Introduction to Analysis of Algorithms

CS 4820/5820 Fall 2022 Syllabus

General Information

- **Instructor:** Anke van Zuylen, Gates Hall 447, avz2@cornell.edu.
  - Office hours: Tuesday 5pm-6pm (potentially online; check Ed for weekly update), Wednesday 10:30am-11:30am (in-person).
  - You can make an appointment with the instructor using the Calendly 1-on-1 link from Canvas. Appointments slots are limited, and are meant for discussing topics you cannot discuss in public office hours (so not for homework questions!)
- **TAs and TA Office hours:** See schedule on Canvas.
- **Lectures:** MWF 9:05am-9:55am, Uris Hall G01.
- **Website:** `https://canvas.cornell.edu/courses/45045`

  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 Lacy Jordaens at lsl92@cornell.edu to get added manually. All other websites we use (Gradescope for homework submission, Ed Discussions 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.

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), computability theory focusing on undecidability, computational complexity focusing on NP-completeness, and algorithmic techniques for intractable problems, including identification of structured special cases, approximation algorithms, and local search heuristics. This course continues to build on work in previous courses on proofwriting and asymptotic runtime analysis of 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 intractible (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
- Be able to 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 Material

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.*

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 should be fine taking CS 4820 without prior completion of CS 3110. 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 the analysis of 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 based on weekly homework, participation (based on responses to polleverywhere, participation on Ed, and completion of the 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 (regular problems): 30%
- homework (challenge problems): 2.5% (CS4820), 7.5% (CS5820)
- 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.

Homework

Homework is an important part of the course. We will have weekly homework assignments. All homework assignments will be posted on Canvas. Most homework assignments will be due on Wednesday at 11:59pm.

Regular and challenge problems

Most homework assignments will consist of three (regular) problems; in addition, it will have one challenge problem. The purpose of the challenge problems is to allow students to demonstrate knowledge, understanding, and creativity at the "excellent" level, see `https://www.cs.cornell.edu/andru/grading-policy.html`. Challenge problems are worth 2.5% of the total course grade for undergraduate students and 7.5% for graduate students (i.e., those students enrolled in CS5820).

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), 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; subsequent late assignments will have a penalty of 5% for being up to one hour late, and 25% for being 1-24 hours late.

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 Wikipedia, or 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 aren't 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 doesn't 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're certainly not alone, and it's better to seek help then than to wait until you are more confused.
- **Don't panic if you're 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 (but do talk to your homework partners or the course staff if you are not able to get yourself unstuck after thinking (making up examples, thinking about what should happen on those, etc.) on your own).

Exams

There will be two prelims and a final exam. The prelim exams are scheduled for the following two dates: Tuesday October 4, 7:30pm and Tuesday November 15, 7:30pm. The date for the final exam is to be determined. 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 at a sufficient level 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 any person or accessing unauthorized resources during the exam is considered to be 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 Conduct

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.

COVID 19 Symptoms

If you have symptoms of COVID 19, take a test immediately upon noticing symptoms (antigen tests are freely available and you should always have one ready at home). If you test positive, stay home for the next 5 days, and wear a mask in class through the 10th day after symptoms begin. If you test negative, you can come to class, but please wear a mask while you have symptoms.

Zoom recordings of lectures are not available for absences, including absences due to illness. For any 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. 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 further extensions.

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.

Course Material Copyright

Course materials posted on Canvas, gradescope, or Ed Discussions are intellectual property belonging to the author. Students are not permitted to buy or sell any course materials without the express permission of the instructor. 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.

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 first three weeks of the semester, except for unusual circumstances, so arrangements can be made as soon as possible. Students are encouraged to register with Student Disability Services, as we may require verification of eligibility to provide appropriate accommodations.