

## COMP 2402: Abstract Data Types and Algorithms Winter 2021

**Instructor:** Alexa Sharp (she/her)

**Email:** [alexasharp3@cunet.carleton.ca](mailto:alexasharp3@cunet.carleton.ca)

**Lectures:** 2402B: Wednesdays & Fridays 14:35PM - 15:55PM (synchronous on zoom)  
2402A: pre-recorded (asynchronous) video lectures

**Office Hours:** Thursdays 1:00pm-3:00pm on BigBlueButton (cuLearn), or by appointment

**Course Website:** [piazza.com/carleton.ca/winter2021/comp2402](https://piazza.com/carleton.ca/winter2021/comp2402) and <https://carleton.ca/culearn/>

### Description

This course builds upon the principles introduced in COMP 1405 and COMP 1406 and provides a general background for further study in Computer Science. The course will cover object-oriented programming concepts; the design and implementation of data structures (linked lists, stacks, queues, trees, heaps, hash tables, and graphs) and related algorithmic techniques (searching, sorting, recursion); and algorithm analysis. Students will be expected to complete a number of programming projects illustrating the concepts presented. Prerequisites: one of COMP 1406, COMP 1006, SYSC 2004, with a minimum grade of C-.

### Lectures

For Section A, lectures will be synchronous (live) via [Zoom](#), and you are expected to attend if circumstances allow. Recordings of the lecture, with transcript, will be provided at a later time (within 12 hours) on cuLearn. Section B students are welcome to attend these lectures live if timing permits, but otherwise are given the lecture recordings.

Students are expected to remain up to date with the deadlines and due dates provided by the instructor. This course requires reliable high-speed Internet access.

In live lectures, please mute your microphone, but turn on video if your internet connection can handle it. If you have questions, please ask in the chat or virtually "raise your hand".

Lectures are recorded to enable access to students with internet connectivity problems, who are based in different time zones, and/or who have conflicting commitments. If you wish not to be recorded, you need to leave your camera and microphone turned off.

You will be notified at the start of the session when the recording will start, and Zoom will always notify meeting participants that a meeting is being recorded. It is not possible to disable this notification.

Please note that recordings are protected by copyright. The recordings are for your own educational use, but you are not permitted to publish to third party sites, such as social media sites and course materials sites.

## Textbook

The textbook for the course is Pat Morin's Open Data Structures. Free PDF and HTML versions of the book are available at [opendatastructures.org](http://opendatastructures.org). You'll probably want to use the Java version. There are also links there explaining where you can purchase a paperback version.

## Necessary Equipment and Software

A java compiler and your favourite editor. Something to watch the lectures on.

## Course Work & Grading Scheme

### Programming Assignments (60%)

The best computer scientists are the ones that have had the most practice. The programming assignments in this course are meant to give you the opportunity to practice with the topics of this course in a way that is challenging yet also manageable. At times you may struggle and at others it may seem more straight-forward; just remember to keep trying and practicing, and over time you will improve. The only way to really practice with the problem solving process is to experience it.

**There are 5 programming assignments. The best 4 are worth 15% each.** The lowest assignment grade will be dropped (or, more specifically, used as bonus.) No lates are accepted.

### Drills (10%)

If you're a professional soccer player, you practice both match play with scrimmages and skills with focused drills (such as passing or shooting drills).

If you're a professional pianist, you practice both performance with dry runs through your piece, and skills with focused drills (such as practicing certain bars or one hand at a time.)

If you aspire to be a professional computer scientist, you should practice both writing larger and larger programs, and also the individual skills you need to write such programs.

The programming assignments in this course are an attempt to get you the former sort of practice—to simulate the kinds of tasks you may be asked to complete as a software developer. But as with soccer matches and piano performance, the “direct” task can be made easier and more intuitive, with skills practice. That is what these drills attempt to provide.

After each lecture, I will post 1-5 lecture-related “drills”. These are meant to take a few minutes each, and are meant to either hone your intuition for the material, or give you technical practice. They will often focus on the big picture—on the ability to compare and contrast, and to make quick assessments based on intuition—which should help give you a better sense of direction when attacking a new homework assignment. Also, they are similar to test questions—in fact, 50% of the midterm and test questions are drawn directly from the drills.

**There are 22 drills (~1 per lecture). The best 10 are worth 1% each.** The remaining 12 grades will be dropped (or, more specifically, used as bonus). Drills are due within a week of the associated lecture. Multiple attempts within the week are allowed.

Midterm & Final Exam (12%, 18%)

While tests aren’t particularly representative of how you may use your computer science knowledge in practice, they provide some advantages.

- Studying for the exams hopefully improves your recall so that some of the more fundamental information can be recalled quickly.
- They encourage you to review all the course material, not just what is needed for the programming assignments and drills.
- They are the only coursework that evaluates you individually, since the remaining coursework can be collaborated on.

Having said that, both the midterm and the final are worth around as much as an assignment.

**The midterm is worth 12% and the final 18%.** The midterm and final will be multiple-choice questions on cuLearn (50% of questions drawn directly from the drills). The final is cumulative.

## Bonus

There are many opportunities for bonus points, if you have the time and inclination. The 5th programming assignment, and 12 additional drills are available for bonus, and possibly other opportunities as the semester progresses.

**Bonus points are completely optional;** *not* doing bonus points will not negatively impact your final grade. I first compute final grades without considering bonus. Then I look at bonus points to see if they might bump you up a notch or two. Bonus points will not impact your grade by more than a letter, and it cannot move you from an F to a passing grade. You have to pass the required course material in order to pass the course.

## Important Dates

Drills	due within a week of the associated lecture
Assignment 1	due Sunday Jan 24 23:55PM
Assignment 2	due Sunday Feb 7 23:55PM
Assignment 3	due Sunday Feb 28 23:55PM
Midterm Test	Mar 3-4 (section A, B, respectively)
Assignment 4	due Sunday Mar 21 23:55PM
Assignment 5	due Sunday Apr 11 23:55PM
Final	TBD on cuLearn

## How to Get Help

There are many ways to get help on your work in this course, that do not violate the course's academic integrity policy:

- Q&A Forum on [piazza](#)
- Office Hours posted on [piazza](#)
- Discord

## Academic Integrity

Assignments in this course involve coding. Students may collaborate on assignments at the level of discussion, but must write code on their own. Any students caught submitting

copied code or overly sharing details of their code will be reported to the Associate Dean (Undergraduate) who will investigate the matter. The standard penalties for an academic integrity violation are as follows:

- First offence: F in the course.
- Second offence: One-year suspension from program.
- Third offence: Expulsion from the University.

These are standard penalties. More-severe penalties will be applied in cases of egregious offences. For more information, please see Carleton University's [Academic Integrity Policy](#).

## Respect in the Classroom and Forums

Please remember to treat your peers and the course staff with respect. This includes in the zoom chat and on any course-related forums such as piazza and discord. It is not acceptable to use offensive language nor disparage a person or group, no matter the intent. Treat the course spaces as professional spaces and behave accordingly. Behavioural misconduct may be reported to Student Affairs.

If you feel you have been disrespected or abused either by other students or course staff, please let us know (you can **contact us anonymously and privately on piazza**, for example.)

## Statement of Accommodation

The Carleton University Information on [Academic Accommodation](#) applies to this course. Here is [information on how to apply for academic accommodation](#). If there is anything Alexa can do to help you succeed, please let her know as soon as possible so that she can accommodate accordingly.

## Late Policy

Late work is not accepted. Please plan accordingly.

## University Policy

In addition to anything included here, all the standard Carleton University Policies regarding equity and academic regulations apply to this course.

## Undergraduate Academic Advisor

The Undergraduate Advisor for the School of Computer Science is available in Room 5302C HP; by telephone at 520-2600, ext. 4364; or by email at [undergraduate\\_advisor@scs.carleton.ca](mailto:undergraduate_advisor@scs.carleton.ca). The undergraduate advisor can assist with information about prerequisites and preclusions, course substitutions/equivalencies, understanding your academic audit and the remaining requirements for graduation. The undergraduate advisor will also refer students to appropriate resources such as the Science Student Success Centre, Learning Support Services and Writing Tutorial Services.

## SCS Computer Laboratory

SCS students can access one of the designated labs for your course. The lab schedule can be found at: <https://carleton.ca/scs/tech-support/computer-laboratories/>. All SCS computer lab and technical support information can be found at: <https://carleton.ca/scs/technical-support/>. Technical support is available in room HP5161 Monday to Friday from 9:00 until 17:00 or by emailing [support@scs.carleton.ca](mailto:support@scs.carleton.ca).

## Copyright

My lectures and course materials (including all slides, programs, handouts, videos, and similar materials) are protected by copyright. I am the exclusive owner of copyright and intellectual property of all course materials. You may take notes and make copies of course materials for your own educational use. You may not allow others to reproduce or distribute lecture notes and course materials publicly for commercial purposes without my express written consent.

## Territory Acknowledgement

I would like to acknowledge that the location of the Carleton University campus is on the traditional, unceded territories of the Algonquin nation. In doing so, I acknowledge that I and Carleton University have a responsibility to the Algonquin people and a responsibility to adhere to Algonquin cultural protocols.