CSCI 3104 Algorithms Syllabus (Fall 2023)

Contents

1 Logistics

1.1 Instructional Staff

Instructors: Chandra Kanth Nagesh.

Graduate Teaching Assistants: Chi-Hui Lin, Arvind Sreenivas, Gülce Kardes, Melody Hsu, Shamal Shaikh, and Drona Khurana

Undergraduate Course Assistants: Kevin Jacob, Kashyap Challapalli, Jisoo Park, Daniil Garusov, and Emma G. Nickel

1.2 Course Canvas

All announcements will be posted to the course Canvas instance

https://canvas.colorado.edu/courses/95161.

If you are enrolled in the class you should be automatically enrolled for the Canvas instance; if you are not, please contact one of the instructors by email. Students are responsible for checking the Canvas announcements (either by auto-email settings or by visiting the Canvas page) regularly.

1.3 Lecture, Recitation, and Student Hours

Lectures:

- Section 100 (Chandra Kanth Nagesh): TTh 9:30–10:45 AM in HUMN 1B50 (see Canvas page for more details)

All lectures will be recorded and posted on Google Drive (see Canvas for link). Attendance or viewing is highly recommended. Activities in recitation may assume that students have attended or watched all lectures leading up to the recitation, including the the lecture from the preceding days.

Recitations: See Canvas for Zoom links for the first two weeks and for remote sessions. Attendance at recitations it highly recommended. Because recitations involve active learning activities, and because of limited room sizes, we need to keep recitation sections relatively small. Towards that end:

> **Students should attend the recitation section they are registered for.**

Student Hours: See the Canvas Student Hours page for times & locations, or the class Google Calendar. Student hours are in-person or on Zoom at staff discretion; this will be clearly marked on Canvas, but may change over the course of the semester as the status of the pandemic changes. We will notify students of any changes via Canvas announcements.

2 Course Description

2.1 Prerequisites

The prerequisites include Calculus I-II, Data Structures, and some Discrete Math course. This course relies **heavily** on **all** of the prerequisites. Students from outside of Computer Science who are comfortable writing mathematical proofs are likely to be well prepared and are very welcome. Please discuss ASAP with the instructor if you have concerns about your background.

- Calculus I-II (Grade of C- or Better)
- CSCI 2270 or CSCI 2275 Data Structures (Grade of C- or Better)
- Discrete Math (Grade of C- or Better in one of the following: CSCI 2824, Math 2001, APPM 3170, or ECEN 2703).

2.2 Workload

CSCI 3104 is a 4-credit course. During the Fall or Spring semesters, well-prepared students should expect to spend on average 9-12 hours/week outside of class. Students who have significant gaps in their backgrounds may find that they need to carve out additional time to review the prerequisite material.

Algorithms (and Theory/Math courses in general) require more time to gain traction than applied/coding-based courses. Poor early performance is not indicative of one's ability to succeed (or even earn an A) in this course. Furthermore, student growth (both on an individual level and in the aggregate) in Standards-Based Grading tends to be concave-up. That is, students tend to learn the material and earn credit for standards at a faster rate as the semester progresses. As a rough gauge, students who have, after the first midterm, earned credit for at least 3 standards twice and demonstrated proficiency for another 3-4 standards at least once are likely to be on track to pass the course.

2.3 Course Content

CSCI 3104 Algorithms is an undergraduate course in theoretical computer science. The primary goals include surveying fundamental algorithm design techniques, analyzing algorithm runtime complexities, and identifying computational problems that are unlikely to have efficient algorithmic solutions. We will begin the semester with a survey of including greedy algorithms, including shortest path problems, and computing minimum-weight spanning trees. Then we talk about network flow problems and introduce an important network flow optimization algorithm. Afterwards, we will discuss the technicalities of analyzing an algorithm's efficiency, including asymptotic notation (e.g., Big-O), and techniques to ascertain and compare the asymptotic runtimes (e.g., Calculus techniques, Recurrences). Once we have a sense of how to analyze algorithms, we will proceed to discuss both the divide & conquer and dynamic programming paradigms. We will also examine our algorithm design techniques closely, discussing both instances where they apply and where they fail to yield the desired results.

We will then briefly talk about some advanced data structures, including Hash tables, doubling lists and its amortized analysis.

At the end of the semester, we will discuss Computational Complexity, which seeks to classify problems into complexity classes based on how efficiently they can be solved. The goal then is to compare these complexity classes, as opposed to individual problems. We will restrict attention to the complexity classes P (the set of decision problems that have efficient solutions) and NP (the set of decision problems where correct solutions can be verified efficiently). While it is known that $\mathsf{P} \subseteq \mathsf{NP}$, determining whether $\mathsf{P} = \mathsf{NP}$ remains the central open problem in Computer Science and one of the six biggest open problems in Math. Resolving the P vs. NP problem will have far-reaching, real-world implications, including on the security of online transactions (cryptography), curing cancer (protein folding), scheduling, routing, and a host of other combinatorial optimization problems of practical interest. Our goal will be to understand the statement of the P vs. NP problem, including contextualizing the role that our algorithmic techniques play. Our discussions on the structure of these complexity classes will be quite shallow.

Ultimately, this course is mathematical in nature. The obvious connections are with Discrete Math (Math 3110, Math 3170, Math 4440) and Theoretical Computer Science (CSCI 3434, CSCI 3090, CSCI 4114). However, our algorithmic techniques also serve as key tools in application areas, including Artificial Intelligence (CSCI 3202), Machine Learning (CSCI 3832, CSCI 4622), Bioinformatics (CSCI 4314), Network Science (CSCI 3352), Economics (CSCI 7000 Algorithmic Game Theory, Econ 4050), Operations Research (CSCI 5654), and Circuit Design (ECEN 2350). In order to understand how to adapt and apply our techniques (in this course, subsequent courses, job interviews, or your careers), it is necessary to understand how and why these techniques work. For this reason, formal proofs and the underlying ideas will be examined in great detail. Therefore, a key objective in this course is to develop your mathematical maturity; that is, your ability to understand mathematical statements and formulate rigorous mathematical proofs. This will ultimately be the best indicator for success (outside of hard work). We will rigorously prove mathematical statements in class and discuss proof strategy throughout this course. Every student will be expected to formulate proofs on homework and assessments.

Remark. CSCI 3104 is effectively an abstract math course. This is **not** a software engineering/coding course. We also stress that while the material we cover has a myriad of applications, the focus will be on developing and understanding the techniques rather than on the applications themselves. Our goal will be to prepare students to apply the techniques we develop beyond this course.

2.4 Learning Objectives

Algorithms is one of the key maturity courses for undergraduates in Computer Science programs (the others being Systems and Principles of Programming Languages). The obvious course objective is gaining proficiency with the material outlined above. Beyond that, the development of rigorous mathematical thought, mathematical maturity, and sharpness of proof writing will be emphasized. The underlying goal is for you to improve your ability to read and write mathematics, as well as appreciate the design and usage of axioms in a theoretical discipline. A third goal is to provide a solid preparation for subsequent courses that utilize rigorous algorithmic techniques. To this end, we have the following learning objectives.

- Students will work through key algorithms by hand, including Breadth-First Search, Dijkstra's Algorithm, Prim's Algorithm, Kruskal's Algorithm, the Ford-Fulkerson procedure, Mergesort, Quicksort and variations thereof.

- Students will prove theorems by induction.
- Students will prove theorems about greedy algorithms or problems amenable to greedy algorithms using exchange arguments.
- Students will construct functions to model algorithm runtimes, as well as determine closed-form asymptotic solutions for said functions.
- Students will design algorithms using the greedy, divide & conquer, and dynamic programming techniques.
- Students will ascertain when algorithm design techniques fail to apply, clearly justifying their reasoning.
- Students will begin to think critically about the ethical ramifications of the design choices & applications of algorithms.

2.5 Course Text

We will provide course lecture notes via Canvas, which will serve as the official course text. If you would like supplemental reading, there are several good options. We recommend the book by Kleinberg & Tardos [KT05]. The classic by Cormen, Leiserson, Rivest, and Stein [CLRS09] is a good encyclopedic reference. Also, the notes by Michael Levet will be helpful in working out solutions to problems in this class.

Many of the algorithms we study have minor variations, which may impact the final answer or intermediary steps. **The official version of the algorithms for the purpose of course questions will be those presented in the lecture notes (and not in supplemental texts).** On course assignments you are responsible for using the version of the algorithm presented in the lecture notes.

We also highly recommend MIT's Open Courseware notes [MIT11] and Jeff Errickson's notes [Eri] as supplemental resources. These are incredibly high-quality resources. In particular, Jeff Errickson has devoted considerable efforts to creating materials that are both accessible and useful for Algorithms students.

In contrast, there are a number of popular online resources can actually be harmful to use. Amongst the most popular of these is Geeks for Geeks. Many of their articles make subtle, but crucial errors (e.g., forgetting key base cases, incorrect arguments, etc.). These errors are not always apparent to students. Folks who use Geeks for Geeks and similar low-quality resources often find that their grades suffer.

If you find high-quality sources you like, please email them to the course staff at csci3104staff@colorado.edu and we will attempt to vet them and share them with your classmates via Canvas.

3 Course Structure and Grading

3.1 Grading Scheme

This course will use **Proficiency-Based** (sometimes called **Mastery-Based**) and **Standards-Based Grading**. For an introduction to this kind of grading, see, e. g., the first 6 minutes of this video.

1. There are 31 content standards (listed below).
2. Within each content standard, you will have at least 4 opportunities to demonstrate proficiency, spaced throughout the semester: a homework, a weekly quiz, a midterm (there are two midterms, not cumulative), and a final exam (cumulative). To have a standard count towards your grade, you must demonstrate proficiency **TWICE**.[1]
3. Your final grade is determined by how many standards your demonstrate proficiency in (twice), as follows:

A	A-	B+	B	B-	C+	C	C-	D+	D	D-	F
≥ 29	28	27	26	25	24	23	22	21	20	19	0–18

[1] This is not true for a few standards near the end of the class, for which there isn't enough time to provide four opportunities. For those standards, and those standards only—which will be clearly marked—you will have at least two opportunities to demonstrate proficiency, but need only do so once to have it count towards your grade. We will also remind you of this when it arises.

Roughly speaking: Grades in the **A** range indicate strong preparedness for subsequent courses in Theoretical Computer Science and Math. Grades in the **B** range indicate a strong understanding of the mechanics and a moderate understanding of the theoretical underpinnings. Grades in the **C** range indicate comfort with the mechanics, such as how to execute the algorithms or solve procedural problems.

3.2 Standards

1. Proof by induction.
2. (Greedy) Graph traversals: BFS/DFS.
3. (Greedy) Single source shortest path algorithm: Dijkstra's algorithm.
4. (Greedy) Examples where greedy algorithms fail.
5. (Greedy) Correctness: Exchange arguments.
6. (Greedy) Minimum-weight spanning trees: safe, useless, and undecided edges.
7. (Greedy) Minimum-weight spanning trees: Kruskal's and Prim's algorithm.
8. (Greedy) Huffman Encoding
9. (Greedy) Introduction to A* Search
10. (Greedy) Network flows: terminology.
11. (Greedy) Network Flows: Ford–Fulkerson Algorithm.

    Midterm 1 will cover standards 1–11, and occur in Week 8, the week of October 16–October 20

12. Asymptotics: Calculus I and II Techniques (Polynomials, Polylogarithmic Functions, L'Hopital's Rule, Exponentials, Ratio and Root Test)
13. Analyzing Code I: Independent nested loops
14. Analyzing Code II: Dependent nested loops
15. Analyzing Code III: Writing down recurrences
16. Solving Recurrences I: Unrolling
17. (Divide & Conquer) Basics & counterexamples
18. (Divide & Conquer) Mergesort
19. Solving Recurrences II: Tree method
20. (Divide & Conquer) Quicksort, modifications, and analysis
21. (Dynamic Programming) Identify the precise subproblems.
22. (Dynamic Programming) Write down recurrences
23. (Dynamic Programming) Using recurrence to solve.
24. (Dynamic Programming) Backtracking to find solutions.
25. (Dynamic Programming) Design dynamic programming algorithms.

    Midterm 2 will cover standards 12–25, and occur in Week 15, the week of December 4–December 8

26. (Data Structures) Hash tables, collisions
27. (Data Structures) Doubling lists & their amortized analysis

28. (Computational Complexity) Formulating Decision Problems

29. (Computational Complexity) Showing problems belong to P.

30. (Computational Complexity) Showing problems belong to NP.

31. (Computational Complexity) Structures and consequences of P vs NP.

    The Final Exam will cover all content standards 1–30, and will occur on Gradescope during Finals Week.

3.3 Grading

Submissions will be graded for correctness and clearly articulated reasoning. It is not enough to arrive at the correct answer. Supporting work and reasoning must also be included and **easy to follow**. That is, both your work (including attention to intermediary details) and the clarity in which it is presented will be graded.

Each problem will receive a score of $0, 1, 2, 3$, or 4 as follows.

- 4 (**Near Perfect**): The solution was near-perfect and clearly explained. This is almost a textbook caliber solution.
- 3 (**Demonstrated Proficiency**): The solution demonstrated proficiency, with perhaps minor (but not content-related) errors. While there may be room to improve the exposition, the solution was relatively easy to follow.
- 2 (**Demonstrated Progress, but Fell Short of Proficiency**): The solution demonstrated some measure of understanding, but had significant errors or was extremely difficult to understand.
- 1 (**Significant Errors or Misconceptions**): The solution demonstrated glaring misunderstandings, had significant or fatal errors.
- 0 (**No meaningful attempt**).

Note that scores of 3 are considered **full credit**, though scores of 3 and 4 count equally. We stress that solutions do not need to be perfect in order to demonstrate proficiency (this is more favorable than partial credit). Scores of $0, 1$, and 2 count equally (in that they do not contribute towards proficiency), but denote feedback as to how close one was to demonstrating proficiency. A score of 2 means that you likely have some handle on the content, but need to iron out some misconceptions or pay closer attention to details. A score of 1 is alarming, and you should attempt to fix any misunderstandings immediately. Please stop by student hours so we can help you!

Each homework will cover one or more standards. Problems will be organized by standard. Each question will receive a score and feedback. The scores on each question for a given standard will be aggregated into an Overall Standard Score, which will be your grade on that opportunity to demonstrate proficiency on that standard. In general, students who demonstrate proficiency on an overwhelming preponderance on the individual questions are likely to earn a score of 3 or 4 for their Overall Standard score. For more challenging standards, an occasional 2 on an individual question will not *necessarily* disqualify one from proficiency on the Overall Proficiency Score. Any scores of 0 or 1 on individual questions will disqualify one from scoring higher than a 2 on the Overall Proficiency Score.

There are no averages. Although we use numbers 0–4, these are merely shorthand codes for the above description. Multiple questions on a single standard on a single assignment do not merely get averaged into your grade for that standard on that assignment. If you demonstrate proficiency on one problem, but do not demonstrate proficiency on another problem (in the same assignment on the same standard), course staff will look into your answers more closely to gauge how you are understanding the material, and where your misunderstanding, if any, may be, and will assign grades accordingly. However, it is still the case that if you get a 3 or more on all questions on a standard, you will definitely get a 3 or more for that standard, and if you get 2 or lower on all questions on a standard, you will definitely get a 2 or lower for that standard.

The Overall Proficiency Score is neither the raw average nor the total number of points accrued. We illustrate this with the following example.

Example. Suppose that on HW2, there are three questions (Q1, Q2, and Q3) associated with Standard 3. Q1 asks students to execute Dijkstra's algorithm on an example graph, while Q2 and Q3 ask students to apply Dijkstra's algorithm to new situations. We note that Q3 is more challenging than Q2.

- Student A earned a 2 on Q1 due to a couple of careless mistakes, and 3's on both Q2 and Q3. They were awarded a 3 for their Standard 3 Overall Proficiency Score on HW2.
- Student B earned a 3 on Q1, a 3 on Q2, and a 2 on Q3 due to several fundemantal misunderstandings about Dijkstra's algorithm. For this reason, Student B was awarded a 2 for their Standard 3 Overall Proficiency Score on HW2.

3.4 Homework

Homework will be assigned regularly, with clearly posted deadlines, at least a week from when the homework is released. You are responsible for being aware of both the **dates** and **times** for these deadlines. Tentative dates/times can be found on the Google Calendar for assignments, quizzes, and exams. **These are subject to change,** but changes will be announced via Canvas announcements, email, and updated on the Google Calendar.

Late homework will not be accepted, except in Exceptional Circumstances. No late submissions will be accepted once solutions have been released. In exceptional circumstances, there are usually several other ways the course staff can help you get back on track, see Exceptional Circumstances, below.

Please submit your homework via Canvas.

- Each homework must be submitted as a single PDF file on **Gradescope**. The written homework must be **typed** using LaTeX. Diagrams (e. g., graphs, trees) may be hand-drawn and embedded in the LaTeXdocument as an image and **oriented so that we do not have to rotate our screens to grade your work.** Please note that **handwritten solutions or those prepared without LaTeXwill not be graded.** Similarly, **if we have to rotate our screens to grade your work, then it may not be graded.**
- Both your **name** and **student ID** must be included in the appropriate fields. You **must** include these on your assignment; otherwise, Gradescope may not be able to match your assignment to your submission. In this case, your submission will not be graded.
- *The first question on every homework will be an honor pledge.* Failure to indicate that you have upheld the honor code will result in your assignment not being graded.
- You are welcome to discuss the problems with your classmates, as well as reference outside resources. **Anything you submit must be in your own words and reflect your understanding of the material. You should be able to explain your solutions to the instructor, such as in an interview grading session.** If there are any questions about this, it is your responsibility to contact the instructor (and not the TAs) reasonably ahead of the submission deadline. **Copying from other sources (including your classmates) is an honor code violation.** You must **cite** any resource (other than the course text, instructor, TAs, or CAs) that you use. This includes any classmates with whom you collaborate. Failure to cite your sources will be treated as an **honor code violation.** See Section 4.12.
- Posting to online forums for help (e.g., Chegg, Discord, Reddit, StackExchange, etc.) is an **honor code violation.** See Section 4.12.

3.5 Quizzes & Midterms

There will be regular, online, written quizzes covering the given standards. They occur roughly once a week, beginning in week 4.

Each written quiz will have between 1-3 questions and will be timed for 45 minutes. The intent is that students will have 30 minutes (scaled for students with extra time accommodations) to take the quiz and 15 minutes to submit to Canvas. In practice, students are welcome to allocate the 45 minutes as they see fit. As we have allocated 15 minutes to prepare the quizzes for submission, late quizzes will **not** be accepted. If your internet goes out, you may take a picture and send a **legible** picture (in JPEG, PNG, or PDF formats) within the 45 minute window to one of the instructors (**not** the TAs).

We will provide a LaTeXtemplate for the quiz. You may either type your work using the LaTeX template, or you may handwrite your work and embed it as an image in the LaTeXtemplate. If you choose to handwrite your work, the image must be **legible** and **oriented so that we do not have to rotate our screens to grade your work.** If your work is illegible or we have to rotate our screens, your work will not be graded.

Quizzes are open-book and open-note, but are individual efforts. Consulting anyone who is not a member of the instructional staff about a quiz, which includes your classmates, tutors, and posting online (e.g., Chegg, Reddit, Discord, StackExchange, etc.) constitutes an **honor code violation.** See Section 4.12.

You are welcome to email csci3104staff@colorado.edu with clarification questions about the quiz. Please clearly mark such emails with the subject line "Quiz N question". Submitting such questions counts against your allotted time, and we cannot guarantee that we will respond to you in time, though we will make our best effort. Emailing us at the staff email address above helps ensure more timely responses.

Midterms are effectively the same as largely coordinated midcourse retakes. The purpose is to give you time to review and reflect upon the material covered, as well as to demonstrate proficiency. If you have demonstrated proficiency twice on a given standard, then there is no obligation to answer the corresponding questions on the Midterm. Questions on the Midterm will be clearly marked with what standard they correspond to.

3.6 Retake Tokens

As part of our Standards-Based Grading system, students will be able to make additional attempts (beyond the homework, quizzes, midterms, and final) towards standards of their choice, subject to a few guidelines. Please read through this section carefully before submitting a reassessment request.

- Students have **two retake tokens**. Each token is redeemable for a single opportunity to demonstrate proficiency on a **single standard once during Midterm 1** and again during **Midterm 2**. Multiple tokens cannot be used for the same standard. Please note that tokens are redeemable for reassessment attempts, not grades. That is, reassessing does not guarantee you a score of proficiency. Your work will still be graded for correctness. Tokens are non-refundable. Once your retake request is approved, your token has been consumed, regardless of whether you take the quiz. You may not use retakes to attempt standards we have not covered.

- In order to be eligible to use a retake token, students must write corrections and reflections for the relevant standard. The point is for students to iron out any misconceptions before attempting a retake. You may correct any HW, Quiz, or Exam problem on which you received below a 4. If you received below a 3 on any problem, you must correct one of those problems. It suffices to correct one problem.

- Submissions must be made through the following Google form: `https://forms.gle/Hbzx3Hin4F4hpVL88`

- You may submit corrections and reflections for relevant homework, quiz, and exam problems, provided you got below a 4 on the problem. Please clearly state the HW/Quiz/Exam and Problem upon which you are correcting/reflecting (e.g., PS6, Problem 1a). You may use a prior retake for your corrections and reflections.

- Please rework the **entire** problem, in addition to reflecting on your work. For reflections, please discuss points such as clear misunderstandings and technical gaps, as well as clearly explaining the techniques or concepts needed to correctly solve the problem. Alternatively, you could opt to write up a tutorial for how to solve the problem, while clearly explaining the key concepts and techniques. That is, address both the "how" and the "why." Writing a tutorial is a great way to correct/reflect if one fails to submit a problem set or quiz. You can also reflect on what you understand now that you didn't understand in your previous attempt, that led your previous attempt to not demonstrate your proficiency.

- If you failed to submit an assignment, you may still use that for your "corrections" and reflections by correctly working out the relevant problems and submitting reflections as described in the last bullet point.

- The corrections and reflections process may be iterative. If your submission has clear misconceptions or failed to thoroughly correct/reflect upon your solution, we will let you know about the issues and ask you to revise your reflection and resubmit through the Google form. the Google form. Note that your token is only considered redeemed once your corrections and reflections are accepted. This is intended to be a collaborative approach, to help ensure you better learn the content and succeed on future assessment opportunities.

- Reflections that make no attempt to meet these standards will be asked to revise/resubmit. Repeat offenders who consistently ignore our feedback over multiple iterations may lose one of their retake tokens without being permitted to reassess the standard in question. This is at the discretion of the instructional staff. This is not a first or even second offense penalty.

- Examples of reflections that don't meet the standards include, but are certainly not limited to:
  - "I missed the due date."
  - "I forgot to study."
  - "I didn't know the derivative of 4^n."
  - Reflections and corrections that exhibit minimal effort, such as when no effort is made to correct or reflect.
  - Disputing your original grade.

We expect to close the retaking form shortly after the Midterm 2 grades are returned. We will not accept new or revised Retake Requests once the form has been closed. **Please allow for a two week processing time.** As we near the end of the semester, we will work to expedite the processing time. Any student who submits a retake request before the form closes will have their request reviewed to allow sufficient time for any retaking to be done before the final retaking period.

4 Course Policies

4.1 Student Hours: Norms and Expectations

Student hours will be held online or in-person according to staff preference (see Canvas for locations, timing, and/or Zoom links). The purpose of student hours is to supplement lecture, recitation, and the associated readings with more individualized help. In order to get the most out of student hours, we recommend the following.

- Watch the lectures and read through the lecture notes. In particular, work through the provided examples. These materials are there to help you!

- Spend some time working the problems first. Try to identify specific approaches you have made, as well as identify where you are stuck. If you are spending more than 30 minutes on a single problem without making much progress, then we strongly encourage you to seek help in student hours!

- If you wish to discuss specific work in an online student hour, please take a picture or have it typed up so that you can more easily share your screen on Zoom. It can be hard for us to help you if your work is on paper and you are holding it up to the camera.

- Our goal is to help students improve their understanding and guide students toward solutions, as well as help students obtain momentum to keep working. In particular, we aim to help students arrive at the solutions on their own. It is completely normal to need time to digest a hint, and then come back to student hours with more questions! Learning Math is an iterative process - we encourage students to iterate!

- Please note that the course staff will not provide entire solutions in student hours, nor will they grade work ahead of the due date. If a member of the course staff states that they are unable to offer more hints without giving away a solution, please respect this. Students are welcome to bring any concerns about this to the instructors.

Student Hours vs. Email: The course staff are generally happy to help with questions via email to csci3104staff@colorado.edu. This email address is accessible by all course staff—instructors, graduate TAs, and undergraduate CAs—in order to help ensure timely responses. However, while email can be effective for answering brief questions, it is not usually a conducive medium for tutoring, and we recommend longer conversations be brought to student hours.

4.2 Regrade requests

All regrade requests must be made through this Google form the Google form (link also available through Canvas). There are two types of regrade requests: clerical (0 score given for attempted problem, or Gradescope doesn't match grade report) and conceptual (everything else). Clerical requests may be handled by any course grader; conceptual requests will be handled by a course instructor. Clearly include: what assignment, what question, and where you believe an error was made in the original grading. When you ask us for a conceptual regrade of one question on an assignment, instructors reserve the right to regrade the entire assignment, and your grade on those opportunities to demonstrate proficiency may go up or down.

We **require regrade requests within one week** of receiving your graded work.

We will respond to all regrade requests within **two weeks** of receiving a request through the form the form. Before Midterms and near the end of the semester, we will try to have faster turnaround times, but cannot guarantee it.

TAs and CAs are not empowered to change grades without instructor approval; please do not try to argue your grades with them. If you do try to argue your grade with a TA or CA, they should refer you to the instructors.

4.3 Solutions

Solutions for Homework, Quizzes, and Midterms will be posted on Canvas and in the Google Drive folder, generally within a couple days of the due dates. We do not anticipate providing solutions for retakes. Solutions will typically **not** be posted for the recitation activities.

4.4 Late Work

The course staff will evaluate more than 10,000 pages of homework this semester. For this to run smoothly, we need your help.

Problem sets are due Tuesdays at 8pm, as LaTeX'd solutions, in PDF, via the class Canvas page. (These deadlines may change, but if they do, it will be widely announced via Canvas announcements and the class email lists, as well as updating the class Google calendar.)

No credit will be given for assignments submitted in any other way (e.g., email) or after the deadline.

Note that missing the homework or quiz deadlines by a few minutes is not a valid reason for late work to be accepted. Homework due dates and times will be clearly posted, and students will have 15 minutes to submit their quizzes (on top of 30 minutes to take their quizzes). Please plan accordingly.

4.5 Exceptional Circumstances & Extensions

In the case of exceptional circumstances, e. g., significant illness or injury that preclude submitting an assignment on time, we have several possible remedies, depending on the situation. Examples of *unexceptional* circumstances include registering late, travel for job interviews or conferences or fun, forgetting the homework deadline, or simply not finishing on time.

To request an extension on an assignment, please use the form linked on the Canvas page. this form. Note: in such exceptional circumstances, it is often to your benefit to request an additional retake token rather than an extension. Taking an extension can result in one assignment spilling over into another, making you late on subsequent assignments. In contrast, retakes can be scheduled in coordination with course staff at a time that works for you. Our goal is for you to become proficient in the material and to demonstrate that to us, and in exceptional circumstances we will work with you to come up with a suitable arrangement.

Because there are so many opportunities to demonstrate proficiency already built in to the course structure, students have often found that short-term exceptional circumstances require no additional extensions or retakes. If you are concerned about your situation, however, please contact the course instructors and/or fill out the extension request form. extension request form.

4.6 Attendance

Attendance is not required for the lecture sections. However, it is expected that students are watching the lecture videos within 24 hours of when they are posted. Recitations will be designed based on the assumption that students have watched all of the lectures. Students who are enrolled for the in-person lecture sections may attend in-person or online. Students may attend any online lecture section. However, students enrolled for an in-person lecture section may only attend in-person for they section in which they are enrolled. Due to COVID restrictions, students enrolled for online sections **may not** attend in-person.

Students may only attend recitations for which they are enrolled. Students may **not** attend a different recitation without the instructor's consent. Occasionally, in emergency situations, students may be permitted to attend a different recitation section. However, they **must** discuss this in advance with the instructor.

4.7 Student Feedback

Student feedback regarding this course is encouraged and welcome. Those who wish to leave feedback anonymously may do so using this Google form: `https://forms.gle/wVSY83RucfgotWwm8`. Students are also welcome to reach out to the instructors via email or in student hours to discuss their concerns.

4.8 Classroom Behavior

Both students and faculty are responsible for maintaining an appropriate learning environment in all instructional settings, whether in person, remote or online. Those who fail to adhere to such behavioral standards may be subject to discipline. Professional courtesy and sensitivity are especially important with respect to individuals and topics dealing with race, color, national origin, sex, pregnancy, age, disability, creed, religion, sexual orientation, gender identity, gender expression, veteran status, political affiliation or political philosophy. For more information, see the policies on classroom behavior and the Student Conduct & Conflict Resolution policies.

4.9 Requirements for COVID-19

As a matter of public health and safety due to the pandemic, all members of the CU Boulder community and all visitors to campus must follow university, department and building requirements and all public health orders in place to reduce the risk of spreading infectious disease. Students who fail to adhere to these requirements will be asked to leave class, and students who do not leave class when asked or who refuse to comply with these requirements will be referred to Student Conduct and Conflict Resolution. For more information, see the policy on classroom behavior and the Student Code of Conduct. If you require accommodation because a disability prevents you from fulfilling these safety measures, please follow the steps in the "Accommodation for Disabilities" statement on this syllabus.

As of Aug. 13, 2021, CU Boulder has returned to requiring masks in classrooms and laboratories regardless of vaccination status. This requirement is a temporary precaution during the delta surge to supplement CU Boulder's COVID-19 vaccine requirement. Exemptions include individuals who cannot medically tolerate a face covering, as well as those who are hearing-impaired or otherwise disabled or who are communicating with someone who is hearing-impaired or otherwise disabled and where the ability to see the mouth is essential to communication. If you qualify for a mask-related accommodation, please follow the steps in the "Accommodation for Disabilities" statement on this syllabus. In addition, vaccinated instructional faculty who are engaged in an indoor instructional activity and are separated by at least 6 feet from the nearest person are exempt from wearing masks if they so choose.

Students who have tested positive for COVID-19, have symptoms of COVID-19, or have had close contact with someone who has tested positive for or had symptoms of COVID-19 must stay home. In this class, if you are sick or quarantined, please **do not come to any in-person class activities.** Lectures are live-streamed and recorded, and assignments can be made up or additional "free" retakes offered (not counting towards your three retake tokens). In many cases, because there are so many opportunities to demonstrate proficiency of a standard, students find that additional retake opportunities are not even needed. Rest, get better, and help keep others safe—we'll figure out how to help you catch up after.

4.10 Accommodation for Disabilities

If you qualify for accommodations because of a disability, please submit your accommodation letter from Disability Services to your faculty member in a timely manner so that your needs can be addressed. Disability Services determines accommodations based on documented disabilities in the academic environment. Information on requesting accommodations is located on the Disability Services website. Contact Disability Services at 303-492-8671 or dsinfo@colorado.edu for further assistance. If you have a temporary medical condition, see Temporary Medical Conditions on the Disability Services website.

4.11 Preferred Student Names and Pronouns

CU Boulder recognizes that students' legal information doesn't always align with how they identify. Students may update their preferred names and pronouns via the student portal; those preferred names and pronouns are listed on instructors' class rosters. In the absence of such updates, the name that appears on the class roster is the student's legal name.

4.12 Honor Code

All students enrolled in a University of Colorado Boulder course are responsible for knowing and adhering to the Honor Code academic integrity policy. Violations of the Honor Code may include, but are not limited to: plagiarism, cheating, fabrication, lying, bribery, threat, unauthorized access to academic materials, clicker fraud, submitting the same or similar work in more than one course without permission from all course instructors involved, and aiding academic dishonesty. All incidents of academic misconduct will be reported to the Honor Code (honor@colorado.edu); 303-492-5550). Students found responsible for violating the academic integrity policy will be subject to nonacademic sanctions from the Honor Code as well as academic sanctions from the faculty member. Additional information regarding the Honor Code academic integrity policy can be found on the Honor Code website.

Intellectual dishonesty or plagiarism of any form, at any level, will not be tolerated.

Discussing problems with other students is encouraged, but you must list your collaboration on the page where you give the solution. If you discussed it with 20 other people, then all 20 names should appear in your solution. If someone was particularly helpful, say so. Be generous; if you?re not sure whether someone should be included in your list of collaborators, include them. For discussions in class, in section, or in office hours, where collecting names is impractical, it?s okay to write something like ?discussions in class.? There is no penalty for discussing problems with other students.

Copying from any source in any way is strictly forbidden. This includes both the Web and other students (past or present). If you are unsure about whether something is permitted, please ask the instructors before the assignment is due.

Posting to online forums for help (e.g., Chegg, Discord, Reddit, StackExchange, etc.) is an **honor code violation.**

Write everything in your own words and cite all outside resources. You are strongly encouraged to use outside resources, but you must write your solutions yourself. We are not interested in seeing Wikipedia?s or anyone else?s solution. The only sources you are not required to cite are the textbook and lecture notes, and the prerequisite material.

There will be a zero-tolerance policy to violations of this policy. Violators will be removed from the class, given a grade of F, and reported to the University Honor Council, which may choose to impose additional penalties.

4.13 Sexual Misconduct, Discrimination, Harassment and/or Related Retaliation

The University of Colorado Boulder (CU Boulder) is committed to fostering an inclusive and welcoming learning, working, and living environment. CU Boulder will not tolerate acts of sexual misconduct (harassment, exploitation, and assault), intimate partner violence (dating or domestic violence), stalking, or protected-class discrimination or harassment by or against members of our community. Individuals who believe they have been subject to misconduct or retaliatory actions for reporting a concern should contact the Office of Institutional Equity and Compliance (OIEC) at 303-492-2127 or email cureport@colorado.edu. Information about OIEC, university policies, reporting options, and the campus resources can be found on the OIEC website.

Please know that faculty and graduate instructors have a responsibility to inform OIEC when made aware of incidents of sexual misconduct, dating and domestic violence, stalking, discrimination, harassment and/or related retaliation, to ensure that individuals impacted receive information about their rights, support resources, and reporting options.

4.14 Religious Holidays

Campus policy regarding religious observances requires that faculty make every effort to deal reasonably and fairly with all students who, because of religious obligations, have conflicts with scheduled exams, assignments or required attendance. In this class, we will make reasonable efforts to accommodate such needs if you notify the professors of their specific nature by the end of the 3rd week of class (Friday September 10, 2021).

See the campus policy regarding religious observances for full details.

References

[CLRS09] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, *Introduction to algorithms, third edition*, 3rd ed., The MIT Press, 2009.

[Eri] Jeff Erickson, *Algorithms homepage*, Available at `https://jeffe.cs.illinois.edu/teaching/algorithms/`.

[KT05] Jon Kleinberg and Eva Tardos, *Algorithm design*, Addison-Wesley Longman Publishing Co., Inc., USA, 2005.

[MIT11] *MIT open courseware algorithms lecture notes*, 2011, Available at `https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-006-introduction-to-algorithms-fall-2011/lecture-notes/`.