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-.

Instructor Alexa Sharp (she/her) --- call her Alexa or Prof Alexa
Email alexasharp3@cunet.carleton.ca (note: do not use cmail.carleton.ca)
Lectures Section A (hyflex): Tues & Thurs 11:35pm - 12:55pm
Section B (asynchronous – no scheduled time): recordings provided
Location see Carleton's schedule for most up-to-date location
Student Hours schedule held both in-person and on discord voice channel
Course Resources lecture for most up-to-date zoom id, see brightspace
piazza piazza.com/carleton.ca/winter2023/comp2402
schedule topics, readings, slides, videos, deadlines
brightspace https://brightspace.carleton.ca/d2l/home/
textbook http://opendatastructures.org/
Q&A Forums piazza (primary, structured), discord (non-anonymous, informal)

Lectures

Whatever your course delivery preference, there is an option for you.

For any given lecture, if you want to attend live, 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.

For any given lecture, if you want to attend live over zoom, you are welcome to attend Section A’s lecture over zoom (even if you’re section B). 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 — but realistically she doesn't usually see them. Do not use the chat for irrelevant comments so that Prof Alexa has a chance to see your questions.

For any given lecture, if you want to view asynchronous recordings (on your own schedule), you are welcome to view section B's recordings that are posted on the course schedule (even if you're section A). 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 students are expected to remain up-to-date with deadlines and due dates. Please note that recordings are protected by copyright. The recordings are for your own educational use; you are not permitted to publish to third party sites, such as social media sites and course materials sites.

**To be clear: this is very flexible.** Technically, there is a HyFlex section (A) and an asynchronous (unscheduled) section (B). Regardless of which section you are registered in, you are welcome to the resources for either section, whether that is attending sometimes in-person, sometimes over zoom, or sometimes via video recordings. On any given day you choose, and you do not (and should not) inform Prof Alexa of your choice. The course work for both sections (and the material covered) is the same.

**Asynchronous Learners**

Note that while almost all of this course may be taken asynchronously by any student (regardless of section), you must be available synchronously (but not in-person) for our midterm test (see the schedule) and our final test, which is scheduled by Carleton at a later time.

**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. 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**

Every student that has been enrolled in a 1000-level (i.e., first year) course offered by the School of Computer Science after the 2020/2021 school year is required to have a laptop. This includes COMP1001, COMP1005, and COMP1006. For more information, please visit https://carleton.ca/scs/scs-laptop-requirement/ and then review the requirements at https://carleton.ca/scs/scs-laptop-requirement/laptop-specs/.

You will need an internet-connected device to access video lectures and notes, brightspace for drills and tests, gradescope for labs, and piazza and discord for Q&A and office hours. You will need a java compiler and your favourite editor to complete the lab assignments.
You will need (free) accounts on piazza, gradescope, and discord; please use your preferred (recognizable) name on these platforms. You will need a secret key from our course submission server.

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/.

Course Work & Evaluation

Important Dates & Deadlines

<table>
<thead>
<tr>
<th>Drill Practice</th>
<th>3:00pm of the Monday the week after associated lectures</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pre-labs</td>
<td>3:00pm of the Wednesday the week before relevant lab due</td>
</tr>
<tr>
<td>Post-labs</td>
<td>3:00pm of the Wednesday the week after relevant lab due</td>
</tr>
<tr>
<td>Lab 1</td>
<td>3:00pm on Wednesday Jan 25</td>
</tr>
<tr>
<td>Lab 2</td>
<td>3:00pm on Wednesday Feb 8</td>
</tr>
<tr>
<td>Lab 3</td>
<td>3:00pm on Wednesday Mar 1</td>
</tr>
<tr>
<td>Midterm Test (brightspace)</td>
<td>during your section's scheduled time on Mar 13 or 14</td>
</tr>
<tr>
<td>Lab 4</td>
<td>3:00pm on Wednesday Mar 15</td>
</tr>
<tr>
<td>Lab 5</td>
<td>3:00pm on Wednesday Mar 29</td>
</tr>
<tr>
<td>Final Test (brightspace)</td>
<td>TBD by Carleton</td>
</tr>
</tbody>
</table>

Grade Computation

<table>
<thead>
<tr>
<th>Drills</th>
<th>10%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lab Assignments</td>
<td>60%</td>
</tr>
<tr>
<td>Pre-labs</td>
<td>4%</td>
</tr>
<tr>
<td>Labs</td>
<td>48%</td>
</tr>
<tr>
<td>Post-labs</td>
<td>8%</td>
</tr>
<tr>
<td>Midterm Test</td>
<td>12%</td>
</tr>
<tr>
<td>Final</td>
<td>18%</td>
</tr>
</tbody>
</table>
Schedule, Readings, Lecture Notes & Recordings

Please refer to the detailed course schedule for suggested textbook readings, lecture notes, lecture and workshop recordings, as well as links to labs, drills, and any other coursework. You’ll want to bookmark the page.

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 labs (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.

Each week Prof Alexa will post 5-15 lecture-related “drill questions” on brightspace. These are meant to take 20-40 minutes per week, and provide technical practice with the newest concepts, or hone your intuition for the material. 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 programming lab. Also, they are similar to test questions—in fact, 40% of the midterm and final questions are drawn directly from the drills.

There are 12 drills (1 per week); you only have to do 8. The best 8 are worth 1.25% each; the remaining ~4 are dropped (or, more specifically, used as bonus.) Drills (on brightspace) are due by 3:00pm on Monday of the week following the associated lectures. Multiple attempts within the week are allowed; feedback is provided between attempts so that you can learn and improve between attempts.

Labs (60%)

The best computer scientists are the ones that have had the most 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. 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 labs, equally weighted. 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 one week before the programming portion
- 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 one week before the associated programming lab that encourages you to adopts 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.

Labs

This is the programming portion of the lab assignment, where you usually complete 5-10 parts. Each part 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 not use web search/the internet to look for approaches to the labs; you may not use AI services such as copilot to write any part of your code; you may not share or receive code; 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 best overall 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 the 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 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 labs and drills.
- They are the only coursework that evaluates you individually, since the remaining coursework can be (somewhat) collaborative.
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.

**Late Policy & Emergencies**

Drills, pre-labs, and post-labs must be submitted by the deadline, no lates accepted.

Programming assignments may be handed in up to 24 hours late without question, after which no lates are accepted, no exceptions (due to posted solutions.)

**In the case of an emergency**, you can, without explanation, not complete up to 4 drills, and 1 lab (pre-lab, lab, and post-lab combo). Please, please, please save this for emergencies such as personal or family illnesses. If you need to drop more than these 5 assessments then you should consider taking the course at another time when your life has less emergencies; there is no shame in that! You need to do a minimum amount of practice in this course in order to deeply engage with the material. For this reason, do not “squander” the emergencies for non-emergencies.

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

There are many opportunities for bonus points, if you have the time and inclination. Any remaining drills and labs outside of your best submissions (as described above) are available for bonus, as well as other opportunities throughout the semester. More information about bonus will be provided as the semester progresses.

**Bonus points are completely optional**; not doing bonus points will not negatively impact your final grade. Bonus points cannot add more than 3% to your final mark, and it cannot move you from an F to a passing grade.

**Learning Outcomes**

By engaging with the course material through 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.
Necessary Equipment, Accounts & Software

You will need an internet-connected device to access lectures and notes, to access brightspace for drills and tests, to access gradescope to submit your labs, and to access piazza and discord for Q&A and student hours.

You will need (free) accounts on piazza, gradescope.ca, and discord; please use your preferred (recognizable) name on all of these platforms. You will be authenticated on both.

Finally, you will need a java compiler and your favourite editor (preferably IntelliJ as that is what our course staff is most familiar with and can therefore help you with).

Late Registration

While the registrar will allow you to register for this course as late as Friday January 20th, we strongly discourage registration past January 13th, as it is difficult to miss more than a week of class and still catch up. We do not wait until January 20th 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.

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:

- 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)
  - good for questions that require more back-and-forth, that cannot be answered on piazza or discord
- Check the Q&A on discord (ideally after you’ve searched piazza to confirm it hasn’t been answered there.)

Finally, note that there might not be any official support on the weekends, so please plan your time accordingly.
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. Also note that there is a student (e.g. cmail) email account that does not reach Prof Alexa. For an email to be received, you must use the cunet email listed at the top of the Syllabus; even better, send a private post on piazza so you don’t have to worry about this confusing email situation. Prof Alexa tries not to work in the evenings and weekends so during those times please use a piazza post to Instructors if possible; otherwise, expect a response within 24h. If you do end up sending an email and you don’t hear back within 1-2 working days, please check you’ve used the correct cunet email.

Academic Integrity

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 write (i.e. problem solve) your code on your own. If your solutions are basically line-by-line the same as a peer's, that's 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 on the internet.

You must not use the internet to search for or solicit approaches or ideas.

You must not use AI programmers such as copilot for anything related to this course.

You must not post our lab specifications or solutions on the internet, before or after the due date.

Any violation of these rules is a very serious offence and will be treated as such. 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 blackmail and extortion.

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

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 co-operation or collaboration. Information on this policy may be found [here](#).

**Plagiarism.** As defined by 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". Please refer to the course outline statement or the instructor concerning this issue.

**Respect in the Classroom and Forums**

There is a zero-tolerance policy in this course for disrespect of any kind.

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, and you will be removed from course forums with no warnings.

We recommend you read over our piazza posting etiquette (pinned on piazza) as well as our discord #rules-please-read channel. You are responsible for behaving within these parameters.

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.)

**Pandemic Measures**

It is important to remember that COVID is still present in Ottawa. The situation can change at any time and the risks of new variants and outbreaks are very real. There are a number of actions you can take to lower your risk and the risk you pose to those around you including being vaccinated, wearing a mask, staying home when you’re sick, washing your hands and maintaining proper respiratory and cough etiquette.

**Feeling sick?** Remaining vigilant and not attending work or school when sick or with symptoms is critically important. If you feel ill or exhibit COVID-19 symptoms do not come to class or campus. If you feel ill or exhibit symptoms while on campus or in class, please leave campus immediately. In all situations, you must follow Carleton's [symptom reporting protocols](#).
**Masks:** Carleton has paused the COVID-19 Mask Policy, but continues to strongly recommend masking indoors, particularly if physical distancing cannot be maintained. It may become necessary to quickly reinstate the mask requirement if pandemic circumstances were to change.

**Vaccines:** Further, while proof of vaccination is no longer required as of May 1 to attend campus or in-person activity, it may become necessary for the University to bring back proof of vaccination requirements on short notice if the situation and public health advice changes. Students are strongly encouraged to get a full course of vaccination, including booster doses as soon as they are eligible, and submit their booster dose information in **cuScreen** as soon as possible. Please note that Carleton cannot guarantee that it will be able to offer virtual or hybrid learning options for those who are unable to attend the campus.

All members of the Carleton community are required to follow requirements and guidelines regarding health and safety which may change from time to time. For the most recent information about Carleton's COVID-19 response and health and safety requirements please see the [University's COVID-19 website](#) and review the [Frequently Asked Questions (FAQs)](#). Should you have additional questions after reviewing, please contact covidinfo@carleton.ca.

**Doctor's note or medical certificate:** in effect for Winter 2023 term. In place of a doctor's note or medical certificate, students are advised to complete the self-declaration form available on the Registrar's Office website to request academic accommodation for missed course work including exams and assignments. Students should also discuss with the course instructor the required accommodations arising from the COVID-19 situation.

**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 Prof Alexa can do to help you succeed, please let her know as soon as possible so that she can accommodate accordingly.

**Late Policy**

Please refer to the [Late Policy & Emergencies](#) section above.

**Copyright**

Lectures and course materials (including all notes, programs, labs, quizzes, handouts, videos, and similar materials) are protected by copyright. Prof Alexa is 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 reproduce or distribute lecture notes and
course materials publicly for commercial purposes, or allow others to, without 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.

More information about how Prof Alexa is trying to take some responsibility can be found on piazza in the post titled “Beyond the Land Acknowledgement.”

 Undergraduate Academic Advisors

The Undergraduate Advisors for the School of Computer Science are available in Room 5302HP; or by email at scs.uq.advisor@cunet.carleton.ca. The undergraduate advisors can assist with information about prerequisites and preclusions, course substitutions/equivalencies, understanding your academic audit and the remaining requirements for graduation. The undergraduate advisors will also refer students to appropriate resources such as the Science Student Success Centre, Learning Support Services and Writing Tutorial Services.

 University Policies

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

 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.