COMP 1805: Discrete Structures I
Early Summer 2021
Syllabus

Introduction to discrete mathematics and discrete structures. Topics include: propositional logic, predicate calculus, set theory, complexity of algorithms, mathematical reasoning and proof techniques, recurrences, induction, finite automata and graph theory. Material is illustrated through examples from computing. **Precludes** additional credit for MATH 1800. **Prerequisite(s)**: one Grade 12 university preparation mathematics course. **Minimum grade of C-** in COMP 1805 is required in order to take COMP 2804, COMP 3005, COMP 3007, or COMP 4001.

Instructor	Alexa Sharp (she/her)
Email	alexasharp3@cunet.carleton.ca
Lectures	Tuesday & Thursday 17:35PM - 20:25PM (recorded with live option)
Tutorials	Tuesday & Thursday 20:35PM - 21:25PM (recorded with live option)
Office Hours	Wednesday 13:00-15:00PM on discord, or by appointment
Course Website	piazza piazza.com/carleton.ca/summer2021/comp1805 brightspace https://brightspace.carleton.ca/d2l/home/53141
Q&A Forums	piazza (primary, structured), discord (non-anonymous, informal)

tl;dr

This course is meant to introduce you to the language of discrete mathematics. As with any language, you have to learn a lot of vocabulary and rules in order to speak the language, and then you have to practice, practice, practice. The lectures provide the vocabulary and rules, and the drills, tutorials, and problem sets provide the opportunity and incentive to get this practice with varying degrees of guidance. Every aspect of the course, every question, every evaluation, serves a purpose to get you to our final goal of you being able to fluently speak (discrete) mathematics, so that you can be a kick-ass problem solver. But you must put in the practice.

In a perfect world your course workflow would be:

- attend lecture to be exposed to the material (perhaps skimming the text in advance),
- do drills to practice the vocabulary, notation, and high-level concepts (e.g. what does that word or symbol mean again? When is this approach better than the previous one?),
- attend tutorials and complete accompanying exercises to get hands-on practice with some guidance (e.g. make one sentence at a time using new vocabulary and rules)
- complete problem sets to get more independent practice (e.g. write an essay using your new vocabulary and new ideas), reflect upon one's improvements in the self-evaluation,
- reference the textbook whenever more concrete information is needed.

Lectures

You may take this course completely asynchronously if you choose, with the option to attend some lectures (or parts of lectures) synchronously (live). Any synchronous lectures will be on Zoom, with recordings and transcripts provided within 12 hours. In live lectures, please mute your microphone, but turn on video if your internet connection and living arrangements can handle it. If you have questions, please ask in the chat. If you wish not to be recorded, you need to leave your camera and microphone turned off.

Textbook

We will use David Liben-Nowell's *Connecting Discrete Mathematics and Computer Science*. A free PDF version of the book is available at https://cs.carleton.edu/faculty/dln/book/.

Course Work & Evaluation

Drill Practice	**8%***	due the Monday following associated lecture 23:55pm
Tutorial Practice	**8%***	due the Monday following associated tutorial 23:55pm
Problem Set 1	12%	due Friday May 14 23:55PM
Problem Set 2	12%	due Friday May 21 23:55PM
Problem Set 3	12%	due Friday May 28 23:55PM
Midterm Test	12%	June 1 & 3 ("in class" and take-home, respectively)
Problem Set 4	12%	due Friday June 11 23:55PM
Problem Set 5	**12%***	due Friday June 18 23:55PM
Self-Evaluation	6%	due Friday June 18 23:55PM
Final Test	18%	TBD

*** best 4 problem sets, best 8 drills, best 8 tutorials count. Remaining used as bonus.**

Drills (8%)

If you're a professional soccer player, you prepare for matches both with practice scrimmages and with focused skills drills such as passing or shooting drills.

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

If you aspire to be a professional computer scientist or mathematician, you prepare for tackling new problems both with practice solving problems with known solutions, and with focused skills drills that target vocabulary, notation, and approaches you need to solve such problems and communicate your solution.

The problem sets in this course are an attempt to get you the former sort of practice--to simulate the kinds of mathematical problems you may be asked to solve as a computer scientist and mathematician. 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 4-10 lecture-related “drill” questions. These are meant to take a few minutes each, and provide technical practice with new definitions and notations, 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 problem set. Also, they are similar to short-form test questions---in fact, 20% of the (midterm and final) test questions are drawn directly from the drills (with only very minor adjustments).

There are ~13 drills (~1 per lecture). The best 8 are worth 1% each. The remaining drill grades will be dropped (or, more specifically, used as bonus). Drills are due by the Monday night following the associated lecture. Multiple attempts within the week are allowed.

Tutorials (8%)

While the drills provide specific, fine-tuned practice, they will not involve writing proofs. The tutorials guide you through solving a new problem, and ask you to practice on your own with low-stakes evaluation. As with the drills, 20% of the (midterm and final) test questions are drawn directly from the tutorials.

There are ~12 Tutorials (~1 per lecture, 1 used for the midterm test). The best 8 are worth 1% each. The remaining tutorial grades will be dropped (or, more specifically, used as bonus.) Tutorial videos will be 15-30 minutes long and will be available on Tuesdays and Thursdays. Tutorial exercises are due by the Monday night following the associated tutorial.

Problem Sets (48%)

The best computer scientists are the ones that have had the most practice. The problem sets 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. You should be just 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 and you will improve. The only way to really practice with the problem solving process is to experience it.

Note that 10% of each problem set is set aside for an optional self-evaluation. See the self-evaluation section below for more information.

There are 5 problem sets. The best 4 are worth 12% each. The lowest problem set grade will be dropped (or, more specifically, used as bonus.) **I recommend you "save" your dropped problem set for that emergency that is likely to crop up**. I will not be able to drop 2 problem sets. Problem sets are due on Fridays at 23:55pm on brightspace, no late submissions will be accepted.

Midterm & Final Test (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 hopefully improves your recall so that some of the more fundamental information can be recalled quickly.
- Tests encourage you to review all the course material, not just what is needed for the problem sets, tutorials, and drills.
- Tests are the only coursework that evaluates you individually, since the remaining coursework can be collaborative to some extent.

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

The midterm is worth 12% and the final 18%. The midterm will have 2 parts both on brightspace: the first part will consist of multiple-choice questions that must be answered during that day's tutorial (time-bounded by 60 minutes). The second part will consist of free-form questions that you will have 48 hours to work on. The final will be a combination of multiple-choice and free-form questions that must be answered on brightspace during our scheduled final time (TBD). The final is cumulative.

40% of both the midterm and final test questions will be drawn from drills and tutorials.

Self-Evaluation (a.k.a. Show Me Your Intangibles) (6%)

Disclaimer: this is a pedagogical experiment. I will always compute two grades for you: one with the experiment and one without, then I will assign you whichever is best. **The self-evaluations are meant to alleviate stress and anxiety, so if it causes you any, please just ignore them.**

Problem sets and tests generally evaluate you based on the extent to which you are able to complete assigned problems. When you are not able to complete a problem, your grade does not always reflect the partial learning you have achieved, or the skills that you have improved in attempting the problem. The self-evaluation is meant to be a place for you to reflect on what you have learned and what skills you have improved in attempting the assessment in question. It is a way for you to indicate your engagement with the material, even if you weren't able to "get" everything.

10% of each problem set, and 6% of your final grade can be based on self-evaluations.

To help you with your self-evaluation (since this is new to us all!) I will provide a list of possible skills to consider in each evaluation. You will be asked to choose a letter grade for your evaluation and write a few short sentences of justification. This grade may be higher or lower than what you end up getting on the remaining 90% of your problem set or 94% of your final grade. If you self-evaluate lower than your remaining grade, you will get your maximum grade. (If you self-evaluate higher, you still get the higher grade.) In this way, you are not ever penalized by participating in the experiment.

Note that I am not asking you to guess your problem set grade; rather, this is a way for you to adjust your problem set grade to better reflect the learning and practice you have achieved. I am trying this out because I have observed that often a student's grade does not reflect some of the intangible skills I witness them learning. Please try to be truthful; if everyone always gives themselves the top grade and there doesn't appear to be great self-reflection (it's only a few sentences folks!) then I will no longer use the experiment in subsequent semesters.

What am I hoping to get out of this? Firstly, I hope this gives you some sense of agency in your assessment. Secondly, I hope your final grades better reflect the learning and practice you have done throughout the course. Finally, I hope to more closely examine the discrepancies between self-assessments and non-self-assessment grades, to see if I can better design future coursework and assessments to narrow any gaps.

But remember: if you just give yourself 0% on all self-assessments, you will still get the same final grade as if we didn't have any self-assessments at all. So feel free to ignore them at any point, but I hope that you don't :-)

Bonus

There are many opportunities for bonus points, if you have the time and inclination. Any remaining problem sets, tutorials, and drills outside of your best submissions (as described above) are available for bonus.

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 overall in order to pass the course.)

Learning Outcomes

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

1. become comfortable reading and using mathematical terminology (e.g. vocabulary)
    - sets, functions, propositional and predicate logic, asymptotic notation, recursion, graphs

2. become comfortable reading and writing beautiful mathematical proofs
    - direct proofs, proofs by counterexample, proofs by construction, proofs by contradiction, proofs by contrapositive, proofs by induction
3. improve problem solving and critical thinking skills, such as
    - using examples, counter-examples, diagrams, simpler cases, similar problems, etc, to better understand a mathematical statement,
    - recognize a broken proof or a false start and using them to find a new result or approach,
    - thinking critically about which proof paradigm is most appropriate.

Necessary Equipment, Accounts & Software

You will need an internet-connected device to access lectures and slides, to access brightspace for drills, problem sets, and tests, and to access piazza and discord for Q&A and office hours.

You will need (free) accounts on piazza and discord; please use your preferred (recognizable) name on both of these platforms. You may optionally want (free) accounts on overleaf and anki for LaTeX typesetting and flashcards, respectively.

You don't need any special software for this course, but you need a way of typesetting mathematical symbols and producing a **pdf** document (Google docs, Microsoft Office, or LaTeX can all do this.) A scanned, photographed or non-pdf document will receive 0 marks.

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 relevant proof paradigm sheets (posted on piazza)
- Reference the problem solving tips sheet (posted on piazza)
- 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
- Go to Office Hours (posted on piazza)
    - 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.)

Academic Integrity

You **may** talk with peers and TAs at a high-level. But you must write up your solutions on your own. If your solutions are basically word-for-word the same as your peer's, that's working too closely.

You **must not** show or otherwise share your solution.

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

You **must not** post our problem sets or solutions on the internet, before or after the due date.

Students may collaborate on problem sets at the level of discussion, **but must write their solutions on their own**. Any students caught submitting copied solutions or overly sharing details of their solutions must 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.

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.

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.

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 reproduce or distribute lecture notes and course materials publicly for commercial purposes, or allow others to, 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.

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

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 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. You must also read the academic regulations of the university.

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/technical-support/. Technical support is available by emailing SCS.Tech.Support@cunet.carleton.ca.