CSCI 0500: Data Structures, Algorithms, and Intractability

Figure 1: HERE TO HELP [https://xkcd.com/1831](https://xkcd.com/1831)

Instructor Information

Name: Philip Klein  
Office: CIT 503  
Email: klein@brown.edu  
Office Hours: Monday, Wednesday, and Friday, 4:30 – 5:00p

Prof. Klein would like every student to come to office hours at least once. The professor will try to put together groups of students to meet together with him. If you cannot make the scheduled office hours, please email Prof. Klein to arrange another time.

Class Information

Classroom: CIT 477 (Lubrano)  
Time: Monday, Wednesday, and Friday 3 – 4:20p  
Textbook-like resources: [Jeff Erickson’s *Algorithms* 1st Edition](https://algs4.cs.princeton.edu) and [Kent Quanrud’s excellent course notes](https://csci1090.sites.brown.edu)  
Each includes much content that will not be addressed in the course.

Prerequisites

A completed introductory program sequence (CSCI0200 or CSCI0190) along with some introduction to proof-writing and probability (CS0220 or equivalent with permission of the instructor).

Course Description

This course will cover the basics of how to design and analyze data structures and algorithms. We will develop algorithmic intuition through rigorous analysis of algorithmic correctness and performance. We will also study the theory of NP-completeness, which helps us understand which problems are computationally intractable.

The course website can be found at [https://cs.brown.edu/courses/csci0500/](https://cs.brown.edu/courses/csci0500/)
Course Objectives

- Develop your ability to think algorithmically.
- Ensure that you understand some well-established techniques, algorithms, and methods of analysis.
- Give you some technical tools to predict for a given computational problem what kind of performance is possible.
- Hone your ability to reason rigorously about computational problems and algorithms and quantities such as running time.

Topics

The course will address the following topics:

- Arithmetic and algebraic computation
- Sorting and selection
- Data structures (hash tables, priority queues, binary trees, union-find)
- Graph algorithms (topological sorting, biconnected and strongly connected components, shortest paths, minimum spanning trees)
- Dynamic programming
- Finite automata (briefly) and Turing machines
- NP-completeness and the Cook-Levin Theorem

The course will emphasize formal proofs of correctness and methods of analysis.

Course Structure

- Three times a week we will meet for a combination of lecture and workshop. In workshop, you will work (sometimes alone, sometimes in a small group) to solve a problem arising during the lecture and produce a short writeup.
- There is a weekly lab in which students will work in pairs to write, program, and test algorithms for their correctness and performance. At each lab session, students will be paired randomly.
- There will be weekly homework assignments.
- There will be two midterm exams and a final exam. The midterms are held in class on February 16 (the Friday before Long Weekend) and March 22 (the Friday before Spring Break). The final exam will be 9:00 am - 12:00 pm on May 8.

Estimated time requirements

- Lecture + Workshop: four hours a week
- Lab: two hours a week
- Homework: four hours a week
- Studying/review: three hours a week

Grading Policy

To pass the course, a student must score at least 50% in each of the categories: homework, workshop, labs, and exams. Assuming the student achieves that, their grade will be determined by the following weighted average.

1If there is someone in your lab session that you do not want to be paired with, email the professor.
<table>
<thead>
<tr>
<th>Assignment</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Homework</td>
<td>30%</td>
</tr>
<tr>
<td>Workshop</td>
<td>15%</td>
</tr>
<tr>
<td>Labs</td>
<td>15%</td>
</tr>
<tr>
<td>Midterm 1</td>
<td>10%</td>
</tr>
<tr>
<td>Midterm 2</td>
<td>10%</td>
</tr>
<tr>
<td>Final</td>
<td>20%</td>
</tr>
</tbody>
</table>

You will not be graded “on a curve.” Therefore you are not competing with your classmates.

Workshops are each graded out of four points, and only the top 85% scores will count toward your course average.

Every student has a budget of two late days which can be applied to homework (maximum of one per assignment). The lowest homework score will be dropped.

Labs should be completed during the assigned two-hour block. Students can make up any missed checkpoints with their lab TA at the beginning of the next lab for half-credit. The lowest lab score will be dropped.

**Collaboration Policy**

We allow (and encourage!) discussion of the material presented in class as well as discussion of concepts involved in homework assignments. Written work is another matter. *You must write up your homework solutions by yourself*. This ensures that you understand the material even if you collaborated on the problem. *You should not take notes away from collaboration sessions and should be by yourself when writing the solution.*

Rule of thumb: If you cannot independently reproduce what you hand in, don’t hand it in. We reserve the right to request that you explain your solution to a problem, so we can ensure that the solution was your own work.

For the topics we address in class, there are resources available (from books and from scientific journals, and from the Web) to help you understand. You are welcome to consult such resources to help you learn the material. Similarly, you can consult people (e.g. a tutor) to help you with understanding course material. However, you are forbidden from using resources from outside the class to address topics or problems arising in assignments (homework, workshops, labs, and exams). This includes AI tools such as ChatGPT, Bard, and Github Copilot; you must not consult such tools in the course of preparing solutions to assignments.

In the event that you inadvertently stumble upon information relevant to a solution to a problem, just close the site or book before reading details. If you feel you crossed the line of the policy, contact course staff and document your source. Depending on the extent, you might not receive credit for your solution, but a citation will protect you from being charged with violating the course collaboration policy.

Helping someone else in the class to violate the collaboration policy is also a violation of the collaboration policy. Do not provide any written resources to another student in the class that the other student could physically/digitally retain and use in writing up solutions.

During exams, you must not use any resources other than your mind (and pen and blank paper). You are not allowed to consult the Web, for example. For specific exams, the instructor might provide guidance specifying what notes you can bring with you.

**Diversity and Inclusion**

The course intends to provide a welcoming environment for all students. We especially welcome diverse ideas and perspectives during class discussions.

All members of the CS community, including faculty and staff, are expected to treat one another professionally. Toward this goal, TAs have undergone training in diversity and inclusion. However, despite our best efforts, we may accidentally slip up, so please feel free to speak to any member of the course staff with any concerns you have during the semester, and do not hesitate to contact Professor Klein directly. We will take your concerns very seriously.
SAS Accommodations

If you feel you have physical, psychological, or learning disabilities that could affect your performance in the course, we urge you to contact SEAS (https://www.brown.edu/campus-life/support/accessibility-services/). We will do whatever we can to support accommodations recommended by SEAS.

If you require SEAS accommodations, please let Professor Klein know at the beginning of the semester.