Introduction to Analysis of Algorithms

Fall 2023 Syllabus
Cornell Bowers CIS Department of Computer Science

General Information

- **Instructor:** Anke van Zuylen, Gates Hall 447, avz2@cornell.edu/cs4820-prof@cornell.edu.

  - Office hours: Tuesday 2pm-3pm, Wednesday 10:30am-11:30am in Gates 447.
  - You can make a 1-on-1 appointment with the instructor using Calendly. Appointments slots are limited and are meant for discussing topics you cannot discuss in public office hours (so not for homework questions!).

  - Email: To email the instructor or the course administrator about logistical matters, please use cs4820-prof@cornell.edu. Please do not use email to ask questions about the material, policies, exam dates, etc. You can submit questions like these to Ed Discussion, where everyone can view the reply as well (and don't forget to search whether a question has been asked already).

- **Course Administrator:** Sara Perkins, Gates 401, sep247@cornell.edu/cs4820-prof@cornell.edu.

- **TAs and TA Office hours:** Please see Canvas for their information and office hours schedule.

- **Course Website:** https://canvas.cornell.edu/courses/56996

  We will be using Canvas for announcements and lecture materials. You should be enrolled automatically into Canvas, but if not (for example, because your registration has not been processed yet), please contact Sara Perkins at cs4820-prof@cornell.edu to get added manually. All other websites we use (Gradescope for homework submission, Ed Discussion for Q&A, etc.) are linked from the course Canvas site; you should automatically get added to the rosters for these other sites if you access them through the Canvas link.

Credit Hours and Credit Hour Options

- 4.0 credits, Letter Grade or Student Option--Satisfactory (S)/Unsatisfactory (U).

- Graduate students need to register for CS5820 (and are graded differently, see Grading section below); undergraduate students need to register for CS4820.

- In addition to the scheduled lectures and accompanying homework assignments, students will have regular assignments for which they need to independently study how to

implement the algorithms discussed in the course using a programming language of their choice (typically Java or C++).

Prerequisites

The prerequisites for the course are having successfully completed all three of CS 2800, CS 2110, and CS 3110; if you received an A- or better in both CS 2800 and CS 2110, you are welcome to take CS4820 without prior completion of CS 3110.

Time and Location

Lectures are held MWF 9:05am-9:55am, in Uris Hall G01, for a total of 42 sessions.

Course Description

This course develops techniques used in the design and analysis of algorithms, with an emphasis on problems arising in computing applications. Example applications are drawn from systems and networks, artificial intelligence, computer vision, data mining, and computational biology. This course covers four major algorithm design techniques (greedy algorithms, divide-and-conquer, dynamic programming, and network flow), undecidability and NP-completeness, and algorithmic techniques for intractable problems (including identification of structured special cases, approximation algorithms, local search heuristics, and online algorithms).

Learning Objectives

On completing this course, students should be able to:

- Identify problems solvable with a **greedy algorithm**, design and prove the correctness of such an algorithm, and supply asymptotic running time for a variety of given algorithms.

- Recognize problems to which **divide and conquer** or **dynamic programming** approaches may apply, design algorithms with these approaches, and analyze their computational efficiency;

- Apply **randomization** to produce tractable algorithms for several specific computationally challenging problems;

- Reduce resource management as well as partition problems to **network flow or cut problems**, implement correct strategies for finding optimal flows/cuts, and understand the properties of these strategies;

- Recognize whether or not certain problems are computationally intractable (e.g., **NP-complete, uncomputable**), and use reductions to known problems to prove intractability;

- Use **approximation algorithms** to efficiently produce near-optimal solutions for intractable problems, and bound how close these algorithms are to being optimal;

- Use **online algorithms** to produce near-optimal solutions when only partial information about a problem is available, and bound how close these algorithms are to being optimal; and

- Recognize, implement, and understand the properties of several famous and important algorithms, including:

  - **Gale-Shapley method** for stable matchings,
  - **Prim's and Kruskal's algorithms** for finding minimum spanning trees,
  - **Bellman-Ford's algorithm** for finding shortest paths in a graph; and,
  - **Ford-Fulkerson's algorithm** for finding max flows in networks.

Course Materials

The textbook for the course is Algorithm Design by Jon Kleinberg and Eva Tardos (available at Cornell Store). Although this book was designed for this course, there will be topics covered in lecture that are not in the text and there will be topics in the text that are not covered in lecture. You are responsible for topics covered in lecture and for any assigned reading in the text. The following books are also useful references.

- T. Cormen, C. Leiserson, R. Rivest. *Introduction to Algorithms.*

- S. Dasgupta, C. Papadimitriou, and U. Vazirani. *Algorithms.*

- A. Aho, J. Hopcroft, J. Ullman. *The Design and Analysis of Computer Algorithms.*

- M. Garey and D. Johnson. *Computers and Intractability.*

- D. Kozen. *The Design and Analysis of Algorithms.*

Note on Prerequisites

We assume that everyone is familiar with the material in CS 2110, CS 3110, and CS 2800, and we will use it as necessary in CS 4820. This includes elementary data structures, probability (conditional probability, expectation, variance), sorting, and basic terminology involving graphs (including the concepts of depth-first search and breadth-first search), and coding (in Java, or one of the other languages supported – see Canvas >Modules >Resources >Programming Assignment Instructions). Some of these are reviewed in the text. The lectures and homework involve analyzing algorithms at a fairly mathematical level. A few of the homework exercises

consist of writing code in Java. We expect everyone to be **comfortable reading and writing proofs at the level of CS 2800, as well as writing code in Java**.

Grading

Your grade will be based on weekly homework, participation (based on responses to poll everywhere, and completion of the mid and end-of-semester course evaluation), two prelims, and one final exam. Each of these components will be given a weight in the following ranges:

- homework: 30%
- homework (advanced problems): 7.5% (for CS5820), 2.5% (for CS4820)
- participation: from 0% to 3%
- prelim 1: from 10% to 25%
- prelim 2: from 10% to 25%
- final exam (cumulative): from 25% to 40%

At the end of the semester, we will compute the best possible score for each individual student using weights in the indicated ranges. Typically, the median score is set as the cutoff for the grade of B+ (that is, 50% of the students in the course receive a B+ or higher).

Homework

Homework is an important part of the course. This course has weekly homework assignments. All homework assignments will be posted on Canvas. Most homework assignments will be due on Wednesdays at 11:59pm. Your lowest homework assignment will be dropped.

Homework and Advanced Problems

Most homework assignments will consist of three problems; in addition, some advanced problems will be assigned over the course of the semester. The advanced problems require students to synthesize different components of the course material. **Graduate students (i.e., those students enrolled in CS5820) are expected to complete all advanced problems**; for undergraduate students, completion of advanced problems is encouraged, but not required.

Typesetting

We will require problem sets to be typeset and submitted as a PDF. This requirement is for everyone's benefit. In general, we recommend that you first develop your solutions in draft form, and then write or type your solution in a concise way. Typesetting not only makes the last

step essential (instead of handing in solution in draft form), but it also makes it much easier for you to edit and improve your writeup, as well as easier for your TAs to read your proofs. It is up to you which tool you use; though we recommend LaTeX, tools like the Equation Editor in Microsoft Word can be surprisingly effective as an alternative.

For some proofs or writeups, it may be helpful to use a figure to explain your thinking more concisely. This is encouraged! Again, it is up to you how you want to include that in your writeup, whether it is a picture of a drawing in your notebook that you took with your phone or something you made digitally, as long as the figure was produced by you personally and is clear enough to see, it's a great idea to include it.

Late Submissions

Late submissions are accepted up to 24 hours after the deadline. You can submit **two** late assignments without penalty; each subsequent late submission incurs a 1.5-point deduction on your homework average (on a scale from 1 to 100).

The two "free" late submissions are meant to help you deal with disruptions to your schedule due to interviews, illness, etcetera, and **requests for additional late days will not be granted except in truly extraordinary circumstances**, such as longer-term illness or the death of a loved one. If such an emergency arises, contact the instructor as soon as possible.

Collaboration

In the real world of algorithms research, collaboration and conversation is an important part of how ideas get generated. So too in this course; we encourage you to discuss with your peers in the course to brainstorm ideas for how to get through homework. However, **your solution must be written up completely on your own; you are not allowed to share digital or written notes or images of your work in any form with each other**. Just like in research, your work must also include acknowledgements of all students with whom you collaborated. Both the physical or digital distribution of information about solutions and the failure to acknowledge collaborators are serious violations of academic integrity.

Admissible Resources

For the homework, **it is not admissible to use resources beyond course material and student discussions**. In particular, you may not use ChatGPT or other AI tools, Wikipedia, search the Web, or look at any textbook, other than the ones assigned/recommended in the course. Using such additional resources is a violation of academic integrity. If you feel the resources available to you are insufficient, talk to course staff or ask questions on Ed.

Advice for Success

Algorithms assignments can often require creative insights and complex proofs beyond what previous courses have required. Here are a few tips for succeeding in your writeups:

- **Start your assignments early**. Even if you are not writing anything down yet, looking over the problem set well in advance of the due date can ensure you have enough time to brainstorm possible solutions and to clear up confusion about how to interpret a problem. Creativity does not work well on a deadline.

- **Talk with classmates at a similar level about ideas**. As previously stated, while you cannot share physical or digital solutions of any kind to these problems, we actively encourage you to talk to classmates while you work through them. In particular, we recommend finding a group of students to meet with throughout the semester in advance of the deadline to talk about ideas. For best results, make sure those students are at the same level of understanding of the material as you; talking through your ideas with colleagues with a similar level of understanding will make talking through ideas with each other easier and more equitable, and is more likely to leave you prepared for course exams.

- **Ask questions in class, in office hours, and on Ed Discussions**. The material in this class moves quickly and is often cumulative. If you find yourself scratching your head after a lecture, even after consulting the textbook and course notes, you are certainly not alone, and it is better to seek help than to wait until you are more confused.

- **Do not panic if you are stuck and confused**. It is normal to struggle, and to be stuck sometimes and have no idea what to do! Struggling with the material helps you get a deeper understanding, so rather than panic, remind yourself you are "doing it right" whenever you spend time feeling stuck and confused. Try to come up with examples to guide your thinking, and talk to your homework partners or the course staff if you are not able to get yourself unstuck on your own.

Exams

There will be two prelims and a final exam. The prelim exams are scheduled for the following two dates: Thursday, October 5th, and Thursday, November 16th. The date for the final exam is still to be determined by the university. The exams will be held in-person.

The goal of the exams is to reinforce the material, by giving you an incentive to review the material, and for us to check that you understand the material to a sufficient extent to be able to use it in future courses. You can expect questions on the exam to be similar to the homework problems, so a good understanding of the key ideas used in the homework problems is

necessary to do well, however, exam questions are constructed to take much less time to solve than the homework problems.

Exam policies

- Exams are closed-book and closed-notes.
- Review materials that include questions from previous years' exams will be provided a week before the exam.
- Communicating with anyone or accessing unauthorized resources during the exam is considered a violation of academic integrity. Not reporting requests of other students who ask for help is also a violation.
- Violations of academic integrity during a prelim or final exam result in an F in the course.

Course Management

We understand that our members represent a rich variety of backgrounds and perspectives. Cornell University is committed to providing an atmosphere for learning that respects diversity. We expect students to communicate in a respectful manner with the instructors, course staff, and fellow students, in a way the honors the unique experiences, values, and beliefs represented by different members of our community.

Academic Integrity

Each student is expected to abide by the Cornell University Code of Integrity, available at https://theuniversityfaculty.cornell.edu/dean/academic-integrity/code-of-academic-integrity/. You are allowed to collaborate on the homework to the extent of formulating ideas as a group. However, **you are required to write up (and understand) the homework on your own, and you should acknowledge the names of the students with whom you collaborated.**

Violations of the homework collaboration policy or accessing homework solutions found online result in a penalty of up to negative 50% (-50%) for the assignment. Violations of academic integrity on a prelim or the final exam result in an F for the course.

Course Material Copyright

Course materials posted on Canvas, Gradescope, or Ed Discussion are intellectual property belonging to the author. Students are not allowed to buy or sell any course materials without the instructor's permission. Such unauthorized behavior constitutes academic misconduct.

Inclusiveness

You should expect and demand to be treated by your classmates and the course staff with respect. You belong here, and we are here to help you learn and enjoy this course. If any incident occurs that challenges this commitment to a supportive and inclusive environment, please let the instructor or a TA know so that we can address the issue. We are personally committed to this and subscribe to the Computer Science Department's Values of Inclusion.

What to do if you get sick

If you are sick, your priority should be getting better: get rest, drink fluids, go to see a doctor if needed, etc. If you feel well enough to come to lecture, please wear a mask so you do not infect others. Zoom recordings of lectures are not available for absences, including absences due to illness. You are expected to keep up with course material by working with a peer in the course and/or accessing the lecture notes on canvas. Please note that a single lecture counts for at most 0.07 (out of 100) towards your final grade, so absences because of a short illness will not affect your grade. You can use a "free" late day to hand in the homework a day later, but the weekly schedule of the homework in this course means we cannot give longer extensions.

Getting support

We urge you to talk to the instructor or any of the TAs if you have any concerns about your learning or progress in the course, if you are sick for more than a week, or if you have other personal difficulties. There is a variety of campus resources are available to you: https://mentalhealth.cornell.edu/get-support/support-students. In case you experience major unexpected events (for instance, being hospitalized unexpectedly, death of a close family member), please set up a meeting using Calendly or send an email, and we will figure out how to handle the situation together. We cannot accommodate requests for exceptions for minor issues (leaving early for a break, forgetting to submit a homework assignment, for example) — the course policies and grading scheme have been set up to take care of this.

Accommodations

This course complies with the university policies and equal access laws, and we provide accommodations for disability, religious observance, Title IX, varsity athletes, medical emergencies, and military service. Requests for academic accommodations should be made during the add/drop period, **no later than September 6**, except for unusual circumstances, so that arrangements can be made as soon as possible. Testing-related academic accommodations are organized by SDS; please carefully read the next subsection for details. If you require accommodations that are not testing related, please email cs4820-prof@cornell.edu to discuss the logistics of your accommodation(s).

Exam Accommodations

This course is participating in the SDS Alternative Testing Program (ATP) for the Fall 2023 semester. All exams will be centrally managed and supported by the ATP Testing Coordinator in the Office of Student Disability Services (SDS) and accessible in your SDS student portal. Exam logistics will be communicated to you via email from sds-testing@cornell.edu and accessible in your SDS student portal. The ATP will automatically send an email with the exam date, time, location 10 days before the exam, and a reminder email 48 hours prior to the exam date. The ATP will not be involved in the logistics for any make-up exams. If you have a conflict with your scheduled accommodated exam, please email cs4820-prof@cornell.edu, not the ATP.