COMP 1405T: Introduction to Computer Science Fall 2023

Contact

Instructor: Dr. David Mould
Email: mould@scs.carleton.ca
Office: 5346 Herzberg

Nature of the Course

Course calendar description: *Introduction to computer science and programming, for computer science students. Topics include: algorithm design; control structures; variables and types; linear collections; functions; debugging and testing. Special attention is given to procedural programming in a modern language, computational thinking skills, and problem decomposition.*

This is your first programming course, and there are no formal prerequisites beyond being enrolled in the program. However, we do expect that everyone brings a keen mind, an interest in the material, and a willingness to invest time.

This section will employ a *creative coding* ethos. We will use visual examples and build small interactive toys. We will use Processing, a creative coding environment based on Java.

You are expected to have **your own laptop**. More information can be found here:

`https://carleton.ca/scs/scs-laptop-requirement/`

Modality

This course will be held in person.

Topics

The course's main topics include the following:

- Programming and problem solving
- Drawing and interacting within Processing
- Variables and state
- Program building blocks: conditionals, iteration, functions.
- Testing and debugging
- Data structures: arrays and objects
- Data processing: visualization, synthesis, searching and sorting

Language

We will be using *Processing*, an environment for Java particularly oriented towards creating interactive visual applications. You can download Processing from processing.org. We will use Processing 4.3 in this offering.

Resources and Help

At minimum, you should be aware of these major resources:

- processing.org, which has a wealth of information in addition to letting you download the Processing software itself.
- the recommended textbook: *Processing*, by Nyhoff and Nyhoff.
- *Learning Processing*, by Daniel Shiffman, which provides an alternative perspective on the course material.
- the online course materials: examples, discussion forums, and ongoing news about the course as we progress through the term, accessible through Brightspace.

The School of Computer Science offers additional support. The **Undergraduate Advisor** for the School of Computer Science is available in Room 5302C HP, or by email at scs.ug.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.

Further, as an SCS student, you can access one of the **designated labs** for your course. The lab schedule can be found at:

`https://carleton.ca/scs/tech-support/computer-laboratories/`

All SCS computer lab and technical support information can be found at:

`https://carleton.ca/scs/technical-support/`

Technical support is available in room HP5161 Monday to Friday from 9:00 until 17:00 or by emailing SCS.Tech.Support@cunet.carleton.ca.

Tutorials

Tutorials run weekly, starting in the week of September 11. In each tutorial, you will have a small exercise or series of exercises to be done during the tutorial period. Attendance at your scheduled tutorial is mandatory.

Grading scheme

The marking breakdown is as follows:

- 10% total for tutorial exercises
- 25% total for weekly homework assignments
- 25% for the midterm exam
- 40% for final exam

Passing the course

This course has a "double pass" requirement, meaning that you must receive 50% or higher in both the exam portion (midterm+final exam) and the exercise portion (homework assignments+tutorial exercises). If either of these components is below 50%, you will receive a grade of F for the course.

Many computer science courses require a minimum grade of C- (60%) in the prerequisite, so do aim higher than a bare pass.

Learning Objectives

A student who effectively engages with all course material and completes all assigned exercises will be able to

- write computer programs in an imperative language.
- employ problem-solving strategies such as divide-and-conquer.
- trace and predict the behaviour of simple imperative computer programs.
- explain and apply variables, data types, conditionals, logical and comparative operators, iteration, and functions, in the context of an imperative programming language.
- be able to reason about and employ linear and multidimensional data structures.
- understand and implement basic searching and sorting algorithms.
- understand the notion of program efficiency and be able to compare the efficiency of different approaches to a problem.

On Collaboration

You are expected to do all assignments, exercises, and exams individually. This does not mean that you cannot talk to your classmates, though. Some examples of what is OK and not OK follow.
It is OK to:

- Discuss possible approaches to and interpretations of an assigned exercise.
- Help debug another student's program (in person, looking over their shoulder).
- Post questions on the class forums (without posting code).
- Answer questions on the forums (without posting code).

It is NOT OK to:

- Share working code or code fragments.
- Write code in groups and then share the finished code, or even parts of it.
- Post questions on forums besides our Brightspace or Discord forums.
- Post complete or partial assignment solutions on the forums.
- Claim credit for another student's work, e.g., by including code another student wrote as part of your submission.

Important dates

- Sep 6: Classes begin.
- Sep 11: Labs begin.
- Oct 20: Midterm exam.
- Oct 23-27: Fall break, no classes.
- Dec 8: Last day of classes; classes follow a Monday schedule.
- Dec ??: Final exam, scheduled centrally by the University; exact date not known at the time of writing. May be as late as Dec 22.

University Policies

Full academic regulations are found in the University's calendar, at calendar.carleton.ca. Some key information is excerpted below.

Requests for Academic Accommodation

You may need special arrangements to meet your academic obligations during the term. For details on the processes for academic accommodation requests, visit the following link:

`https://students.carleton.ca/course-outline/`

If you experience short-term incapacitation, e.g., arising from illness, please write to the professor to request accommodation for any missed work. A typical accommodation will involve either a short-term extension on the assignment, if practical, or shifting the weight of the assignment to the final exam.

Academic Integrity

Academic integrity is central to the University's mission. Allegations of academic dishonesty are taken seriously and in this course will be handled by the Office of the Dean of Science. A finding of academic dishonesty will result in sanctions, ranging from a grade of zero on the affected assignment or exam, to a failing grade in the course, and even suspension or expulsion from the university.

You can read about the policies of the Faculty of Science here:

`https://science.carleton.ca/academic-integrity/`

Additional information about academic integrity can be found here:

`https://carleton.ca/registrar/academic-integrity/`