CSOR 4231 – Analysis of Algorithms I, Spring 2022

Basic Logistics:

Times: Mon/Wed, 4:10-5:25pm
Format: In-person
Location: Fayerweather 313
Teaching Staff:

Instructor	*Email*	*Office Hours & Location*
Prof. Rachel Cummings	rac2239@columbia.edu	Tues 2:15-3, Thurs 4:15-5pm Mudd 535E
Teaching Assistants	*Email*	*Office Hours & Location*
Millend Roy (TA)	mr4404@columbia.edu	Wed 1-2pm, Fri 10-11am Mudd 301
Anjali Smith (CA)	as6467@barnard.edu	Tues 11am-1pm Mudd 301

Textbook: Introduction to Algorithms, 3rd edition, by Cormen, Leiserson, Rivest, and Stein (Required). Algorithm Design, by Kleinberg and Tardos (Optional).

Courseworks/ Ed Discussion: This course will be hosted on Courseworks. Homework assignments and solutions, lecture notes, and other class announcements will be posted on Courseworks. There is an Ed Discussion forum set up for this course, which can be accessed through Courseworks. You are encouraged to discuss *class lectures* and *related topics*, as well as *ask questions or clarifications* so that other students or TA or the instructors can answer, and that other students with similar questions will benefit from the discussion. **You may not discuss homework solutions on Ed Discussion.**

Contacting the instructor: If you have general question about the content, homeworks or logistics of the course, you are encouraged to post it on Ed Discussion. This will likely get a faster response than email, since other students can respond to the post as well. Feel free to also respond to the questions from your friends and classmates on Ed Discussion. If you want to contact the instructors for any other reason, email is the best way. I aim to answer all emails within 48 hours (except weekends/holidays); I cannot guarantee a faster response than that, so please do not send last-minute urgent requests. Please direct any questions about homework grading to the TAs (aside from regrade requests, detailed below).

Course Goals:

The main goals of the class are for you to learn what is an algorithm, how to design algorithms, and how to analyze them:

- We will learn some of the fundamental *algorithmic techniques* by going through a few concrete, classic algorithms.
- We will learn to think about the *efficiency and correctness* of algorithms. In particular you will learn how to formally analyze algorithms using rigorous mathematical proofs.

- Overall, we will develop an *algorithmic mindset*: how to think from specification of a given problem to designing an actual algorithm solving it.

Prerequisites:

Two things are crucial for making the class easier for you:

- being able to read and write **formal mathematical proofs**: mathematical formalism provides a robust method to analyze and verify correctness of algorithms, and hence in this class we place a high emphasis on formal proofs;
- some familiarity with algorithms/data structures or a strong coding background: while we will learn algorithms without assuming that you know algorithms, it will be easier for you to absorb the new material if you have seen, and reasoned about, some examples of algorithms previously.

Formal prerequisites are: COMS W3134, COMS W3136, or COMS W3137 (Data Structures), and COMS W3203 (Discrete Mathematics)

Schedule (subject to change):

Lec	*Date*	*Topic*	*Reading (CLRS)*	*HW out*	*HW due*
1	1/17	Introduction, big numbers multiplication	Ch. 1,2	HW1 out	
2	1/22	Divide and Conquer, Asymptotic notation	Ch. 2,3		
3	1/24	Recurrences, Merge Sort	Ch. 4		
4	1/29	Heapsort, Loop invariants	Ch. 6		
5	1/31	Deterministic Selection	Ch. 5, 9.3	HW2 out	
6	2/5	Indicator Random Variables, Randomized Analysis	Ch. 5		HW1 due
7	2/7	Quicksort	Ch. 7		
8	2/12	Randomized Selection	Ch. 9.2		
9	2/14	Dynamic programming <3	Ch. 15	HW3 out	
10	2/19	Dynamic programming	Ch. 15		HW2 due
11	2/21	Dynamic programming	Ch. 15		
12	2/26	Greedy Algorithms	Ch. 16		
13	2/28	Greedy Algorithms	Ch. 16	HW4 out	
14	3/4	Amortized Analysis	Ch. 17		HW3 due
	3/6	**Midterm (in class)**			
15	3/18	Graphs: BFS, DFS	Ch. 22.1-22.3		
16	3/20	Topological Sort, Strongly Connected Components	Ch. 22.4, 22.5		
17	3/25	Minimum Spanning Tree, Disjoint Sets	Ch. 21, 23		
18	3/27	Shortest paths: Dijkstra's, Bellman-Ford, DAGs	Ch. 24.1-24.5	HW5 out	HW4 due
19	4/1	Maximum Flows	Ch. 26.1-26.3		
20	4/3	Maximum Flows	Ch. 26.1-26.3		
21	4/8	Linear Programming	Ch. 29.1-2, 29.4		
22	4/10	Hardness, reductions between problems	Ch. 34	HW6 out	
23	4/15	NP-completeness	Ch. 34		HW5 due
24	4/17	NP-completeness	Ch. 34		
25	4/22	*No class (Passover)*			
26	4/24	Approximation algorithms	Ch. 35		
27	4/29	Final review			HW6 due
	5/6	**Final exam (in class) 4:10-7pm**			

Grading:

Your grade will be based on the following three components:

- Homework assignments (20%: 6 assignments with lowest score dropped)
- Midterm (35%)
- Final exam (45%)

Homework:

Homeworks will be assigned roughly every two weeks and will be posted on Courseworks. You will submit your homeworks electronically via Gradescope *in pdf format*. You may write your solutions by hand, in which case you should either scan or photograph your solutions (and convert to pdf format). Homeworks are due on their specified due date at 11:59pm NY time. Exceptions will be made only for exceptional unforeseen circumstances (e.g., serious illness), in which case you will need to provide some additional documentation (e.g., doctor's note). The homework with the lowest score will not count towards your final the grade.

You are strongly encouraged to start working on the homeworks early: some problems may require you to sit on the problem for a while before you get your "aha" moment. Starting early also gives you time to ask questions and make effective use of the office hours of the teaching staff.

Writing up solutions: precise and formal proofs. Remember that a goal of the class is for you to learn to reason about algorithms, precisely describe them, and formally prove claims about their correctness and performance. Hence, it is important that you write up your assignments *clearly, precisely, and concisely*. Legibility of your write-up will be an important factor in its grading. When writing up (algorithmic) solutions, keep in mind the following:

- The best way for you to convey an algorithm is by using plain English description, or a well-explained pseudocode. A worked example can also help. Generally, give enough details to clearly present your solution, but not so many that the main ideas are obscured.
- The analysis of the algorithm has to include both 1) proof of correctness, and 2) upper bound on performance (usually runtime, but sometimes space as well).
- You are encouraged (but not required) to type up your solutions using LaTeX. LaTeX is the standard package for typesetting and formatting mathematically-rich content. Since LaTeX knowledge is a good life skill, now may be a good chance to learn it. A short mini-course on LaTeX is available here: http://www.ctan.org/tex-archive/info/lshort/english/lshort.pdf. Macros to format pseudocode are available at http://www.cs.dartmouth.edu/~thc/clrscode/. You are encouraged to use the template available on Courseworks (the files homework.tex, homework.cls).

For examples of the expected level of rigorousness in your write-up, please consult the textbook and the proofs in it. Note that our lectures will generally be at a lower level of formalism (in lectures we will focus on intuition and main ideas; you will use the textbook to learn the formal details and proofs).

Clarity points. To encourage clarity (and conciseness), for each problem, 20% of the points are given for the clarity of your presentation. In particular, you will be awarded a default 20% of the points for an empty solution (note that, if you submit no coversheet whatsoever, you get only

0%). Note that you can lose these 20% if you write something that is unintelligible, does not lead to a solution, or is excessively long (including scoring a 0%). Clarity points policy does not apply to the exams.

Academic Integrity:

You are expected to abide by the principles of academic integrity, and the Columbia University Honor Code: https://www.wikicu.com/SEAS_Honor_Code and https://www.gradengineering.columbia.edu/academics/academic-integrity.

Collaboration: You are permitted (and even encouraged) to discuss the homework *assignments*. If you do collaborate, you must write the solutions *individually* (without looking at anybody else's solutions), and acknowledge anyone with whom you have discussed the problems. It will be considered an honor code violation to consult solutions from previous years, in the event that homework problems have been previously assigned, or solutions from the web or elsewhere, or solutions are available elsewhere. Honor code violations will be treated harshly and can lead to a score of 0, grade reduction, and/or a report to the Academic Committee (which may decide on further penalties that can permanently affect your academic record).

The main goal of the homework assignments is education, not evaluation. We must also be mindful of rules regarding academic honesty and plagiarism. To facilitate these goals, we will use the following policy:

- All work submitted for credit must be your own.
- You may discuss the homework problems with your classmates, the teaching assistant(s), and the instructors. For each problem, you must acknowledge the people with whom you discussed your work, and you must independently write up your own solutions.
- Any written sources used (apart from the textbook) must also be acknowledged; however, you may not consult any solutions from previous years' assignments whether they are student or faculty generated.
- Solutions to problems may be available somewhere on the web. Do not use these solutions. If you do use them, cite them. If you do choose to copy another student's work, or to copy from some other source, please state this in writing on your homework assignment.

Please ask if you have any questions about this policy. Violations will be treated harshly. This means that if you violate the policy, even once, your grade on homework for the entire semester will be 0, and the infraction will be reported to the dean. THERE ARE NO EXCEPTIONS TO THIS POLICY. Every semester, several students are caught violating this policy and they are reported to the dean. Typically, students use unacknowledged sources on the web. Do not do this. If a homework problem is too hard, start earlier, ask for help, or don't answer the question. If you can't answer some of the questions, you are still a good person who can go on and have a productive life. Academic dishonesty is never the correct solution. Note that allowing someone else to copy your solution is just as serious as copying someone else's solution.

Exams:

The midterm exam will be in class on March 6, 2024 in the regular classroom and at the regularly scheduled class time. The final exam for this course is tentatively scheduled for Monday May 6, 2024 at 4:10-7pm, according to the Columbia University Final Exam Schedule. The final exam will be written and will be comprehensive of all material covered in the course. Missing either exam will be accommodated only in case of a pre-arranged, University-approved absence. No makeup exams will be given.

Regrading:

Regrade requests will be accepted via email to the instructor and the TA within 5 days of the graded assignment/exam being returned. Requests must include (1) a PDF of the original submission and (2) a LaTeX-produced PDF detailing which problems were graded incorrectly and an argument that the submitted solution is indeed correct. Regrades may only be requested if it is believed that a correct answer was marked as incorrect, not because insufficient partial credit was given to an incorrect or partially correct solution. If you request a regrade, you accept that the entire assignment/exam will be regraded, not just the problem(s) believed to be graded incorrectly.

Disability Services:

Columbia has policies regarding disability accommodations, which are administered through Columbia Health and Disability Services (https://health.columbia.edu/content/disability-services). If you require special accommodations, make an appointment to discuss your special needs and to obtain an accommodations letter. Please also e-mail me as soon as possible in order to set up a time to discuss your learning needs.