CSC473 (Winter 2023) Advanced Algorithms

Overview

Welcome to the course webpage for the Winter 2023 term of CSC473, Advanced Algorithms. Here is the course content description:

> Advanced algorithm design techniques, with emphasis on the role that geometry, approximation, randomization, and parallelism play in modern algorithms. Examples will be drawn from linear programming; randomized algorithms; streaming algorithms and parallel algorithms.

This is a *theoretical* and *advanced* course. While we will cover algorithmic techniques useful in practice, our focus will be on proofs, theoretical analysis, and creative problem solving. Mathematical maturity, and a strong background in probability theory, linear algebra, data structures, and algorithm design are all essential.

Prerequisites for the course:

- CSC373
- MAT221H1/MAT223H1/MAT240H1

Check this website and Piazza frequently to make sure you receive any course announcements. Check the **Lectures (https://q.utoronto.ca/courses/291260/pages/lectures)** page for the required reading. Also keep an eye on the Announcements tab in Quercus.

Where and When

Type	Lecture	Tutorial
Room	GB 244	GB 244
Time	Wed and Fri 12pm - 1pm	Mon 12pm - 1pm

Contact information

Instructor	Aleksandar Nikolov
Email	anikolov -at- cs.toronto.edu
Office	Sandford Fleming 2301B
Office Hours:	Fridays 1-3pm, or by appointment

Prof. Nikolov will attempt to respond to legitimate email inquiries from students within 48 hours. Please include "CSC473" in the subject line of the email.

Piazza

The link to sign up for our Piazza forum is **https://piazza.com/utoronto.ca/winter2023/csc473 (https://piazza.com/utoronto.ca/winter2023/csc473)**. You can also access Piazza from within Quercus.

Piazza is a third-party software. It will be used in this class strictly as a discussion board. All announcements will be made on Quercus. When posting, abide by the academic integrity policy. In particular, **do not post solutions to homework problems**. Make sure to read the Piazza terms of use before signing up, and if you have any concerns, contact the instructor directly. If you decide to participate in Piazza, only provide content that you are comfortable sharing under the terms of the Privacy Policy and Terms of Use.
When using Piazza, be respectful to your instructors and fellow students. Offensive language and threatening behavior will not be tolerated. Keep in mind that when posting "anonymously", you are anonymous only to other students, but not to the instructors.

Grading Scheme

Your mark for the class will be based on the following components:

- Homework assignments: 32%
- Bi-weekly quizzes: 6%
- Midterm Exam: 20%
- Final Exam: 42%

A *quiz* will be made available on Quercus every other week (except Reading Week) on Friday, and will be due by midnight on the following Thursday. Each quiz is worth 1% of the grade, and consists of two basic questions about the lecture material and the reading assignment for the week. Quizzes should be completed **individually**.

There will be four *homeworks*, each worth 8%, to be done in groups of at most two. See **link (https://q.utoronto.ca/courses/291260/pages/homeworks)** .

The *midterm exam* will be one hour long, and will take place on February 27, in the usual tutorial time slot and room. It will cover all the material in the first six weeks of the course.

The will be a comprehensive *final exam*. **You need to score at least 40% on the final exam to pass the course.**

Academic Integrity

Every student must abide by the **University of Toronto academic integrity policy (https://www.academicintegrity.utoronto.ca/)** , and the **Code of Student Conduct (http://www.governingcouncil.utoronto.ca/Assets/Governing+Council+Digital+Assets/Policies/PDF/ppjul012002.pdf)** . Academic misconduct is taken **very seriously**! See the **Homeworks (https://q.utoronto.ca/courses/291260/pages/homeworks)** page for information about what resources you are allowed to use when working on your assignments.

Lectures

Navigation	**Home page (https://q.utoronto.ca/courses/291260/pages/csc473-winter-2023-advanced-algorithms)**	**Lectures (https://q.utoronto.ca/courses/291260/pages/lectures)**	**Homeworks (https://q.utoronto.ca/courses/291260/pa**

General Info

The lectures allow us to explain new material, how it relates to the rest of the course (and what you've learned in other courses), and to show examples of applying the material. Lecture notes that go into more details will be made available on this page.

Students often learn a lot from working with one another. You are encouraged to meet with other students taking the class for this purpose.

Lectures roughly fall into three main topics: randomized algorithms; linear programming and primal-dual algorithms; approximation algorithms. There is, however, significant overlap between the topics: for example, some of the approximation algorithms we will study are randomized, and many approximation algorithms use linear programming or the primal-dual method.

Please review these **Probability Theory (https://q.utoronto.ca/courses/291260/files/24112143/preview)** **(https://q.utoronto.ca/courses/291260/files/24112143/download?download_frd=1)** lecture notes at the beginning of class, and try the exercises. Understanding this material is essential for success in the courtse

Tentative Schedule of Lectures

Schedule of lectures

Week	Topic	Readings	Tutorials
Week 1: Jan 9 – 15	Monte Carlo Algorithms: Global Min-Cut	From "Algorithm Design" by Kleinberg and Tardos: **Min Cut (https://q.utoronto.ca/courses/291260/files/24112144/preview)** **(https://q.utoronto.ca/courses/291260/files/24112144/download?download_frd=1)** **Karger-Stein Algorithm (https://q.utoronto.ca/courses/291260/files/24112156/preview)** **(https://q.utoronto.ca/courses/291260/files/24112156/download?download_frd=1)** Slides: **MinCut (https://q.utoronto.ca/courses/291260/files/24269116?wrap=1)** **(https://q.utoronto.ca/courses/291260/files/24269116/download?download_frd=1)** **(https://q.utoronto.ca/courses/291260/files/24112156/preview)**	No tutorial
Week 2: Jan 16–22	Las Vegas Algorithms: Closest Pair of Points Approximate Near Neighbours	From "Algorithm Design" by Kleinberg and Tardos: **Closest Pair of Point (https://q.utoronto.ca/courses/291260/files/24112142/preview)** **(https://q.utoronto.ca/courses/291260/files/24112142/download?download_frd=1)** **Approximate Near Neighbour Search (https://q.utoronto.ca/courses/291260/files/24112158/preview)** **(https://q.utoronto.ca/courses/291260/files/24112158/download?download_frd=1)**	**(https://q.utoronto.ca/courses/291260/files/24112159/preview)** **Contraction (https://q.utoronto.ca/courses/291260/files/24112155/download?wrap=1)** **(https://q.utoronto.ca/courses/291260/files/24112155/download?download_frd=1)**
Week 3: Jan 23–29	Approximate Near Neighbours	**Approximate Near Neighbour Search (https://q.utoronto.ca/courses/291260/files/24112158/preview)**	**Contraction Exercises (https://q.utoronto.ca/courses/291260/files/24112159/preview)** **(https://q.utoronto.ca/courses/291260/files/24112159/download?**

		(https://q.utoronto.ca/courses/291260/files/24112158/download?download_frd=1)	download_frd=1) [LSH exercises (https://q.utoronto.ca/courses/291260/files/24112161/download?wrap=1)] (https://q.utoronto.ca/courses/291260/files/24112161/download?download_frd=1)
Week 4: Jan 30–Feb 5	Approximate Near Neighbours Streaming Algorithms	[Approximate Near Neighbour Search (https://q.utoronto.ca/courses/291260/files/24112158/preview)] (https://q.utoronto.ca/courses/291260/files/24112158/download?download_frd=1) [Streaming Algorithms (https://q.utoronto.ca/courses/291260/files/24112164/preview)] (https://q.utoronto.ca/courses/291260/files/24112164/download?download_frd=1)	[Variance and Chebyshev (https://q.utoronto.ca/courses/291260/files/24112146/preview)] (https://q.utoronto.ca/courses/291260/files/24112146/download?download_frd=1)
Week 5: Feb 6–12	Streaming Algorithms	[Streaming Algorithms (https://q.utoronto.ca/courses/291260/files/24112164/preview)] (https://q.utoronto.ca/courses/291260/files/24112164/download?download_frd=1)	[Streaming Exercises (https://q.utoronto.ca/courses/291260/files/24112165/download?wrap=1)] (https://q.utoronto.ca/courses/291260/files/24112165/download?download_frd=1)
Week 6: Feb 13–19 **(Reading week: Feb 20-26)**	Linear Programming	[Linear Programming (https://q.utoronto.ca/courses/291260/files/24112178/preview)] (https://q.utoronto.ca/courses/291260/files/24112178/download?download_frd=1) [LP Duality Cheatsheet (https://q.utoronto.ca/courses/291260/files/24112168?wrap=1)] (https://q.utoronto.ca/courses/291260/files/24112168/download?download_frd=1)	[LP duality exercises (https://q.utoronto.ca/courses/291260/files/24112162?wrap=1)] (https://q.utoronto.ca/courses/291260/files/24112162/download?download_frd=1)
Week 7: Feb 27–Mar 5	Linear Programming Matchings	[Linear Programming (https://q.utoronto.ca/courses/291260/files/24112178/preview)] (https://q.utoronto.ca/courses/291260/files/24112178/download?download_frd=1) Goemans's [matching lecture notes (http://math.mit.edu/~goemans/18433S09/matching-notes.pdf)].	Midterm in tutorial slot
Week 8: Mar 6–12	Matchings and the Hungarian Algorithm	Goemans's [matchings lecture notes (http://math.mit.edu/~goemans/18433S09/matching-notes.pdf)]. [(Lectures/Rounding.pdf)]	[Matchings exercises (https://q.utoronto.ca/courses/291260/files/24112173?wrap=1)] (https://q.utoronto.ca/courses/291260/files/24112173/download?download_frd=1)
Week 9: Mar 13–19	Rounding Algorithms Derandomization	Sections 1.7, 5.1, 5.2 of the [Williamson and Shmoys book (http://www.designofapproxalgs.com/)]. All of Chapter 1 of the book is recommended.	
Week 10: Mar 20–26	Chernoff bounds	Section 5.2, 5.10-5.12 of the [Williamson and Shmoys book (http://www.designofapproxalgs.com/)]. Chapter 4 of the [Motwani-Raghavan (http://go.utlib.ca/cat/8230181)] book has more information on tail inequalities.	

Week 11: Mar 27–Apr 2	Chernoff bounds Primal-dual approximation algorithms	Section 5.2, 5.10-5.12 of the **Williamson and Shmoys book (http://www.designofapproxalgs.com/)**. Section 1.7 and 7.1-7.2 of the **Williamson and Shmoys book (http://www.designofapproxalgs.com/)**.	
Week 12: Apr 3–9	Primal-dual approximation algorithms	Section 1.7 and 7.1-7.2 of the **Williamson and Shmoys book (http://www.designofapproxalgs.com/)**.	

Suggested Exercises

In addition to the exercises below, always attempt the exercises in the posted readings and lecture notes above.

- Suggested exercises for Min Cut:
  - From Motwani and Raghavan's Randomized Algorithms (available through the U of T library at **http://go.utlib.ca/cat/8230181 (http://go.utlib.ca/cat/8230181)**) try the exercises 10.13 - 10.15 after Chapter 10.
  - From Jeff Erickson's lecture notes, try the exercises after **Lecture 13 (http://jeffe.cs.illinois.edu/teaching/algorithms/notes/08-mincut.pdf)**.
  - After doing exercise 1 from Erickson's notes, try to give an O(m) time algorithm to execute the contraction algorithm, without using the Klein-Karger-Tarjan MST algorithm.

Learning Objectives

By the end of this course, you should be able to:

- Distinguish between Monte Carlo and Las Vegas algorithms.
- Design and analyze simple randomized algorithms.
- Have a deep understanding of Karger's Contraction algorithm, including its analysis and how to implement it efficiently.
- Design and analyze locality sensitive hash functions for different distance metrics, and use them for approximate near neighbour search and related problems.
- Use sampling to estimate sizes of sets, and apply sampling in streaming and other algorithms. Use variance computation and Chebyshev's inequality to analyze sampling algorithms.
- Define basic terms in polyhedral geometry, like vertex, face, facet, polytope.
- Model optimization problems as linear programs, and derive the dual of a linear program.
- State the complementary slackness theorem, and understand the analysis of primal-dual algorithms, like the Hungarian algorithm.
- Be able to apply Chernoff's bound to analyze how close a sum of independent random variables is to its expectation. Be able to apply the bound to analyze simple randomized algorithms.

Further Reading

Below you can find some surveys and research articles related to the topics in this course. The research articles may be challenging for you, and this is normal. As a start, you can just read the introduction and try to understand the statements of results.

- *Matchings*: **Paths, Trees and Flowers (https://doi.org/10.4153/CJM-1965-045-4)**, Edmonds's paper which gave the first efficient algorithm for maximum cardinality matching in general graphs. Check the section which argues why "efficient" can be abstracted as "polynomial time". If you want to learn the algorithm, a better place to start may be these **lecture notes (http://math.mit.edu/~goemans/18453S17/matching-nonbip-notes.pdf)**.
- *Streaming Algorithms*: **old survey (http://algo.research.googlepages.com/eight.ps)**; **survey on graph streaming algorithms (https://people.cs.umass.edu/~mcgregor/papers/13-graphsurvey.pdf)**.
- *Minimum Cut*: **practical algorithms (https://arxiv.org/abs/1708.06127)**; **deterministic algorithm (https://arxiv.org/abs/1411.5123)**. **(https://homepages.cwi.nl/~lex/files/histtrpclean.pdf)**

Homeworks

Navigation	[Home page](https://q.utoronto.ca/courses/291260/pages/csc473-winter-2023-advanced-algorithms) (https://q.utoronto.ca/courses/291260/pages/csc473-winter-2023-advanced-algorithms)	[Lectures](https://q.utoronto.ca/courses/291260/pages/lectures) (https://q.utoronto.ca/courses/291260/pages/lectures)	[Homeworks](https://q.utoronto.ca/courses/291260/pa) (https://q.utoronto.ca/courses/291260/pa

Overview

The course has 4 *group assignments* to be completed in groups of maximum two people each, and submitted on MarkUs. Additionally, a short quiz will be released on Quercus every other week on Friday, and due by the following Thursday. Quizzes should be completed *individually.* Quizzes check basic understanding of the material covered in the lectures and the assigned readings, while group assignments check that you can use the new concepts and techniques creatively to solve algorithmic problems.

Each assignment is due **by midnight** on its due date.

Assignment	Date out	Date due	Problems/Solutions
Assignment 1	Jan 13	Jan 27	
Assignment 2	Feb 3	Feb 17	
Assignment 3	Mar 3	Mar 17	
Assignment 4	Mar 24	Apr 10	

Working in Groups

A *group assignment* is to be done by at most two people. For these assignments you are strongly encouraged to work with a partner, rather than work alone. You and your partner should discuss the questions with one another, and come up with solutions *together*, but you may **not** discuss them with other students. For each problem, one student in the group is responsible for writing the solution, and the other student is responsible for proof-reading and revising it. The first page of your submission **must** list the name, student ID, and UTOR email address of each group member, and also indicate, for each problem, who wrote the solution, and who revised it. The purpose of these rules is to ensure that *each student* fully understands the solution of every problem in the group submission.

If you would like to work with someone but you don't know anybody who could be your partner, simply post a "Search for Teammates" message in Piazza. Also, make an effort to speak with your classmates during lectures and tutorials - you may find that there are many others in the same situation as yours.

Academic Integrity

When working on assignments, you are not allowed to consult other books, solution manuals, or solutions to assigned problems or similar problems on the Internet. You should **not** discuss homework solutions with anyone other than the professor, the TA, and your partner (if working on a group assignment).

Failure to comply with these guidelines is a serious academic offense.
If you have any questions about this policy, make sure you ask the professor or the TA. More information about why plagiarism is bad and what happens to cheaters can be found at http://www.cs.toronto.edu/~fpitt/documents/plagiarism.html (http://www.cs.toronto.edu/~fpitt/documents/plagiarism.html) .

Lateness Policy

No late *quiz answers* will be accepted. Every student has **one** grace credit, which allows them to be late on one *group assignment* for up to 24 hours. After the credit is used, no other late submission from the same student will be accepted for the remainder of the course. If you are working in a group, then the credit is taken from both members of the group, and no other late assignments will be accepted from either group member for the remainder of the course.

Special Consideration

Please contact your instructor as soon as possible in case you are unable to complete an assignment.

Remarking Requests

Remarking requests will be accepted up to one week after the due date of the homework assignment. A remarking request can be used to alert us to possible mistakes in the grading of an assignment, but *not to question the marking scheme of the assignment.*

Submission Instructions

Quizzes are to be completed on Quercus.

Group assignment submissions will be done using the *MarkUs* system. (A link to our MarkUs instance will be posted here soon.) All group assignments should be **typed and not handwritten.**

To submit as a group (**only for group assignments**), one of you needs to *invite* the other to be their partner, and then the other student needs to accept the invitation. To invite one a partner, navigate to the appropriate Assignment page, find "Group Information", and click on "Invite". You will be prompted for the other student's CDF user name. To accept an invitation, find "Group Information" on the assignment page, find the invitation listed there, and click on "Join". You should do this **before the deadline** even if you are planning to use your grace credit or have been granted an extension.

Once you have submitted, click on the file's name to check that you have submitted the correct version-**and that it is in PDF**.

Remember to put the name, student ID, and UTOR email address of the group member who wrote the solution, and also the name, student ID, and the UTOR email address of the group member who proof-read and revised it on the first page of the submission.

File Formats and LaTeX resources

You are encouraged to use LaTeX to typeset your homework solutions (see below for links to LaTeX resources). However, the use of LaTeX is not required - what matters is that your submissions all be in PDF and typed. **Scans of hand-written solutions will not be accepted!**

LaTeX resources LaTeX is a general-purpose typesetting system that makes it easy to generate high-quality documents, particularly when formatting mathematical formulae. In addition, Piazza supports typesetting equations with LaTeX syntax (by enclosing the equation in double dollar signs, for example `$$e^{2\pi i} - 1 = 0$$`). Here are some links to get you started.

- **TeXworks (http://www.tug.org/texworks/)**, a cross-platform LaTeX front-end.
- The **LaTeX Wikibook (http://en.wikibooks.org/wiki/LaTeX)**.
- Additional **LaTeX Documentation (http://www.latex-project.org/guides/)**, from the home page of **the LaTeX Project (http://www.latex-project.org/)**.
- **Overleaf (https://www.overleaf.com/)** allows you to create and collaboratively edit a LaTeX without having to install LaTeX on your machine. It does require an internet connection, however.