Carleton UNIVERSITY | Department of Systems and Computer Engineering

ECOR 1041
Computation and Programming
Fall 2024

Instructional Support Coordinator (ISC)**:** Dr. Cristina Ruiz Martin (She/Her/Hers)

Sections A/B: Dr. Lynn Marshall (She/Her/Hers)

For students' hours and TA information check Brightspace.

Questions about the engineering content of this course ("programming questions") will not be answered by email. To get help from your instructor or a TA, you should ask questions:

- during lectures;
- during student hours (these are posted on Brightspace);
- during labs;
- by posting to the appropriate discussion forum on Brightspace.

All questions related to administrative matters must be sent to the Instructional Support Coordinator: ECOR1041@carleton.ca. Questions sent directly to your lecture instructor, or a TA will be discarded (or receive a reply telling you to email the ISC). Your email must be sent from your Carleton (cmail) email account. You will receive a reply from either the ISC, the lecture section instructor, or a TA.

Course Description and Requirements

1) Course Description

Software development as an engineering discipline, using a modern programming language. Language syntax and semantics. Tracing and visualizing program execution. Program style and documentation. Testing and debugging tools and techniques. Binary number system to represent data in a computer.

Lectures three hours per week, laboratories three hours per week.

http://calendar.carleton.ca/undergrad/courses/ECOR/

2) Prerequisites

This course may not be taken concurrently with ESLA 1300 or ESLA 1500. Precludes additional credit for COMP 1005, COMP 1405, ECOR 1051, ECOR 1606, SYSC 1005.

3) Prior Knowledge

No prior experience in computer programming is required. The only background we assume is:

- *language*: reasonable proficiency in reading and writing English;
- *math*: understanding of integers and operations on integers; understanding of functions as mappings from one set (the domain) to another set (the codomain);
- *logic*: familiarity with logical *and*, *or* and *not*;
- *computer literacy*: ability to use email, browse the World Wide Web, and edit textfiles.

4) Course Objectives

The objective of the course is to learn the fundamental concepts of computation and procedural programming, using Python as the programming language. Students will gain practical experience designing, coding, testing, and debugging small-scale Python programs. By the end of the course, students will have developed a "mental model" of computation and learned how to reason about and visualize the execution of program code. The course also introduces students to the internal representation of numeric values in computers, using the binary number system.

5) Accreditation Units

For more information about Accreditation Units, please visit: https://engineerscanada.ca/

The course has 27 AUs divided into:

Math	Natural Science	Complementary Studies	Engineering Science	Engineering Design
			100%	

6) Learning Outcomes / Graduate Attributes

By the end of this course, students should be able to:

- Understand the operations supported by Python types `int`, `float`, `str` and `list`. Evaluate Python expressions consisting of literal values, variables, and operators without the aid of a Python interpreter.
- Design and implement functions and interactive programs that satisfy given specifications. The code must demonstrate the proper use of the following fundamental constructs for controlling program execution: (1) sequential execution as determined by the order of executable statements; (2) selection (`if`, `if-else` and `if-elif-else` statements); (3) repetition (`for` and `while` statements); (4) calls to built-in functions, functions imported from modules and programmer-defined functions.

- Perform unit testing of functions using the Python shell and by writing simple test scripts.
- Use simple techniques to debug programs; for example, inserting print statements to instrument code and using a program visualization tool.
- Trace and visualize the execution of short Python programs. Explain what happens, step-by-step, as the computer executes each statement. Explain the diagrams produced by a program visualization tool that depicts the variables in the program's global frame, arguments and local variables in the function activation frames, and the objects that are bound to the variables.
- Represent unsigned and signed decimal integers as binary numbers, and interpret binary numbers as unsigned and signed decimal integers.

The Canadian Engineering Accreditation Board requires graduates of undergraduate engineering programs to possess 12 attributes. Courses in all four years of our programs evaluate students' progress towards acquiring these attributes. Aggregate data (typically, the data collected in all sections of a course during an academic year) is used for accreditation purposes and to guide improvements to our programs. Some of the assessments used to measure GAs may also contribute to final grades; however, the GA measurements for individual students are not used to determine the student's year-to-year progression through the program or eligibility to graduate.

This following list provides the GAs that will be measured in this course, along with the learning outcomes that are intended to develop abilities related to these attributes.

GA - Indicator	Assessment Tool
1.3 Knowledge base for engineering: Fundamental engineering concepts	All

7) Texts

Textbook

Practical Programming (Third Edition): An Introduction to Computer Science Using Python 3.6, Paul Gries, Jennifer Campbell, Jason Montojo, Pragmatic Bookshelf, 2017, ISBN-13: 978-1-68050-268-8.

Despite the phrase "Python 3.6" in the title, the book is equally applicable to the Python release we will be using.

A **free** digital copy of the book is available through O'Reilly (a digital library) and can be accessed by following these steps:

1. Go to https://library.carleton.ca/find/databases/oreilly
2. Click on the "Connect: O'Reilly" hyperlink
3. Sign in using your MyCarletonOne credentials.

4. Type “Practical Programming Jennifer Campbell" in the search box at the top of the page (to the right of the red O'Reilly logo), then tap the <Enter> key.
5. Scroll down until you find the link to "Practical Programming, 3rd Edition", then click the link.
6. Click the red Continue button below the picture of the book’s cover. You can then navigate to the Table of Contents to access any chapter.

Multiple eBook formats can be purchased directly from the publisher's website: pragprog.com. *If you choose to purchase the book,* m*ake sure that you buy the third edition, not the second edition.*

Software

We will be using Python 3 and the Wing 101 Integrated Development Environment (IDE). The software is free and available for the Windows, macOS, and Linux operating systems. We recommend that you replace any older releases of Python 3 on your computer with the newest release. Instructions on installing these will be provided in the first lab.

Students who prefer to use another IDE, such as PyCharm, VS Code, or Spyder, are free to do so, but the TAs or instructors will not provide support for technical issues. Note that some IDEs import libraries behind the scenes, and your code may throw an error when tested without the IDE in Gradescope.

We will also use Python Tutor (pythontutor.com), a free Web-based tool that helps us visualize what happens as the computer executes each line of a program's source code, step-by-step.

8) List of Topics

- Introduction to Python: using the Python shell as a calculator. Numeric types `int` and `float`. Operations supported by numeric types. Construction and evaluation of arithmetic expressions.
- The binary number system. Representation of integers and floating-point numbers in binary. Limitations of real-number representation in binary.
- Variables and assignment. Using variables in expressions.
- Introduction to Python Tutor to help us visualize and understand the execution of Python code.
- Using Python’s built-in functions (e.g., `print`, `abs`, `pow`, etc.). Importing functions from modules.
- Defining functions. Passing arguments to and returning values from functions. Using local variables in function definitions.
- Using a step-by-step recipe for designing functions: Function Design Recipe (FDR).

- Using the Python shell to test functions interactively. Writing simple test scripts to automate testing.
- Control flow and conditional execution: `if`, `if-else`, and `if-elif-else` statements.
- Boolean values (type `bool`). Boolean operators. Relational operators.
- Control flow and loops (repetition): `while` and `for` statements.
- Character strings (type `str`).
- Getting input from the user: the input function. Interactive programs. Try/Except.
- Lists of data (type `list`). Using a for loop to iterate over a list.

9) Course Schedule

Week-by-Week (Tentative)

Lecture	Topic
1	Introduction to Python: using the Python shell as a calculator. Numeric types `int` and `float`. Construction and evaluation of arithmetic expressions.
2	Representation of numbers in the binary number system. Limitations of real-number representation in binary.
3	Variables and assignment. Using variables in expressions. Python Tutor for tracing and visualizing Python code execution. Coding conventions.
4	Using Python's built-in functions (e.g., `print`, `abs`, `pow`, etc.). Importing functions from modules.
5	Defining functions. Passing arguments to and returning values from functions. Using local variables in function definitions. Testing functions: Using the shell to test functions interactively.
6	A step-by-step recipe for designing functions: Function Design Recipe (FDR).
7	Control flow. Conditional (`if`) statements.
Midterm	**Midterm (During lecture time)**
8	Boolean values (type `bool`) and operators. Relational operators. Automated testing
9	Control flow: Loops (`while` and `for` statements).
10	Character strings (type `str`) Getting input from the user. Interactive programs (i.e., Python scripts). `Try/Except`
11	Introduction to lists: Python's `list` type. Using a `for` loop to iterate over a list.
12	Review

10) Evaluation and Marking Scheme

Students will be evaluated using online quizzes, laboratory work, a midterm exam, and a final exam. A numeric mark out of 100 will be calculated by weighing the course components as shown in this table:

Component	**Weight**
Quizzes (best 5 out of 6)	5%
Labs (best 5 out of 6)	10%
Bonus: Lab 7	2%
Bonus: Class Participation	3%
Midterm	25%
Final exam	60%

This mark will be converted to a letter grade (using the table of percentage equivalents shown in the *Undergraduate Calendar, Academic Regulations of the University, Section 5.4, Grading System)* as long as the student meets the requirements stated below. Otherwise, the student's grade will be an F*.*

Requirement for all students: To pass the course (i.e., D- or higher), students must achieve at least 25/60 on the final exam.

Requirements for students writing the deferred final exam: Students who are approved for a deferred final exam by the Registrar's office must have achieved the grades below on the listed course components to be eligible to pass the course:

- Labs (including Lab 7): 8/12
- Quizzes: 2/5
- Midterm: 10/25

For the above calculation, if a student defers the midterm, the grade in the midterm will be considered 0/25. Thus, the final grade for students deferring both the midterm and the final exam will be an F, regardless of their grade on the deferred exam.

A grade of at least C- in ECOR1041 is a pre-requisite for ECOR1042. ***This pre-requisite will be enforced.***

If you do not achieve a grade of C- in ECOR1041, you will be ***removed automatically from ECOR1042 (late Fall) and automatically re-enrolled in a new ECOR1041 section****.* ***Do not drop*** *ECOR1042 if you fail ECOR1041. Otherwise, you will not be re-enrolled in ECOR1041 (late Fall).*

If you do not want to be re-enrolled in ECOR1041 for late Fall in the case that you do not achieve a grade of C-, you must email ECOR1041@carleton.ca before 20th October 2024. *Note that the pre-requisite will be enforced regardless of your decision to be re-enrolled in ECOR1041 or not, and you will have to re-take ECOR1041 at a later time (Winter or Summer) before being able to take ECOR1042.*

Note: Second-year status is a prerequisite for the second-year engineering, science, and mathematics courses that are part of your program. One of the requirements for achieving second-year status is a minimum grade of C- in each of ECOR 104x courses, including ECOR1041.

Breakdown of course requirements

Attendance

Students are expected to attend all lectures and lab periods. The University requires students to have a conflict-free timetable. For more information, see the current *Undergraduate Calendar, Academic Regulations of the University, Section 2.1.3, Course Selection and Registration and Section 2.1.7, Deregistration.*

Lectures & Lecture Participation (3% Bonus)

Lectures will be delivered in person. Lectures will not be streamed or recorded.

Participation during lectures is worth a 3% (bonus). To complete this course component, students need to be present in the lecture hall. No accommodations will be made for missed lectures. Students who miss all lectures can still achieve 100% in the course.

Note: Participating in the lecture polls while not present in the lecture hall will be considered an academic offense, and will be reported to the Dean's office. In this case, the grade for the lecture participation component will be zero no matter the circumstances.

Quizzes (5%)

Reading assignments from the textbook will be posted on Brightspace. Short online multiple-choice quizzes related to the reading assignments will be posted. The best 5 out of 6 quizzes will count toward your grade. All quizzes are equally weighted. You will need to complete each quiz by the due date listed on Brightspace. **Extensions will not be granted.**

Labs (10%)

Labs consist of short programming exercises that are intended to help you understand concepts that have been introduced in the lectures. The best 5 out of 6 labs will count toward your grade. All labs are equally weighted. Labs must be submitted to Brightspace for grading by the due dates. **Extensions will not be granted.**

If your code does not run on Gradescope or has a run time error, you will receive 0 for your submission.

Lab 7 (2% Bonus)

Lab 7 is an extra lab reviewing the last lectures of the course. Despite being a bonus lab in terms of grading, the material covered is part of the course, and it will be covered in the final exam. **Extensions will not be granted.**

Exams: Midterm (25%) and Final (60%)

The midterm and final exams will be held in person on Campus to assess your individual learning. Exams will be written on paper. Each exam will cover **all material** covered up to that point in the course. Both the midterm and final exams will be **closed-book and proctored**. No calculators will be allowed. You will be expected to complete the exams on your own, without any support from or communication with others. Detailed instructions about the midterm and final exams will be posted on Brightspace.

Requests to increase the weight of the final exam because of poor performance in the midterm, assignments, or labs, with a corresponding reduction in the weight of the other component(s), will not be considered.

Final exams are for evaluation purposes and will not be returned/discussed with students.

Missed Term Work

Students who claim short-term extenuating circumstances (normally lasting up to five days) as a reason for missed term work are held responsible for **immediately** informing the instructor (with a **completed self-declaration** form) and for making alternate arrangements with the instructor. In all cases, this must occur no later than three (3) days after the term work is due. The alternate arrangement must be made before the last day of classes in the term as published in the academic schedule. Consult the Section 4.4 of the University Calendar:

https://calendar.carleton.ca/undergrad/regulations/academicregulationsoftheuniversity/examinations/#deferred-term-work

For labs and quizzes, **extensions will normally not be granted.** They are posted in advance, and students have a long period of time to work on them. Normally, requests after the submission deadline will be denied.

Students who are unable to write the midterm exam because of illness or other extenuating circumstances beyond their control can request to get the weight of the midterm added to the weight of the final exam. These requests must be made no later than 3 days after the exam date and must be fully supported by appropriate documentation. These students can also request permission to write the midterm at a later date. These exams will be graded to provide feedback, but the mark will not be used when calculating the final grade.

Missed Final Exam

Students who are unable to write the final examination because of a serious illness/emergency or other circumstances beyond their control may apply for accommodation by contacting the Registrar's office. Consult the Section 4.3 of the University Calendar:
https://calendar.carleton.ca/undergrad/regulations/academicregulationsoftheuniversity/examinations/

Copyright

The materials (including the course outline and any slides, posted notes, videos, labs, project, assignments, quizzes, exams and solutions) created for this course and posted on this web site are intended for personal use and may not be reproduced or redistributed or posted on any web site without prior written permission from the author(s).

Advising and Counselling Services

a) Engineering Academic Advising

The Engineering Academic Support Service: https://carleton.ca/engineering-design/current-students/undergrad-academic-support/ assists undergraduate engineering students with course selection, registration, and learning support from first-year through to graduation.

Academic Advisors Contact: https://carleton.ca/engineering-design/current-students/undergrad-academic-support/undergraduate-advisors/

b) Student Mental Health Service

As a University student, you may experience a range of mental health challenges that significantly impact your academic success and overall well-being. Carleton's Wellness Services Navigator https://wellness.carleton.ca/navigator/ is designed to help students connect with mental health and wellness resources. If you need to talk to someone, please reach out for assistance: https://carleton.ca/health/emergencies-and-crisis/

Learning and Working Environment

The University and all members of the University community share responsibility for ensuring that the University's educational, work and living environments are free from discrimination and harassment. Should you have concerns about harassment or discrimination relating to your age, ancestry, citizenship, colour, creed (religion), disability, ethnic origin, family status, gender expression, gender identity, marital status, place of origin, race, sex (including pregnancy), or sexual orientation, please contact the Department of Equity and Inclusive Communities at equity@carleton.ca

We will strive to create an environment of mutual respect for all through equity, diversity, and inclusion within this course. The space which we work in will be safe for everyone. Please be considerate of everyone's personal beliefs, choices, and opinions.

Academic Integrity and Plagiarism

a) Please consult the Faculty of Engineering and Design information page about the Academic Integrity policy and our procedures: https://carleton.ca/engineering-design/current-students/fed-academic-integrity. Violations of the Academic Integrity Policy will result in the assignment of a penalty such as reduced grades, the assignment of an F in a course, a suspension or, expulsion.

b) One of the main objectives of the Academic Integrity Policy is to ensure that the work you submit is your own. As a result, it is important to write your own solutions when studying and preparing with other students and to avoid plagiarism in your submissions. 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." This includes reproducing or paraphrasing portions of someone else's published or unpublished material, regardless of the source, and presenting these as one's own without proper citation or reference to the original source.

Examples of violations of the policy include, but are not limited to:

- Any submission prepared in whole or in part, by someone else;
- Using another's data or research findings without appropriate acknowledgment;
- Submitting a computer program developed in whole or in part by someone else, with or without modifications, as one's own;
- Failing to acknowledge sources of information through the use of proper citations when using another's work and/or failing to use quotations marks; and
- Unless explicitly permitted by the instructor in a specific course, the use of generative AI and similar tools to produce assessed content (such as text, code, equations, images, summaries, videos, etc.).

Academic Accommodations

You may need special arrangements to meet your academic obligations during the term. For an accommodation request the processes are as follows:

Pregnancy obligation: Contact us with any requests for academic accommodation during the first two weeks of class, or as soon as possible after the need for accommodation is known to exist. For accommodation regarding a formally-scheduled final exam, you must complete the Pregnancy Accommodation Form (click here).

Religious obligation: Contact us with any requests for academic accommodation during the first two weeks of class, or as soon as possible after the need for accommodation is known to exist. For more details click here.

Academic Accommodations for Students with Disabilities: The Paul Menton Centre for Students with Disabilities (PMC) provides services to students with Learning Disabilities (LD), psychiatric/mental health disabilities, Attention Deficit Hyperactivity Disorder (ADHD), Autism Spectrum Disorders (ASD), chronic medical conditions, and impairments in mobility, hearing, and vision. If you have a disability requiring academic

accommodations in this course, please contact PMC at 613-520-6608 or pmc@carleton.ca for a formal evaluation. If you are already registered with the PMC, contact your PMC coordinator to send us your Letter of Accommodation at the beginning of the term, and no later than two weeks before the first in-class scheduled test or exam requiring accommodation (if applicable). After requesting accommodation from PMC, contact us, if needed, to ensure that accommodation arrangements are made.

You should request your academic accommodations in the Ventus Student Portal, for each course at the beginning of every term. For in-term tests or midterms, please request accommodations at least two (2) weeks before the first test or midterm.

Please consult the PMC website for the deadline to request accommodations for the formally-scheduled exam (if applicable).

Survivors of Sexual Violence: As a community, Carleton University is committed to maintaining a positive learning, working and living environment where sexual violence will not be tolerated, and where survivors are supported through academic accommodations as per Carleton's Sexual Violence Policy. For more information about the services available at the university and to obtain information about sexual violence and/or support, visit: https://carleton.ca/equity/sexual-assault-support-services

Accommodation for Student Activities: Carleton University recognizes the substantial benefits, both to the individual student and for the university, that result from a student participating in activities beyond the classroom experience. Reasonable accommodation will be provided to students who compete or perform at the national or international level. Contact us with any requests for academic accommodation during the first two weeks of class, or as soon as possible after the need for accommodation is known to exist: https://carleton.ca/senate/wp-content/uploads/Accommodation-for-Student-Activities-1.pdf