# CGSC 1005A – Computational Methods in Cognitive Science
# DEPARTMENT OF COGNITIVE SCIENCE, CARLETON UNIVERSITY
# Fall 2024

| | |
|---|---|
| Instructor: | Kasia (Katarzyna) Muldner, Ph.D. |
| Office: | DT 2205 |
| E-mail: | kasia.muldner@carleton.ca |
| Office Hours: | TBA or by appointment |
| TAs: | TA1: TBA<br>TA2: TBA<br>TA3: TBA |
| TA Office Hours: | By appointment |
| Pass Coordinator: | TBA |

## Course Description

Introduction to computational methods, with an emphasis on programming. No prior programming background required. Programming is inherently a creative problem-solving activity that requires computational thinking. We will approach programming using the lens of cognitive science, namely that programming involves finding appropriate *representations* for the problem at hand and implementing *operations on those representations*.

On the representation side, we will cover *variables*, *standard data types (e.g., numeric, strings, Boolean)*, *data scope*, advanced data structures (e.g., lists). On the operation side, we will cover standard programming building blocks needed to create operations on these representations, including *iteration*, *conditional execution*, and *functional abstraction*.

The skills that we will emphasize during the class include *program design*, by discussing various algorithms and approaches to solving problems, *program implementation*, by writing Python code, and *debugging*, namely the ability to identify errors in programs through pattern matching and code tracing.

## Hardware + software requirements

All students taking CGSC 1005 are required to either own or have daily access to a computer (either Mac or PC; desktop or laptop) that has the programming language Python installed. ***Netbooks, Chromebooks, and Smartphones are not suitable*** – the computer must be running either Windows or a Mac operating system.

## Text book required

There is no required text book. A supplementary source that is freely available includes:

"*Python for Everybody: Exploring Data Using Python 3*", by Charles R. Severance . The textbook is freely available online under an open source license, and can be downloaded here:

http://do1.dr-chuck.com/pythonlearn/EN_us/pythonlearn.pdf

## Course Web Page (Brightspace)

The course website is located at https://carleton.ca/brightspace/ On this site you will find the course syllabus, slides, and any supplementary materials. Please note that the course slides are there for you to use and you are welcome and encouraged to do so, **but you are prohibited from sharing the slides.** My slides are my intellectual property, are Copyrighted and may not be shared or repurposed outside of this class. Sharing the slides either by electronic or non-electronic means is a violation of Copyright and I reserve the right to take action if you do so. For more information on Carleton's policy on Copyright infringement see: https://calendar.carleton.ca/copyrightcompliance/

## Evaluation

### Weighting of the Final Grade

| | | |
|---|---|---|
| **Assignments:** | **18%** | See table below for exact dates |
| **Tutorials:** | **10%** | Weekly – see table below for schedule |
| **Test 1:** | **20%** | October 1, in person during class |
| **Test 2:** | **20%** | November 5, in person during class |
| **Final Exam:** | **32%** | Regularly scheduled exam period, in person |

### Assignments

There are 5 assignments (see table below for due dates and weights). Unless otherwise stated, a**ll assignments are due by 9:30am on the specified date**. Please avoid putting the assignment off until the last minute as this does not work well with programming tasks. Note that the assignments are not weighted equally because they increase in difficulty.

| Assignment | Due Date | Weight |
|---|---|---|
| **A1** | Tuesday, Sept 24, 9:30am | 3% |
| **A2** | Thursday, Oct 10, 9:30am | 3% |
| **A3** | Thursday, Oct 31, 9:30am | 4% |
| **A4** | Thursday, Nov 21, 9:30am | 4% |
| **A5** | Thursday, Dec 5, 9:30am | 4% |

**ChatGPT Policy:** ChatGPT is a large language model capable of solving various tasks, including writing Python programs. In this class, you are allowed to use ChatGPT to help with assignments but there are number of important caveats:

- If you use ChatGPT for help with your assignment, **you must declare this in the assignment cover page**. This requirement is here to reflect general academic integrity (whenever an outside source is used for an assignment (in any class), this must be acknowledged). There is no downside for declaring use of ChatGPT but failure to do so is in violation of academic integrity.
- The tests and exams assess programming skills without access to ChatGPT. You will not learn how to program effectively if you ask ChatGPT to produce the answer right away without any reflection first. We will go over recommended ChatGPT usage strategies in class, but here are a few tips:
    - Instead of asking for help from ChatGPT right away, generate a plan first, and then write the actual program yourself. If you get stuck, use ChatGPT to help you debug your program, to provide explanations, etc. You can also use it to compare your program to its program
    - There is a wealth of research showing this strategy (*do it yourself first, ask for help only when really stuck*) results in better performance/learning than copying the answer (e.g., from ChatGPT) – I will share some of this research in class

- The prompts you give ChatGPT influence the quality of the answer it provides. We will go over prompt strategies in class.
- **Do NOT trust ChatGPT output!** It is not always correct and has produced conflicting and incorrect answers, even for basic Python programs. That's why it's so important to learn the foundations so that you can critically examine its output. In general, it is your responsibility to test and debug so that the program you submit is correct.
- This class allows the use of ChatGPT but not all classes do. It is your responsibility to check the syllabus for each class for the ChatGPT policy (in some classes, using it will correspond to an academic violation). If you are not sure, always check with your instructor.

**Plagiarism and collaboration policy**: The University Academic Integrity Policy defines plagiarism as "presenting, whether intentionally or not, the ideas, expression of ideas or work of others as one's own." In the context of this course,

- You **may** use ChatGPT to help with the assignments for this class but you must declare using it (you will not lose any points for this declaration but failure to declare could result in an allegation of plagiarism).
- You may collaborate with others at the **conceptual** level of assignments and tutorials (where conceptual is the algorithmic level, but NOT the code level). You may also collaborate at the code level with **one** other person – if you do, you must clearly indicate the name of the person you worked with on the assignment you pass in.
    - Group collaboration at the code level with more than 2 people is not permitted.
- Using code from another source (e.g., a person outside of this class, a human tutor, or a website) is considered plagiarism.
- If you are not sure whether something is allowed, please check in with your course instructor.

Per Carleton policy, suspected cases of plagiarism are forwarded to the Dean, who follows a rigorous process for such allegations, including reviewing documents and interviewing the student. Penalties for violations may include a final grade of "F" for the course.

**Late policy and extensions:** Assignments must be passed in on time – there is no grace period. This policy is to ensure that we can provide feedback in a timely manner. Late assignments will be accepted for up to 2 days after the posted deadline, but they incur a penalty of 10% for each 12 hours, as follows: 0-12 hours late = -10%; over 12 hours late up to 24 hours = -20%; over 24 hours up to 36 hours = -30%; over 36 hours up to 48 hours = -40%. If the submission is more than 2 days late (i.e., 48 hours) the assignment will be assigned a grade of 0. Technical problems such as connectivity issues do not exempt you from this penalty, so please don't wait till the last minute to submit. Some advice:

1. upload partially completed submissions as you work on them (you can submit multiple times)
2. submit the correct type of file (all the assignment require a python file, so please submit a file that ends in .py, not a word file or a PDF file; likewise pictures of a program will not be accepted)
3. aim to submit your final submission at least 30 minutes in advance of the due date and time
4. download your submission and verify the contents after submitting. Note: you have to open it using the programming environment app (e.g., IDLE) – do not just double click it as unless you have set up your system, that will result in the window rapidly opening and closing again.

**Extensions**: If you require an extension due to unforeseen circumstances, please email the course instructor **before** the assignment is due (requests after a deadline has passed will be denied).

**Issues with uploaded files.** You will be asked to upload python code, which must be a .py file, rather than a word/PDF document. It is your responsibility to ensure that the file you uploaded is correct (see point 4 above, i.e., after uploading, download to ensure the correct file(s) was uploaded). If you are not sure, please ask before the deadline – we are happy to help.

**Appeals:** Contact the TA that marked your assignment within one week of the date the assignment was returned (posted to Brightspace).

## Tutorials

Like the class, the tutorials will be conducted in person. The goal of the tutorials is to provide hands on practice on concepts discussed in tests, and thus solidify learning of those concepts. To receive a grade for a given tutorial, you must be present and work on the tutorial material. This is because the only way to learn programming is through repeated practice, and the tutorials provide that practice over and beyond assignments. However, some time will be provided in certain tutorials for assignment work. Each tutorial grade is out of 1, assigned as follows: 0 if did not show up or did not do any work during the tutorial, 0.5 if you came late and/or did some work, and 1 if came on time and worked on the tutorial materials the whole period.

* You have **one** "work from home" pass, allowing you to complete a tutorial at another time; the work must be submitted before the following week's tutorial by uploading it to Briightspace **and** emailing your TA to let them know. Note: (1) you must let you TA know **before** the tutorial if you plan to use this pass and (2) you are responsible for figuring out the tutorial material on your own.

## Tests + Exam

**Test 1:** The first test will cover content from the first portion of the course prior to test 1. More details will be provided before the test. The test will be conducted in person during class time.

**Test 2:** The second test will focus on content starting with material after test 1 up to the content covered prior to test 2, but it will also include concepts prior to test1 (since the nature of programming is inherently cumulative). The test will be conducted in person during class time.

**Final Exam:** The final exam (180 minutes) will be **cumulative**. More details will be provided closer to the exam date. The final exam will be conducted in person on the Carleton campus.

The tests and exam are based on: (1) course slides, which will be made available on Brightspace, (2) problem solving exercises done during tutorials and/or homework, (3) any assigned readings, (4) information presented during class, including class discussions. In particular, class attendance is strongly encouraged because **information will be provided during class beyond that in the textbook or slides**, including explanations of course themes, perspectives that are not in the textbook or in the slides, hints for tests, etc.

### Policies:

**Tests:** The tests must represent individual work – suspected cases of plagiarism will be forwarded to the Dean.

Illness and bereavement (supported by appropriate documentation – see Carleton procedures) are the only reasons accepted for missed tests – a zero grade will be assigned otherwise. If you miss a test for one of an accepted reason, please obtain the appropriate documentation, e.g., for illness, a self-declaration form (https://carleton.ca/registrar/cu-files/covid-19-self-declaration-form/) and contact me with the documentation **within 24 hours to arrange a make-up test**. Requests for a make up test made after this time window will not be accommodated.

A make-up test is only available for <u>one</u> of the two tests (either test1 or test2). If a make-up was written for test1, and test2 needs to deferred for medical or bereavement reasons, then its weight will be added to the final exam. Note this is the only situation in which reweighting the exam (to make it worth more) is possible. In particular, if you did not defer test 1 and could not write test2 for one of the accepted reasons, then you must write a make-up test for test2.

Appointments will be provided after the tests so you can review your test with the course instructor (and/or TA).

**Exam**: If you are unable to write the final exam, please follow the procedures listed in the section *Petitions to Defer*, located later in the outline.

**Bonus credits**. If there are any bonus opportunities, they will be announced during the course and will be available to all students. I sometimes get requests for individual bonus opportunities at the end of class to raise an individual's grade – these are not possible, because implementing them is not fair to the class a whole, so please plan accordingly.

**Final grade approval.** Standing in a course is determined by the course instructor subject to the approval of the Faculty Dean. This means that grades submitted by the instructor may be subject to revision. No grades are final until they have been approved by the Dean.

## E-mail Protocol and Guidelines

I will respond to e-mails within 24-48 hours (excluding weekends and holidays). Please do not send us code to be debugged over email unless you include a detailed code trace (details on what that involves will be provided in class) and a hypothesis for why it does not work. Often, if you have questions that require more than a yes/no type answer, the best forum for answering them is during class or office hours.

## Lecture Schedule

Please note that dates for topics are approximate and may change; weeks colored blue have an assignment due; yellow highlighting indicates a test:

| Lecture Date(s) | Lecture Topic | Tutorial Information |
|---|---|---|
| Week 1<br>Sept 5 | welcome and introduction | *No tutorials this week* |
| Week 2<br>Sept 10, Sept 12 | foundations (variable, data types)<br>conditionals | Tutorial 1 |
| Week 3<br>Sept 17, Sept 19 | conditionals continued<br>iteration | Tutorial 2 |
| Week 4 (A1 due, Tues)<br>Sept 24, Sept 26 | Iteration continued<br>review | Tutorial 3 |
| Week 5<br>Oct 1, Oct 3 | **Test 1**<br>iteration (for loops) | Tutorial 4 |
| Week 6 (A2 due, Thurs)<br>Oct 8, Oct 10 | files<br>lists | Tutorial 5 |
| Week 7<br>Oct 15, Oct 17 | lists + algorithms | Tutorial 6 |
| reading week, no classes or tutorials (Oct 21 - 25) | | |
| Week 8 (A3 due, Thurs)<br>Oct 29, Oct 31 | nested lists<br>review | Tutorial 7 |
| Week 9<br>Nov 5, Nov 7 | **Test 2**<br>more nested lists | Tutorial 8 |
| Week 10<br>Nov 12, Nov 14 | functions | Tutorial 9 |
| Week 11 (A4 due, Thurs)<br>Nov 19, Nov 21 | functions | Tutorial 10 |
| Week 12<br>Nov 26, Nov 28 | putting it all together (algorithms, functions, etc.) | Tutorial 11 |
| Week 13 (A5 due, Thurs)<br>Dec 3<br>Dec 5 | review<br>no class – extra office hours | *No tutorials* |

## Statement on Student Mental Health

As a University student you may experience a range of mental health challenges that significantly impact your academic success and overall well-being. If you need help, please speak to someone. There are numerous resources available both on- and off-campus to support you. For more information, please consult https://wellness.carleton.ca/

## Requests for Academic Accommodations

Carleton is committed to providing academic accessibility for all individuals. You may need special arrangements to meet your academic obligations during the term.

Details on accommodations for the  specific course components are above. Additional information on the accommodation request processes, including information about the Academic Consideration Policy for Students in Medical and Other Extenuating Circumstances, are outlined on the Academic Accommodations website (students.carleton.ca/course-outline).