CSC265 (Fall 2023) Enriched Data Structures and Analysis

Navigation	**Home page** (https://q.utoronto.ca/courses/314895/pages/csc265-fall-2023-enriched-data-structures-and-analysis)	**Lectures** (https://q.utoronto.ca/courses/314895/pa

Overview

Welcome to the course webpage for the Fall 2023 term of CSC265, Enriched Data Structures and Analysis. Here is the course content description:

> Algorithm analysis: worst-case, average-case, and amortized complexity. Standard abstract data types, such as graphs, dictionaries, priority queues, and disjoint sets. A variety of data structures for implementing these abstract data types, such as balanced search trees, hashing, heaps, and disjoint forests. Design, implementation, and comparison of data structures. Introduction to lower bounds.

This course is an "enriched" version of CSC263, Data Structures and Analysis. While we cover roughly the same topics, we will go at a faster pace, in greater depth and with more rigour, and with more challenging assignments. Greater emphasis will be placed on proofs, theoretical analysis, and creative problem-solving. Certain topics briefly mentioned in CSC263 may be covered in more detail in this course, and some additional topics may also be covered.

Check this website and Piazza frequently to make sure you receive any course announcements. Check the **Lectures** (https://q.utoronto.ca/courses/314895/pages/lectures) page for the required reading. Also keep an eye on the Announcements tab in Quercus.

Where and When

Type	Lecture	Tutorial
Room	ES B142	ES B142
Time	Mon and Wed 3pm - 4pm	Fri 3pm - 4pm

Contact information

Instructor	Aleksandar Nikolov

Email	anikolov -at- cs.toronto.edu
Office	Sandford Fleming 2301B
Office Hours:	Wednesdays 12-2pm, or by appointment

Prof. Nikolov will attempt to respond to legitimate email inquiries from students within 48 hours. Please include "CSC265" in the subject line of the email.

Piazza

The link to sign up for our Piazza forum is **https://piazza.com/utoronto.ca/fall2023/csc265h1flec0101** **(https://piazza.com/utoronto.ca/fall2023/csc265h1flec0101)** . 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: 30%
- Midterm Exam: 26%
- Final Exam: 44%

There will be five *homeworks*, each worth 6%, to be done in groups of at most two. See **link** **(https://q.utoronto.ca/courses/314895/pages/homeworks)** .

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

There 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/314895/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 314895 pages csc265-fall-2023-enriched-data-structures-and-analysis)	**Lectures** (https:_q.utoronto.ca courses 314895 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.

We will not publish lecture notes. You are expected to read the assigned material, and make your own notes during lectures and tutorials. While making notes, you should not aim to record everything that is said and written during the lecture, but instead write down the main points that will help you to understand the reading material, and to study for exams.

Students often learn a lot from working with one another. You are encouraged to meet with other students taking the course. For example, you might work through exercises in the course text together or discuss any material you found confusing in lecture or in the text.

Tentative Schedule of Lectures

Unless otherwise specified, all readings are from Introduction to Algorithms, by Cormen, Leiserson, Rivest, and Stein, which will be abbreviated CLRS. You can find an **online version** (https:_librarysearch.library.utoronto.ca permalink 01UTORONTO_INST 14bjeso alma991106378851006196) of the 3rd edition from the U of T Libraries website.

Week and Topic	**Readings**
Week 1: Sept 11 –15 ADTs, Priority queues, Heaps	*Review*: Appendices A, C; Chapters 2, 3 Chapter 6
Week 2: Sept 18–22 Priority queues:	*Review*: Chapter 12; **Binomial heaps (from CLRS ed 2)** (https:_q.utoronto.ca courses 314895 files 27551906_wrap_1)

mergeable heaps Dictionaries: Review of BSTs	(https://q.utoronto.ca/courses/314895/files/27551906/download?download_frd=1) .
Week 3: Sept 25-29 Lower bounds	Section 8.1.; **Notes on lower bounds** (https://jeffe.cs.illinois.edu/teaching/algorithms/notes/12-lowerbounds.pdf) ; **Notes on adversarial method** (https://jeffe.cs.illinois.edu/teaching/algorithms/notes/13-adversary.pdf) **Watch video lectures!** (Links to be announced)
Week 4: Oct 2–Oct 6 Balanced Binary Search Trees	**Notes on AVL trees by V. Hadzilacos.** (https://q.utoronto.ca/courses/314895/files/27551905?wrap=1) (https://q.utoronto.ca/courses/314895/files/27551905/download?download_frd=1) Chapter 18
Week 5: Oct 9–13 Augmenting Data Structures Introduction to Hashing	Chapter 14; Sections 5.1-5.3 and 11.1-11.3 Review of Probability Theory (also take a look at Appendix C)
Week 6: Oct 16–20 Universal Hashing	Section 11.5; Sections 31.1-31.4; **Review of Modular Arithmetic** (https://q.utoronto.ca/courses/314895/files/27551908?wrap=1) (https://q.utoronto.ca/courses/314895/files/27551908/download?download_frd=1)
Week 7: Oct 23–27 Randomized Quicksort Amortized Analysis	Chapter 7; Sections 17.1-17.3
Week 8: Oct 30–Nov 3 Dynamic	Section 17.4; Sections 21.1-21.3

Tables Disjoint sets	
Nov 6–10 NO CLASS	
Week 9: Nov 11–17 Finish Disjoint sets	**log* analysis (https://q.utoronto.ca/courses/314895/files/27551907?wrap=1)** ↓ **(https://q.utoronto.ca/courses/314895/files/27551907/download?download_frd=1)** (from CLR 1st edition).
Week 10: Nov 18–24 BFS	Graphs: Appendix B.4 BFS: Sections 22.1-22.2
Week 11: Nov 25–Dec 1 DFS	Chapter 22.3
Week 12: Dec 2–8 Minimum spanning trees	Chapter 23

Learning Objectives

By the end of this course, you should have the following skills

- You should be able to analyze the worst case running time of algorithms and the worst case expected running time of randomized algorithms.
- You should be able to implement and use data structures for these ADTs: priority queue, dictionary, disjoint sets. In particular, you should know how to implement and use binary heaps, know a mergeable heap data structure, know how to implement and use balanced binary search trees, hash tables, and data structures for disjoint sets.
- You should be able to perform amortized analysis of sequences of operations on a data structure. You should be familiar with the basic tools of amortized analysis, in particular the potential function method.
- You should be familiar with data structures for storing graphs, and with basic graph algorithms, such as graph search algorithms and minimum spanning tree algorithms.
- You should be able to prove simple lower bounds on the worst case running time of algorithms in the decision tree model. You should be familiar with the information theoretic and the adversarial

methods of proving such lower bounds.

Homeworks

Navigation	Home page (https://q.utoronto.ca/courses/314895/pages/csc265-fall-2023-enriched-data-structures-and-analysis)	Lectures (https://q.utoronto.ca/courses/314895/pages/lectures)	Homeworks (https://q.utoronto.ca/courses/314895/pa

Overview

The course has 5 *group assignments* to be completed in groups of maximum two people each, and submitted on MarkUs.

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

Assignment	Date out	Date due	Problems/Solutions
Assignment 1	Sept 15	Sept 29	
Assignment 2	Oct 6	Oct 20	
Assignment 3	Oct 20	Nov 3	
Assignment 4	Nov 3	Nov 17	
Assignment 5	Nov 17	Dec 7	

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

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 date a homework assignment is returned. 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

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.