

## Abstract Data Types and Algorithms

### COMP 2402AB (Winter 2024)

### Course Information

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 using the concepts presented. **Precludes** additional credit for COMP 2002 (no longer offered), SYSC 2002 (no longer offered), SYSC 2100. Prerequisites: one of COMP 1406, COMP 1006, SYSC 2004, with a **minimum grade of C-**.

<b>Instructor</b>	Alexa Sharp (she/her) --- call her Alexa or Prof Alexa			
<b>Email</b>	<a href="mailto:alexasharp3@cunet.carleton.ca">alexasharp3@cunet.carleton.ca</a> (note: <b>do not use</b> <a href="mailto:cmail.carleton.ca">cmail.carleton.ca</a> )			
<b>Lectures</b>	Section A	Wed & Fri	1:05pm - 2:25pm	hyflex
	Section B	online	anytime	asynchronous
<b>Location</b>	see Carleton's schedule on carleton central for most up-to-date location			
<b>Student Hours</b>	<a href="#">schedule</a>	held both in-person and on discord voice channel		
<b>Course Website</b>	<a href="https://brightspace.carleton.ca/d2l/home/">https://brightspace.carleton.ca/d2l/home/</a>			
<b>Course Resources</b>	<a href="#">lecture zoom link</a>	posted on brightspace		
	<a href="#">lecture recordings</a>	posted on course schedule and brightspace		
	<a href="#">piazza</a>	<a href="https://piazza.com/carleton.ca/winter2024/comp2402">piazza.com/carleton.ca/winter2024/comp2402</a>		
	<a href="#">topic schedule</a>	posted on brightspace		
	<a href="#">gradescope.ca</a>	access your account through brightspace		
<b>Q&amp;A Forums</b>	<a href="#">piazza</a> (primary, structured), <a href="#">discord</a> (non-anonymous, informal)			
<b>Textbook</b>	Pat Morin's <i>Open Data Structures</i> . <b>Free</b> PDF and HTML versions of the book are available at <a href="https://opendatastructures.org">opendatastructures.org</a> ; You'll want the Java version.			

**Inclusivity Statement.** I am committed to fostering an environment for learning that is inclusive for everyone regardless of gender identity, gender expression, sex, sexual orientation, race, ethnicity, ability, age, class, etc. All students in the class, the instructor, the course staff, and any guests should be treated with respect during all interactions.

**Land Acknowledgement.** I would like to acknowledge that the land on which we gather is the traditional, unceded territory of the Anishinaabe 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. More information on Prof Alexa's work to take some responsibility can be found on piazza in the post titled "Beyond the Land Acknowledgement."

## Learning Outcomes and Topics Covered

By engaging with the course material through regular practice, a student should:

1. gain a deeper understanding of how data organization choices impact program performance, including that there is not one perfect data structure;
2. become comfortable reading, writing, and understanding algorithms, including analysis of their time and space efficiency; and
3. master the art of abstraction in order to reduce code complexity and increase code manageability and readability.

A detailed list of topics to be covered, including dates, deadlines, and required reading for each week, are posted on our [course schedule](#).

## Lectures

**Whatever your course delivery preference for a given day, there is an option for you.**

If you want to attend **in-person**, you are welcome to attend section A's lecture (even if you're section B). Prof Alexa believes that in-person lecture is the most engaging and effective way to learn the material, but understands that flexibility is important to you.

If you want to attend **live over zoom (hyflex)**, you are welcome to attend Section A's lecture over zoom. This is not as good as attending in person, as you cannot see all boards at the same time, and **Prof Alexa cannot effectively take your questions**, but it is good enough given the circumstances. You can find the most up-to-date zoom lecture link on brightspace; please do not share this link with anyone outside of this course. Please mute your microphone; if you have questions, please ask by typing it in the chat; Prof Alexa will get to it if she can. Do not use the chat for irrelevant comments so that Prof Alexa has a chance to see your questions.

If you want to view **asynchronous recordings** (on your own schedule), Prof Alexa has recordings posted on brightspace and the [course schedule](#). There are two options for these recordings: the videos of the live classroom lectures, and (minorly edited) videos of the live at-home lectures from Winter 2022. (The course material has not changed so these are still up-to-date!)

**Asynchronous learners** are expected to remain up-to-date with deadlines and due dates. Please note that recordings are protected by copyright; they are for your own educational use. You must not publish to third party sites, such as social media sites and course materials sites. Note that while almost all of this course may be taken asynchronously, **you must be available synchronously (but not in-person) for our midterm and final tests**, scheduled by Carleton.

If you switch between modalities (which is ok!), you may encounter minor variation in the schedule and presentation due to variation in student questions and Prof Alexa's state of mind.

## Community Guidelines

The following values are fundamental to academic integrity and are adapted from the International Center for Academic Integrity<sup>1</sup>. We will seek to behave with these values in mind:

	<b>As students, we will...</b>	<b>As a teaching team, we will...</b>
<b>Honesty</b>	<ul style="list-style-type: none"> <li>• Honestly demonstrate our knowledge and abilities on assignments and exams</li> <li>• Communicate openly without using deception, including citing appropriate sources</li> </ul>	<ul style="list-style-type: none"> <li>• Give you honest feedback on your demonstration of knowledge and abilities on assignments and exams</li> <li>• Communicate openly and honestly about the expectations and standards of the course through the syllabus, and with respect to assignments and exams</li> </ul>
<b>Responsibility</b>	<ul style="list-style-type: none"> <li>• Complete assignments on time and in full preparation for class</li> <li>• Show up to class on time, and be mentally/physically present</li> <li>• Participate fully and contribute to team learning and activities</li> </ul>	<ul style="list-style-type: none"> <li>• Give you timely feedback on your assignments and exams</li> <li>• Show up to class on time, and be mentally &amp; physically present</li> <li>• Create relevant assessments and class activities</li> </ul>
<b>Respect</b>	<ul style="list-style-type: none"> <li>• Speak openly with one another, while respecting diverse viewpoints and perspectives</li> <li>• Provide sufficient space for others to voice their ideas</li> </ul>	<ul style="list-style-type: none"> <li>• Respect your perspectives even while we challenge you to think more deeply and critically</li> <li>• Help facilitate respectful exchange of ideas</li> </ul>
<b>Fairness</b>	<ul style="list-style-type: none"> <li>• Contribute fully and equally to collaborative work, so that we are not freeloading off of others</li> <li>• Not seek unfair advantage over fellow students in the course</li> </ul>	<ul style="list-style-type: none"> <li>• Create fair assignments and exams, and grade them in a fair, and timely manner</li> <li>• Treat all students equitably</li> </ul>
<b>Trust</b>	<ul style="list-style-type: none"> <li>• Not engage in personal affairs while on class time</li> <li>• Be open and transparent about what we are doing in class</li> <li>• Not distribute course materials to others without authorization</li> </ul>	<ul style="list-style-type: none"> <li>• Be available to all students when we say we will be</li> <li>• Follow through on our promises</li> <li>• Not modify the expectations or standards without communicating with everyone in the course</li> </ul>
<b>Courage</b>	<ul style="list-style-type: none"> <li>• Say or do something when we see actions that undermine any of the above values</li> <li>• Accept a lower or failing grade or other consequences of upholding and protecting the above values</li> </ul>	<ul style="list-style-type: none"> <li>• Say or do something when we see actions that undermine any of the above values</li> <li>• Accept the consequences (e.g., lower teaching evaluations) of upholding and protecting the above values</li> </ul>

<sup>1</sup> This class statement of values is adapted from Tricia Bertram Gallant, Ph.D.

## Online Community Expectations

Please be considerate and respectful while engaging with peers and course staff and remember that we are all humans, and that your words matter. 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. We recommend you read over our piazza posting etiquette as well as our discord #rules-please-read channel. You are responsible for behaving within these parameters; violations will result in loss of access privileges to these course resources, and a report to Student Affairs.

If any student witnesses or experiences harassment, I encourage you to reach out to me. Alternatively, you can contact [Ombuds Services](#) or [Carleton Equity and Inclusive Communities](#).

## Course Work & Evaluation

### Schedule, Readings, Deadlines, Lecture Notes & Recordings

The [course schedule](#) is a detailed document with textbook readings, lecture notes, lecture and workshop recordings, as well as links to and deadlines for labs, drills, and any other coursework. You'll want to bookmark the page. For information about Carleton's academic year, including registration and withdrawal dates, see [Carleton's Academic Calendar](#).

### Important Dates & Deadlines

**Note:** 3:00pm weekday deadlines ensure that Prof Alexa is available should things go awry.

<b>Drill Practice</b>	3:00pm of the <b>Monday</b> the week after associated lectures
<b>Pre-labs</b>	3:00pm of the <b>Friday</b> the week before relevant lab due
<b>Post-labs</b>	3:00pm of the <b>Wednesday</b> the week after relevant lab due
<b>Lab 1</b>	3:00pm on <b>Wednesday Jan 24</b>
<b>Lab 2</b>	3:00pm on <b>Wednesday Feb 7</b>
<b>Lab 3</b>	3:00pm on <b>Wednesday Feb 28</b>
<b>Midterm Test (brightspace)</b>	TBD by Carleton
<b>Lab 4</b>	3:00pm on <b>Wednesday Mar 13</b>
<b>Lab 5</b>	3:00pm on <b>Wednesday Mar 27</b>
<b>Final Test (brightspace)</b>	TBD by Carleton

## Grade Computation

Drills (best 8)	10%	Pre-labs	4%	Bonus	3%
Midterm Test	12%	Programming Labs	48%		
Final	18%	Post-labs	8%		

### Drills (10%)

While the end goal of this course is to get you comfortable writing sophisticated, well-designed algorithms, there are many necessary “building block” skills to pick up first. Just as soccer players prepare for matches with passing and shooting drills, you will prepare for your labs (programming assignments) with focused drills that target the knowledge and approaches you need to solve problems and communicate your solution. The drills aim to make the “smaller stuff” second nature, so that you can more easily put it together into masterful programs.

Each week of lecture has an associated drill due on brightspace within a few days of the scheduled lectures. These comprise of 5-15 lecture-related “drill” questions on brightspace. Multiple attempts within the week are allowed and encouraged with the highest score kept, to keep the assessment low-stakes and true to its purpose of drilling the material. Think of this as duo-lingo for discrete structures. The regular, tight schedule of the drills encourages you to keep up with the course. As further incentive to do the drills regularly, ~40% of the midterm and final are drawn directly from the drills (with minor adjustments).

**There are 12 drills (1 per week). To account for emergencies and life conflicts, the best 8 are worth 1.25% each;** the remaining 4 are dropped (or, more specifically, used as bonus).

### Labs (60%)

The best computer scientists are the ones with the most (effective) practice. The labs in this course are meant to give you the opportunity to get hands-on programming practice with the data structures and algorithms of this course in a way that is challenging yet also manageable. **You should be out of your comfort zone** (but not overwhelmed.) 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. Everyone learns differently; be patient with yourself. There is no substitute for practice and experience.

While you are encouraged to collaborate with your peers, you should formulate and write up your solutions on your own. Guidance regarding outside sources are in the section on academic integrity. If you are struggling, we have [many ways to help!](#)

**There are 5 labs. To account for emergencies, the best 4 are worth 15% each;** the lowest lab grade (the pre-lab, lab, post-lab combo) will be dropped (or, more specifically, used as bonus.) **Please “save” your dropped lab for unforeseen emergencies. I cannot drop 2 labs.**

Each lab assignment (15% each) is broken into 3 assessments, discussed in more detail below

- Pre-lab (1%) - multiple-choice questions due 5 days before the programming portion
- Programming Lab (12%) - the programming portion
- Post-lab (2%) - multiple-choice questions due one week after the programming portion

## Pre-Labs

We want to encourage you to have good programming habits, such as carefully reading the specifications of a problem, trying the problem out on examples, and writing a basic “correct” solution before moving on to more complex solutions. We also want to encourage good academic habits, such as starting early, thinking about the objectives of an assignment when working on it, knowing what resources are available to you for help, and using your time more effectively. **The pre-lab is a multiple-choice-style “quiz” on brightspace due 5 days before the associated programming lab that encourages you to adopt these good programming and academic habits.** In order to complete the pre-lab you should carefully read the lab specifications, think about the problems a bit, and possibly watch the lab workshop video posted on the course schedule in the relevant lab’s cell. Two attempts are allowed; your highest score is kept. No lates are allowed.

## Programming Labs

This is the programming portion of the lab assignment, where you usually complete 5-10 problems. Each problem will be practice either with a new data structure or algorithm listed at the top of the specifications, or with previous data structures and algorithms, for comparison purposes. You should work on these problems incrementally (rather than in one sitting) so that you can get help if you get stuck.

While you are encouraged to collaborate with your peers at an ideas-level, you should construct your programs on your own. You **may** use the web/internet and AI bots such as chatGPT for help with syntax-level questions, **but not for algorithmic questions (i.e. it can help you find individual words or phrases, but should not be writing the ideas of the “essay” for you.)** You may **not** use web search/the internet or AI services such as copilot or chatGPT to give you general approaches to the labs or to write any part of your code and you may **not** share or receive code from anyone. If you need help, [we have many ways to provide it](#) without violating [academic integrity](#). Automated tools for detecting plagiarism and AI services will be employed in this course. Your code may be subject to random (manual) code reviews by our TA staff.

Labs are due at 3:00pm on Wednesdays on gradescope (details on each lab specification). You should submit each individual part of each lab to gradescope, where it will be graded automatically. **Submit early and often to avoid last-minute technical issues** (your most recent

score on each part is kept). Lates are accepted within 24 hours, after which no lates are accepted due to posted solutions. If you have an emergency, please see the [Late Policy & Emergencies](#) section.

## Post-Labs

The programming portion of the labs are meant for students to really get their hands dirty with the course concepts. In a perfect world, students use the concepts from lecture and apply them to new (but related) problems, and in doing so, deeply engage with and internalize the important course concepts that the lab focuses on.

However, it is often the case that students *do* engage with the concepts at some level but aren't able to get all the points on the automated grader due to errors in their code. It's also the case that sometimes it's hard to make the connections between what was done and the course material. The post-labs are meant to help both of these problems. They are meant as a reflection on the lab and what it was meant to accomplish. It is also an opportunity for those that weren't able to complete the programming portion of the lab to at least engage with the material at some level.

The post-labs come out after the lab is due, and ask questions that ideally you can answer relatively quickly if you've done the lab, and with some effort (but not as much as doing the lab proper) if you haven't done the lab; a solutions walk-through video will be available to help you. In this way, everyone still gets something out of the lab regardless of whether you were able to do the programming. It doesn't replace programming (as that's the best way to learn the material) but it at least acknowledges that you can still get something out of it even if you didn't do it. Post-labs (on brightspace) are due by 3:00pm on Wednesday of the week following the associated assignment. Two attempts are allowed; your highest score is kept. No lates 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 tests improves your memory for faster recall of fundamentals.
- Tests encourage you to review all the course material, not just what is needed for the programming labs and drills.
- They are the only coursework that evaluates you as an individual.

Having said that, both the midterm and the final are worth around as much as a lab.

**The midterm is worth 12% and the final 18%.** The midterm and final will be multiple-choice questions on brightspace (~40% of questions drawn directly from the drills, with minor modifications). The final is cumulative. More information about the tests will follow on piazza, closer to their dates. You do not need to pass the tests or final to pass the course, but if you don't take them they count as a 0. These will be proctored online; details closer to the tests.



## Late Policy, Emergencies, and Accommodations for Missed Work

Late drills, pre-labs, and post-labs are not accepted.

Late programming labs are accepted within 24 hours at no penalty after which no lates are accepted, no exceptions. This is due to posted solutions and gradescope logistics. **Please submit early and often** so that last-minute technical problems don't derail your mark.

In lieu of requiring students to submit a self-declaration form in the case of an emergency, I prefer the following, more flexible, policy. You can, without explanation, not complete up to 4 drills, and 1 lab (pre-lab, lab, and post-lab combo). **Please save this for emergencies such as personal or family illnesses.** Note that these accommodations are for short-term concerns related to missed work; if you are experiencing chronic, ongoing challenges which necessitate a broader solution, I recommend reaching out to the Paul Menton Centre and/or the Care Support team. You need to do a minimum amount of practice in this course in order to deeply engage with the material, as such, further accommodations will not be available.

### Bonus (a.k.a. Extra Credit)

There are many opportunities for bonus points, if you have the time and inclination. Any drills and labs outside of your best submissions (as described above) are available for bonus, as well as other opportunities throughout the semester. The bonus opportunities and the weights associated with them will be tracked in a piazza post labeled "Bonus Opportunities."

Bonus points are completely optional, so ignore them if they are not for you. Bonus points will add at most 3% to your final mark, and they cannot be applied to an F grade.

## Late Registration

While the registrar will allow you to register for this course as late as Wednesday January 24th, **we strongly discourage registration past January 15th**, as it is difficult to miss more than a week of class and still catch up. We do not wait until January 24th to start the material; we start on January 9th! If you choose to join the course late, it is your responsibility to [read the following document](#) to get caught up. There are no special accommodations for late registrants.

## Necessary Equipment, Accounts & Software

You will need

- an internet-enabled device (laptop/desktop/tablet) and access to reliable internet. Please review the requirements at <https://carleton.ca/scs/scs-laptop-requirement/laptop-specs/>.
- free accounts on [piazza](#), and discord; use a recognizable name on these platforms.
- zoom software if you intend to attend the HyFlex lectures.
- a java compiler and your favourite editor to complete the lab assignments.



Students taking a COMP course can access the SCS computer labs. The lab schedule and location 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/tech-support/>. Technical support staff may be contacted in-person or virtually, see this page for details <https://carleton.ca/scs/tech-support/contact-it-support/>.

## 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. Please use these resources; they are there for you!

- Reference the [problem solving tips sheet](#) (posted on schedule)
- Check the Q&A Forum on [piazza](#) and ask questions there
  - please read the posting etiquette so that you get the fastest, most useful answer
- Attend or watch the lab-specific workshops
- Do the pre-labs
- Go to Student Hours on discord/in person (schedule on [piazza](#) and on this [spreadsheet](#))
  - dedicated times through the week for the course staff to meet with YOU. Pop in to introduce yourself, ask questions about the course, or discuss course content.
  - good for questions not suited to piazza (e.g. that require more back-and-forth),
  - some student hours are in-person, some on discord.
- Check the Q&A on discord (ideally after you've searched piazza to confirm it hasn't been answered there.)

## Emails to Prof Alexa

Prof Alexa has >300 students this semester. To help her provide the best service where it is most needed, please read [the following document](#) before sending her email, in case your inquiry can be better and more promptly answered by another means or person. Note that **there is a student (e.g. cmail) email account that does not reach Prof Alexa**. You must send any email to the cunet email listed at the top of the Syllabus. Prof Alexa tries not to work in the evenings and weekends so during those times please use a piazza post to Instructors if possible.

## Academic Integrity

Every student should be familiar with the Carleton University student academic integrity policy. A student found in violation of academic integrity standards may be awarded penalties which range from a reprimand to receiving a grade of F in the course or even being expelled from the program or University. Examples of punishable offences include plagiarism and unauthorized collaboration. Any such reported offences will be reviewed by the office of the Dean of Science. Information on this policy may be found on the ODS Academic Integrity page [here](#).

**Plagiarism.** As defined by the Senate, "plagiarism is presenting, whether intentional or not, the ideas, expression of ideas or work of others as one's own". Such reported offences will be reviewed by the office of the Dean of Science. Standard penalty guidelines can be found [here](#).

**Unauthorized Co-operation or Collaboration.** Senate policy states that "to ensure fairness and equity in assessment of term work, students shall not co-operate or collaborate in the completion of an academic assignment, in whole or in part, when the instructor has indicated that the assignment is to be completed on an individual basis".

Labs in this course involve coding, which benefits from bouncing ideas off of other people.

You **may** talk with peers and TAs at a high-level. But **you must formulate and write up (i.e. problem solve) your code on your own**. A solution that is basically line-by-line the same as a peer's is too close. Helping a peer with simple debugging is allowed, but be careful that it is a slippery slope between simple debugging and giving away bigger parts of the solutions.

You **must not** show or otherwise share your solution with your peers or anyone on the internet.

You **must not** use the internet or AI programmers (including, e.g. chatGPT, github copilot) to search for or solicit approaches or ideas. You may use these sources to help with syntax-level questions or ask about general course concepts. Think of them as a tutor, not someone doing your work for you.

You **must not** post any of the assessments or their solutions online (to sites like Chegg, CourseHero, OneClass, etc.) at any point in time. You are never permitted to post, share, or upload course materials without explicit permission from your instructor.

Any violation of these rules is a very serious offence and will be treated as such; they are reported to the Dean of Academic Integrity, who launches an investigation. Academic integrity is upheld in this course to the best of Prof Alexa's abilities, as it protects the students that put in the effort to work on the course assessments within the allowable parameters.

Note that contract cheating sites are known, unauthorized, and regularly monitored. Some of these services employ misleading advertising practices and have a high risk of extortion.

Automated tools for detecting plagiarism and AI programmers will be employed in this course.

## 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 you are allowed extra time on tests, ventus should take care of it (and it is your responsibility to confirm such accommodations.) If there is anything Prof Alexa can do to help you succeed, please let her know as soon as possible so that she can accommodate accordingly.

## Copyright

Prof Alexa is the exclusive owner of copyright and intellectual property of all course materials, including all notes, programs, labs, quizzes, handouts, videos, etc. **You may not reproduce or distribute lecture notes, labs, solutions, or any other course materials publicly for any reason, or allow others to, without express written consent.**

## Undergraduate Academic Advisors

The Undergraduate Advisor for the School of Computer Science is available in Room 5302 HP, or by email at [scs.ug.advisor@carleton.ca](mailto:scs.ug.advisor@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.

## Mental Health

If you are struggling, please do not hesitate to reach out. I am happy to listen, and/or direct you to resources that might help. In terms of class, if you need extra help or missed a lesson, don't stress! There is a lot of flexibility built into the grading scheme. Remember that Carleton also offers an array of mental health and well-being resources, which can be found [here](#).

## University Policies

Carleton is committed to providing academic accessibility for all individuals. Please review the academic accommodation available to students: <https://students.carleton.ca/course-outline/>.

## SCS Tech Support

Technical support information can be found at: <https://carleton.ca/scs/techn-support/>. Technical support is available in room HP5161 Monday to Friday from 9:00 until 17:00 or by emailing [SCS.Tech.Support@cunet.carleton.ca](mailto:SCS.Tech.Support@cunet.carleton.ca).