be graded on correctness, only that you spend honest effort during the session. You will get full Lab participation points if you get **at least 80%** of the possible points for the semester.

If you cannot attend a lecture or lab for a legitimate reason, please create a private Piazza post (visible only to Instructors) and tag it with the **#attendance** folder.

Homework

Homework problems: There will be weekly homework assignments, 12 total for the semester. Problems will be posted **Thursdays** and due the following **Wednesday by 11:59PM**, electronically via Gradescope (no late assignments are accepted). Solutions to written problems should be *typed*; 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.

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

Content: Homework problem sets will consist typically of two written problems or one written problem and one short programming assignment. They will allow you to practice (a) solving problems using the ideas from class, often in a new way, (b) assessing and analyzing the correctness of your solution, and (c) communicating your ideas using technical language (precise descriptions, pseudocode, formal claims, proofs).

The homework is designed to aid you in understanding of the material. 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.

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 email or piazza). 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.

Exams:

The two midterm exams will be in the evening 6:00 - 8:00 pm. The final is during the University-assigned final exam slot and is 120 minutes. The exams are closed book, the content of each exam is cumulative.

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

Workload:

CS 330 is a substantial amount of work. There is a problem set every week as well as three 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, Piazza and information given by the course staff without citing it. However, if you make use of any other information, you have to include **proper citation**. If you omit to do so you are committing plagiarism.

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 .

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.

Generative AI Policy

Generative AI tools have become very advanced and widely available, many people use them for various purposes in productive ways. It's worthwhile to learn these tools and use them in a responsible way. However, in the context of this course we believe that using these tools can harm your learning. CS330 is in general not the place where you should employ these. In this class we ***forbid the use of AI tools for solving assigned problems***. It *is* acceptable to use tools to answer questions about the course material (as opposed to getting the solution to a homework problem) or to get feedback on your solutions. **Exams will have part of the content verbatim or very closely related to homework.** You won't be able to successfully solve them if you don't think deeply about the problems yourself.

Academic Conduct:
Academic standards and the code of academic conduct are taken very seriously by the University, the College of Arts and Sciences, and 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.

Any case of academic misconduct, including but not limited to plagiarism and submission of work that was not solved by you—regardless of whether you got the solution from a tutor, friend outside the class, web resource, uncited work, LLM, etc—will be reported to the College and will also carry a grading penalty

Tentative schedule for Spring 2026

KT in the reading refers to the textbook Kleinberg, Tardos, Algorithm Design.

Date	Topic	Reading
01/20/2026	course intro, syllabus, why algorithms? stable matching	KT 1.1
01/22/2026	stable matching continued. intro to asymptotics	KT 1.1, 2.1
01/27/2026	asymptotics continued, graph adj list as nested hash table	KT 2.1, 2.2, 2.4
	HW1 Due	
01/29/2026	BFS (including proof and implementation)	KT 3.1-3.3
02/03/2026	DFS, strong connectivity	KT 3.2, 3.3, 3.5
	HW2 due	
02/05/2026	DAGs	KT 3.6
02/08/2026	Greedy algos, int scheduling	KT 4.1
	HW3 due	
02/10/2026	int part., priority queues	KT 4.2
02/17/2026	NO CLASS—Monday schedule	
	HW4 due	
2/19/26	Dijkstra	KT 4.4
02/24/2026	Dijkstra review, MST	KT 4.5
	HW5 due	
02/26/2026	MST	KT 4.5, 4.6
03/02/2026	HW6 due - Monday HW deadline!	
3/3/2026	review	
3/3/2026	Midterm 1 6:00-8:00 pm	
3/5/2026	D&C: mergesort, recurrences	KT 5.1, 5.2
	SPRING BREAK 3/9-3/13	
3/17/2026	D&C: counting inversions, more recurrences OR closest pair of points??	KT 5.2, 5.3

3/19/2026	D&C: integer and mtx multiplication	KT 5.5
3/24/2026	DP: WIS and Fibonacci	KT 6.1-6.2
	HW7 due	
3/26/2026	DP backtracking, subsetsum	KT 6.4
03/31/2026	DP knapsack, examples	KT 6.4
	HW8 due	
04/02/2026	DP: sequence alignment	KT 6.4
04/06/2026	*HW9 due Monday!*	
4/7/2026	review	
4/7/2026	Midterm 2 6:00-8:00 pm	
4/9/2026	DP: Bellman - Ford	KT 6.6
4/14/2026	Max-flow	KT 7.1-7.2
	HW10 due	
4/16/2026	Flow, MFMC, (Correctness proofs, review FF)	KT 7.1-7.2, 7.5, 7.6
4/21/2026	Flow applications	KT 7.1-7.2, 7.5, 7.6
	HW11 due	
4/23/2026	reductions, computational equivalence	KT 8.1
04/28/2026	NP	KT 8.3-8.4
	HW12 due	
04/30/2026	NP, NP-C	KT 8.3-8.4
5/4-8/2026	Finals week	