DALHOUSIE UNIVERSITY
Faculty of Computer Science

CSCI 6057/4117 - Advanced Data Structures
Course Syllabus
Winter 2024, Dalhousie University

Course website: https://dal.brightspace.com/d2l/home/311201
Students are responsible for checking announcements posted on Brightspace for assignment clarifications, course updates and other information.

Classes: 14:35-15:55, Tuesdays and Thursdays, Carleton Tupper Building Theatre D. First lecture on Tuesday January 9. Last lecture on Thursday April 4.

Instructor: Meng He
Office: Mona Campbell 4245, Phone: (902) 494-4056, Email: mhe [at] cs.dal.ca
Office hours: 16:00-17:00 Tuesdays and Thursdays. In addition, I'm typically available after lectures. You can also make an appointment.

Important Dates:

- Assignments due: January 30, February 13, March 5, March 26
- Midterm exam: February 27, in class
- Final exam (CSCI 4117 only): TBA in the period of April 11-23
- Project Proposal (CSCI 6057 only) due: March 12
- Project presentation (CSCI 6057 only): during the last 1-2 weeks of lectures. A detailed schedule will be available later.
- Project report (CSCI 6057 only) due: April 16.

Course Description: Data structures are fundamental in computer science. They play a central role in many modern applications (think of Google, the digital library of ACM, your mail server and even the operating systems on your computers). They are also essential building blocks of the design of efficient algorithms.

This course will cover some classical results and recent advancements on data structures, and the algorithms acting upon them. Typical topics include breaking the O(lg n) barrier in sorting and searching, reorganizing lists and search trees based on the online sequence of queries to speed up searches, improving efficiency based on the distribution of queries, performing fast text retrieval by constructing indexes, and improving space efficiency of data structures for large data sets.

The goal of this course is to improve students' skills of designing data structures and applying data structure techniques to their own research. This course is not only critical to students who are interested in algorithm research, but also very useful for students who are interested in efficiency issues in large systems and applications such as databases, information retrieval systems, bioinformatics applications and geographic information systems.

Course Prerequisites: CSCI 3110 or equivalent.

Learning Outcomes:

- Compute the entropy of a probability distribution.
- Compute the information-theoretic lower bound of representing a combinatorial object.
- Understand how to break the O(lg n) barrier in sorting and searching by allowing other operations on key values in addition to comparisons.
- Use van Emde Boas trees to implement an associative array.
- Understand various properties of splay trees, such as static optimality and working set properties.
- Use persistent data structures to solve classic data structure problems.
- Perform competitive analysis of self-organizing data structures.
- Take advantage of the non-uniform distribution of data sets to improve efficiency.
- Use full-text indexes such as suffix trees and suffix arrays to facilitate text search.
- Use word-level parallelism to speed up navigational operations on succinct data structures.
- Construct counterexamples to show that transpose and move halfway are not competitive.
- Design succinct data structures to improve space efficiency.
- Perform amortized analysis of data structures.
- Perform suffix sorting on text to construct suffix arrays and suffix trees.
- Prove why move to front is competitive.

Tentative List of Topics:

- Amortized Analysis and Resizable Arrays. One basic skill required for data structure design is amortized analysis, which analyzes algorithms by considering the entire sequence of operations of the program. One classic example regarding amortized analysis is resizable arrays. A typical resizable array structure, however, uses extra space linear in the length of the array. Surprisingly, in recent years, researchers studied this problem again, and found new solutions that use extra space proportional to the square root of the length of the array while providing optimal support for operations. This shows that advanced data structure design techniques can even improve fundamental data structures used in introductory programming courses.
- van Emde Boas trees: data structures in a bounded universe. Many data structures adopt the comparison-based model, in which keys are real numbers and only comparison operations can be performed over keys. In many applications, keys are of finite precision and thus can be treated as integers from a bounded universe. Can we improve query efficiency by making use of bit operations on integers? van Emde Boas tree is a classic example of beating the O(lg n)-bound on sorting and searching by performing bit tricks on integer data.
- Self-adjusting data structures. Given a sequence of search operations, what is the best way of organizing data? We will start with lists, and then learn about search

trees such as splay trees. This line of research is focused on designing efficient data structures to handle input piece-by-piece, without knowing the entire input in advance, which is required in many system applications.

- Entropy, nearly-optimal binary search trees and move-to-front compression. Data structures under the comparison-based model typically treat each query as equally likely, i.e., query distribution is uniform. If the query distribution is non-uniform, we can take advantage of this, and entropy is often used to analyze the query performance. We will use optimal binary search tree construction and move-to-front text compression as examples.
- Text indexing structures. Searching for substrings in textual data is important in information retrieval, text databases and bioinformatics applications. We will study how text indexes such as suffix trees and suffix arrays facilitate searching.
- Succinct data structures. Improving space efficiency of data structures is essential to applications that process massive data sets, such as web search engines and genomic databases. Succinct data structures use little space in addition to the size of the raw data, while providing fast support for operations.

Course Rationale

This course helps students gain the background needed for research on algorithms and data structures and the data structure knowledge needed to address performance issues in systems and applications.

Class Format and Course Communication:

- Content will be delivered via in-person lectures.
- Students must ask the instructor permission before recording class lectures.

Required Textbooks and Other Resources:

- No textbooks are required.
- An introductory book on algorithms and data structures such as *Introduction to Algorithms, 4th edition, Cormen, Leiserson, Rivest and Stein, MIT Press, 2022*, might be helpful, but is certainly not necessary.
- Note taking is expected.
- For topics covered when this course was taught online due to the pandemic (winter 2021 and 2022), recorded lectures and notes may also be made available at Brightspace. However, this is not possible for new or heavily updated topics.

Grading:

CSCI 6057	**CSCI 4117**
• 4 assignments (35%) • Midterm (15%) • Project proposal (5%) • Project presentation (10%) • Project report (35%)	• 4 assignments (40%) • Midterm (20%) • Final exam (40%)

Notes:

- The grade conversion scale in Section 17.1 of the Academic Regulations, Undergraduate Calendar will be used.

Midterm and Final Exam Requirements:

- Photo ID is required.
- Closed book.
- You are allowed to bring one and only one cheat sheet of letter size (8.5 by 11 inches), with anything written on both sides. You can also typeset your cheat sheet, but you are not allowed to tape or staple anything onto it.
- No other aids are allowed.

Assignments:
There will be four assignments. Each assignment is to be completed individually. Assignments will be handed out and due as follows:

Assignments Number	Handed Out	Due
1	January 16	January 30
2	January 30	February 13
3	February 13	March 5
4	March 12	March 26

Project (CSCI 6057 only!):
Basically, to complete the project, you need to choose a research problem related to data structures, read several papers on this problem, apply, improve, extend or implement the results you have found, and write a report of 5-10 pages to present your survey and findings. Detailed requirements are on Brightspace.

Pay attention to the section about plagiarism and legitimate paraphrasing.

Late Policy:
To help you budget your time, each student is automatically granted two 72-hour extensions free of penalty. They apply to assignments, the project proposal (in the case of CSCI 6057) and the project report (again, in the case of CSCI 6057), but they do not apply to exams or the CSCI 6057 project presentation. Each extension can be used for a single piece of coursework only. For example, you cannot combine two extensions to get a 144-hour extension on assignment 1, but you can use a 72-hour extension for assignment 1 and a 72-hour extension for assignment 2, after which you need to submit all remaining assignments, project proposal and project report on time.

You need not ask for these extensions; we will keep track. After you use up your free extensions, no further "late" will be granted. If there is a valid reason such as serious illness, contact the instructor.

Accommodation Policy:
CSCI 4117: If you are unable to write one or more assignments or the midterm for valid, well-documented reasons such as serious sickness, your final exam score can be used to cover the marks for your missing course work. Please contact the instructor if this applies to you.

If you miss the final exam due to a valid reason, an oral final exam will be scheduled, during which you will be asked to use a whiteboard to show your solutions.

CSCI 6057: If you are unable to write the midterm for valid, well-documented reasons, an oral midterm exam can be scheduled, during which you will be asked to use a whiteboard to show your solutions; alternatively, you can choose to write the final exam with undergraduate students.

Student Declaration of Absence
The Student Declaration of Absence policy shall apply. https://www.dal.ca/campus_life/safety-respect/student-rights-and-responsibilities/academic-policies/student-absence.html

Academic Standards
Failure to properly attribute sources in your work will be treated as an academic standards issue and points may be deducted for not following citation requirements. For example, forgetting to quote text taken from other sources, failure to include in-text citations, or a failure to include required information in the citations or references. Please see the resources on proper citation provided by the Dalhousie Writing Center (https://dal.ca.libguides.com/c.php?g=257176&p=5001261).

Please note that if it appears that the error was made with intent to claim other people's work as your own such as a lack of both citations and references, an allegation of plagiarism will be submitted to the Faculty Academic Integrity Officer, which could result in consequences such as a course failure.

Responsible Computing Policy
Usage of all computing resources in the Faculty of Computer Science must be within the Dalhousie Acceptable Use Policies (http://its.dal.ca/policies/) and the Faculty of Computer Science Responsible Computing Policy. For more information please see https://www.cs.dal.ca/downloads/fcs_policy_local.pdf

Use of Plagiarism Detection Software
All submitted code may be passed through a plagiarism detection software, such as the plagiarism detector embedded in Codio, the Moss (https://theory.stanford.edu/~aiken/moss/) Software Similarity Detection System, or similar systems. If a student does not wish to have their assignments passed through plagiarism detection software, they should contact the instructor for an alternative. Please note, that code not passed through plagiarism detection software will necessarily receive closer scrutiny.

https://cdn.dal.ca/content/dam/dalhousie/ pdf/dept/university_secretariat/policy-repository/OriginalitySoftwarePolicy.pdf

Culture of Respect[1]

Every person has a right to respect and safety. We believe inclusiveness is fundamental to education and learning. Misogyny and other disrespectful behaviour in our classrooms, on our campus, on social media, and in our community is unacceptable. As a community, we must stand for equality and hold ourselves to a higher standard.

What we all need to do:

1. **Be Ready to Act:** This starts with promising yourself to speak up to help prevent it from happening again. Whatever it takes, summon your courage to address the issue. Try to approach the issue with open-ended questions like "Why did you say that?" or "How did you develop that belief?"
2. **Identify the Behaviour:** Use reflective listening and avoid labeling, name-calling, or assigning blame to the person. Focus the conversation on the behaviour, not on the person. For example, "The comment you just made sounded racist, is that what you intended?" is a better approach than "You're a racist if you make comments like that."
3. A**ppeal to Principles:** This can work well if the person is known to you, like a friend, sibling, or co-worker. For example, "I have always thought of you as a fair-minded person, so it shocks me when I hear you say something like that."
4. **Set Limits:** You cannot control another person's actions, but you can control what happens in your space. Do not be afraid to ask someone "Please do not tell racist jokes in my presence anymore" or state "This classroom is not a place where I allow homophobia to occur." After you have set that expectation, make sure you consistently maintain it.
5. **Find or be an Ally:** Seek out like-minded people that support your views, and help support others in their challenges. Leading by example can be a powerful way to inspire others to do the same.
6. **Be Vigilant:** Change can happen slowly, but do not let this deter you. Stay prepared, keep speaking up, and do not let yourself be silenced.

University Statements

This course is governed by the academic rules and regulations set forth in the University Calendar and the Senate. https://academiccalendar.dal.ca/Catalog/ViewCatalog.aspx?pageid=viewcatalog&catalogid=111&loaduseredits=False

Territorial Acknowledgement

Dalhousie University is located in Mi'kma'ki, the ancestral and unceded territory of the Mi'kmaq. We are all Treaty people.

---

[1] Source: Speak Up! © 2005 Southern Poverty Law Center. First Printing. This publication was produced by Teaching Tolerance, a project of the Southern Poverty Law Center. Full "Speak Up" document found at: http://www.dal.ca/dept/dalrespect.html. Revised by Susan Holmes from a document provided April 2015 by Lyndsay Anderson, Manager, Student Dispute Resolution, Dalhousie University, 902.494.4140, lyndsay.anderson@dal.ca www.dal.ca/think.

Internationalization

At Dalhousie, 'thinking and acting globally' enhances the quality and impact of education, supporting learning that is "interdisciplinary, cross-cultural, global in reach, and orientated toward solving problems that extend across national borders." https://www.dal.ca/about-dal/internationalization.html

Academic Integrity

At Dalhousie University, we are guided in all of our work by the values of academic integrity: honesty, trust, fairness, responsibility and respect. As a student, you are required to demonstrate these values in all of the work you do. The University provides policies and procedures that every member of the university community is required to follow to ensure academic integrity. (read more: http://www.dal.ca/dept/university_secretariat/academic-integrity.html)

Accessibility

The Student Accessibility Centre is Dalhousie's centre of expertise for matters related to student accessibility and accommodation. If there are aspects of the design, instruction, and/or experiences within this course (online or in-person) that result in barriers to your inclusion please contact: https://www.dal.ca/campus_life/academic-support/accessibility.html for all courses offered by Dalhousie with the exception of Truro.

Conduct in the Classroom — Culture of Respect

Substantial and constructive dialogue on challenging issues is an important part of academic inquiry and exchange. It requires willingness to listen and tolerance of opposing points of view. Consideration of individual differences and alternative viewpoints is required of all class members, towards each other, towards instructors, and towards guest speakers. While expressions of differing perspectives are welcome and encouraged, the words and language used should remain within acceptable bounds of civility and respect.

Diversity and Inclusion — Culture of Respect

Every person at Dalhousie has a right to be respected and safe. We believe inclusiveness is fundamental to education. We stand for equality. Dalhousie is strengthened in our diversity. We are a respectful and inclusive community. We are committed to being a place where everyone feels welcome and supported, which is why our Strategic Direction prioritizes fostering a culture of diversity and inclusiveness (Strategic Priority 5.2). (read more: http://www.dal.ca/cultureofrespect.html)

Student Code of Conduct

Everyone at Dalhousie is expected to treat others with dignity and respect. The Code of Student Conduct allows Dalhousie to take disciplinary action if students don't follow this community expectation. When appropriate, violations of the code can be resolved in a reasonable and informal manner—perhaps through a restorative jus- tice process. If an informal resolution can't be reached, or would be inappropriate, procedures exist for formal dispute resolution. (read more: https://www.dal.ca/dept/university_secretariat/policies/student-life/code-of-student-con.html)

Fair Dealing Policy

The Dalhousie University Fair Dealing Policy provides guidance for the limited use of copyright protected material without the risk of infringement and without having to seek the permission of copyright owners. It is intended to provide a balance between the rights of creators and the rights of users at Dalhousie. (read more: https://www.dal.ca/dept/university_secretariat/policies/academic/fair-dealing-policy-.html)

Originality Checking Software

The course instructor may use Dalhousie's approved originality checking software and Google to check the originality of any work submitted for credit, in accordance with the Student Submission of Assignments and Use of Originality Checking Software Policy. Students are free, without penalty of grade, to choose an alternative method of attesting to the authenticity of their work, and must inform the instructor no later than the last day to add/drop classes of their intent to choose an alternate method. (read more: https://www.dal.ca/dept/university_secretariat/policies/academic/student-submission-of-assigments-and-use-of-originality.html)

Student Use of Course Materials

These course materials are designed for use as part of the CSCI courses at Dalhousie University and are the property of the instructor unless otherwise stated. Third party copyrighted materials (such as books, journal articles, music, videos, etc.) have either been licensed for use in this course or fall under an exception or limitation in Canadian Copyright law. Copying this course material for distribution (e.g. uploading material to a commercial third party website) may lead to a violation of Copyright law.

Learning and Support Resources

Please see https://www.dal.ca/campus_life/academic-support.html