

---

**1. Course Information**

|                       |                                       |                    |  |
|-----------------------|---------------------------------------|--------------------|--|
| <b>Instructor:</b>    | Dr. Christine Laurendeau (she/her)    | <b>Lectures-A:</b> | Tue. and Thu. 10:05 - 11:25 am             |
| <b>Email:</b>         | ChristineLaurendeau@cunet.carleton.ca | <b>Lectures-C:</b> | Mon. and Wed. 1:05 - 2:25 pm               |
| <b>Office:</b>        | HP 5320 / HP 5360                     | <b>Tutorials:</b>  | posted in <a href="#">Carleton Central</a> |
| <b>Office hours:</b>  | posted in <a href="#">Brightspace</a> | <b>Classrooms:</b> | posted in <a href="#">Carleton Central</a> |
| <b>Teaching team:</b> | posted in <a href="#">Brightspace</a> | <b>Web site:</b>   | <a href="#">Brightspace</a>                |

**Land acknowledgement:** Carleton University acknowledges the location of its campus on the traditional, unceded territories of the Algonquin nation.

---

**2. Course Description**

Introduction to system-level programming with fundamental OS concepts, procedures, primitive data types, user-defined types. Topics may include process management, memory management, process coordination and synchronization, inter-process communication, file systems, networking, pointers, heap and stack memory management, and system/library calls. Precludes additional credit for SYSC 2006.

Prerequisite(s): (COMP 1006 or COMP 1406 or SYSC 2004) with a minimum grade of C-.

---

**3. Topics Covered and Learning Outcomes**

The course will cover the following topics, although some material may be omitted due to time constraints:

- Introduction to computer systems
- Data representation: primitive data types, compound data types, pointers
- Memory management: stack and heap, dynamic memory allocation, linked lists
- Program building
- Concurrent computing: concurrent systems, process management, IPC, threads
- Input/Output, program organization, graphics libraries and shell scripts

Once a student successfully completes the course, they will be able to:

- implement intermediate-novice level programs using the C language in a Unix-based environment
- program in accordance with a procedural design, using encapsulated, modular functions
- program using explicit memory management techniques, including usage of the function call stack and the heap, with programmer-managed dynamic memory allocation and deallocation

---

**4. Textbook(s)**

M. Lanthier, C. Laurendeau, and D. Nussbaum, *COMP2401 Course Notes*. [Online].

Available: <https://www.scs.carleton.ca/~clarend/Courses/COMP2401/Notes>

---

**5. SCS Laptop Requirement**

Every student who 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/>.

## 6. Assessment Scheme

---

6.1. Students will be assessed in this course according to the following measures:

| Component                        | Weight | Due dates                                |
|----------------------------------|--------|--|
| Assignments                      | 40%    | Oct. 5, Oct. 19, Nov. 9, Nov. 23, Dec. 7 |
| Tutorials (best 8 out of 10)     | 8%     | Weekly                                   |
| Progress check mini-quizzes (10) | 2%     | Weekly                                   |
| Midterm                          | 15%    | Section C: Nov. 1<br>Section A: Nov. 2   |
| Final exam                       | 35%    | TBA                                      |

6.2. The assessment scheme in paragraph 6.1 will be in effect as long as the course remains in the in-person format for the duration of the term. If courses are shifted to online modality by the University at any point during the term, a new assessment scheme will be provided.

6.3. **Weighting of assignments:** Only four (4) of the five (5) assignments will count towards the final grade, and each is worth 10%. The best two (2) of the first three (3) assignments will be counted for the final grade, and so will both Assignments #4 and #5. Neither Assignment #4 nor #5 will be waived. No additional assignment will be waived, for any reason.

## 7. Assessment Notes

---

7.1. All assignments, tutorials, and all course work must be completed **individually**. Collaborating on any course work is strictly disallowed and will be reported as an academic integrity offence.

7.2. In addition to the time spent attending lectures and tutorials, students can expect to spend at least nine (9) hours per week on this course.

7.3. Students **must** write the midterm in the section (A or C) in which they are registered. Students who write the midterm in the wrong section will receive a midterm grade of zero.

7.4. It is the student's responsibility to ensure that their midterm, tutorial, and assignment marks posted in *Brightspace* are correct. All marking disputes must be addressed with the individual responsible for marking the work (TA or instructor), **within one week** of the marks being posted. In cases where a student and a TA cannot agree, the matter will be referred to the instructor for resolution. For course work that is due close to the end of the term, the dispute period may be shortened to allow for the timely submission of final grades.

7.5. Technical problems do not exempt students from any submission requirement. If students wait until the last minute and then have issues with their computer or internet connection, their submission will still receive a mark of zero.

7.6. There will be no extra credit available in this course.

## 8. Course Material

---

8.1. All concepts covered during the lectures and during tutorials are part of the course material, including the course notes and annotations, all in-class coding examples, tutorial exercises, and in-class and forum discussions.

8.2. Lecture recordings may be provided, but **exclusively** as a supplemental study aid. They are not a substitute for lecture attendance and note taking. *Some lectures may not be recorded, and some recordings may not be available*, for either technical or pedagogical reasons. Students are responsible for learning the material covered during all lectures, whether recordings are available or not.

8.3. All materials created for this course (including, but not limited to, course notes, coding examples, lecture recordings, tutorial specifications, tutorial code bases, assignment specifications, assignment code bases, project specifications, project code bases, marking schemes, midterms, exams, and midterm and exam solutions), except where otherwise noted, remain *the intellectual property of the instructor*. They are intended for the personal and non-transferable use of students registered in the course. Reproducing, reposting, and/or redistributing any course materials, in part or in whole, without the written consent of the instructor, is a violation of IP rights, and is **strictly prohibited**.

## 9. Assignments

---

- 9.1. There will be five (5) assignments in this course. The best two (2) of the first three (3) assignments will count towards the final grade, as will both of the last two assignments. Assignment requirements will be posted in *Brightspace*.
- 9.2. To ensure that the learning outcomes of the course are met, assignment work must be completed **strictly** according to the provided instructions and constraints, using only the library functions and programming techniques covered in the in-class examples. Code that undermines the learning outcomes of the course will not earn marks. Students are expected to seek clarifications by attending the instructor's office hours, or by posting in the corresponding forum in *Brightspace*, as needed.
- 9.3. All assignments **must** be completed within the Virtual Machine (VM) provided for the course. The provided VM is the *only* programming environment in which course work will be graded.
- 9.4. All assignment code submitted for credit, with the exception of base code provided by the instructor, must be *original*, and the student submitting the assignment code must be its sole author.
- 9.5. **Late penalty:** Late assignments will incur a deduction of 5 marks (out of 100) for every hour late, or part of an hour late, up to a maximum of 10 hours past the submission deadline. Once this 10-hour time window has elapsed, the *Brightspace* submission link will expire, and no submissions, substitutions, or corrections will be accepted, for any reason.
- 9.6. **Extensions:** Students may request a 72-hour deadline extension for a maximum of one (1) assignment during the term. Extension requests must be submitted **before** the original due date for the assignment, using the online form provided in *Brightspace*. No additional extensions will be granted, for any reason. Extension requests received after the assignment deadline will automatically be denied.
- 9.7. Only assignment files uploaded into *Brightspace* will be graded for credit. Students are responsible for the integrity of their assignment submissions. Submissions that contain incorrect, corrupt, or missing files may receive a grade of zero, in accordance with the marking scheme. Corrections to submissions will not be accepted after the submission link expires. **You must verify that your submission is correct and complete** by re-downloading it from *Brightspace*, uncompressing it into a fresh directory in the VM, and compiling and running the code.
- 9.8. The **only** valid reason to appeal an assignment grade is an error by a TA in applying the grading scheme. Student errors, including but not restricted to submitting a wrong or corrupted file, or submitting code that doesn't compile or doesn't run, are **not** a basis for appealing a grade. All appeals of this nature will automatically be denied.
- 9.9. Students are expected to make regular backups of their work, to a file system outside the course VM, at least once for every hour of work. No accommodations can be made if submission files get overwritten or corrupted, or if the VM stops working.
- 9.10. Assignment marks will be released to students when **all** the grading is completed.
- 9.11. Prior to each assignment due date, the instructor will hold a 30-40 minute in-class workshop to discuss the assignment requirements and to answer student questions. The workshops dates will be posted in *Brightspace*.

## 10. Tutorials

---

- 10.1. Tutorial attendance:
  - 10.1.1. Tutorials begin on Sept. 12. The complete schedule is posted in *Brightspace*.
  - 10.1.2. There will be ten (10) tutorials. Of those 10, the best eight (8) will count towards the final grade.
  - 10.1.3. You **must** attend the tutorial session *for which you are registered*. We are unable to accommodate requests to attend alternate sessions.
  - 10.1.4. Tutorials will not be posted in advance, for any reason.
- 10.2. Tutorial grading is at the discretion of the lab coordinator and TAs, and is not negotiable. Tutorial grades are for attendance, working on the tutorial questions for the entire session, and answering TA questions about your work.
- 10.3. Tutorial work that is started or completed before your session will earn zero marks.

## 11. Progress Checks

---

- 11.1. Progress checks are weekly mini-quizzes used to regulate student progress through the course material. There will be a total of 10 progress checks, and they will be available in *Brightspace*.
- 11.2. The weekly progress checks begin on Sep. 15 and end on Dec. 3, excluding the week of Oct. 16 and the week of the Fall Break.
- 11.3. Each week, the progress check will be open for three (3) days, beginning on the Friday morning at 12:01 am (midnight) and closing on the Sunday night at 11:59 pm. Students must complete the progress check within that time period.
- 11.4. Each progress check will assess student understanding of the concepts covered during the lectures of that week, and possibly the previous week.
- 11.5. Each progress check will consist of five (5) questions, and it will earn a Pass/Fail grade. Students must correctly answer four (4) of the 5 questions to get a passing grade.
- 11.6. Each progress check will allow for three (3) attempts at earning a passing grade.
- 11.7. Computation of the final progress check grade (out of 2% of the final grade):
  - 11.7.1. If a student earns a passing grade on 9 or 10 progress checks, they get full points (2 out of 2) for their progress check grade.
  - 11.7.2. If they pass 5 to 8 progress checks, inclusively, they earn one point (1 out of 2).
  - 11.7.3. If they pass 0 to 4 progress checks, inclusively, they earn zero points.

## 12. Collaboration Policy

---

- 12.1. Collaborating on any course work, including but not restricted to assignments, projects, tutorials, mid-terms, and final exams, is **strictly disallowed** and will be reported to the Dean of Science as an academic integrity offence. Penalties for such offences can be found on the [ODS web page](#). Students must complete all course work by themselves.
- 12.2. Examples of academic integrity offences include: emailing code to other students; uploading code to a web site, at any time; copying code from any sources, even cited ones; working with other students; getting help from anyone other than the course TAs or the instructor; submitting code, or portion thereof, written by anyone other than the student submitting the work.
- 12.3. If you are unsure of the expectations regarding academic integrity (how to use and cite references, if collaboration with lab- or classmates is permitted, and if so, to what degree), then you must ASK your instructor. Sharing assignment or quiz specifications or posting them online (to sites like Chegg, CourseHero, OneClass, etc.) is ALWAYS considered academic misconduct. You are NEVER permitted to post, share, or upload course materials without explicit permission from your instructor. Academic integrity offences are reported to the office of the Dean of Science. Information, process and penalties for such offences can be found on the [ODS web page](#).
- 12.4. All assessed activities in this course are designed to be completed by an individual working alone. The use of any artificial intelligence tool is strictly prohibited and will be reported as academic misconduct. This includes, but is not limited to, ChatGPT, Copilot, etc.
- 12.5. Posting course work and/or its solutions online, including assignment work, project work, tutorial work, midterm work, and final exam work, and distributing course work and/or solutions to other students **at any time** is strictly prohibited and will be reported to the Dean of Science as academic misconduct. This includes work publicly posted on source control sites like GitHub.
- 12.6. Posting course work after the conclusion of the course is also strongly discouraged, as it is of no benefit to anyone other than future students looking to cheat. Employers want to see evidence of candidates' creativity and initiative, neither of which is demonstrated in 2nd year course work. Coming up with your own creative and original project ideas, and completing these projects on your own time is the best recipe for impressing potential employers.

### 13. Communications Policy

---

- 13.1. Students are expected to check their email on a **daily** basis. Important course-related announcements will be posted in *Brightspace* and forwarded to students' email accounts.
- 13.2. Due to a high volume of emails, the instructor will respond to student emails within 2 to 3 [business days](#). This timeframe excludes weekends, statutory holidays, and other days when the University is closed. Emailed questions that request information already available in a discussion forum, or in an assignment specification, or in the course outline may take longer.
- 13.3. Students are asked to post all course-related and assignment-related questions in the corresponding discussion forum in *Brightspace*. Please verify whether the question has already been answered. If not, the question must be posted in a solution-free manner in the appropriate forum, where it will be answered.
- 13.4. **TA office hours** are the first point of contact for students requiring help with debugging their code. Please note that TAs are not experts in the course material or in the assignment requirements. For questions about course work requirements, please see the instructor during office hours or post the questions in the appropriate *Brightspace* forum, where the instructor will answer and clarify.
- 13.5. The **lab coordinator** is the first point of contact for students requiring help with all matters related to tutorials.
- 13.6. **Instructor office hours** are the first point of contact for students requiring help with the course material, or with understanding assignment requirements, or for academic advising. The instructor will be happy to assist with debugging during office hours as well, however the time available to each student will be limited if a large number of students are seeking help at the same time.
- 13.7. In case of technical issues with the installation or operation of the provided VM, students are required to first [read the documentation](#) provided by the SCS technical staff. Additional assistance may be provided by the course TAs.
- 13.8. Student emails to the TAs, the lab coordinator, and/or the instructor **must** indicate the course code and section in the subject line. Their tone and content must be *professional*, and not personal, in nature. Specifically, they must be written as to a colleague or co-worker, not as to a family member or friend.
- 13.9. Students are expected to behave and communicate in a **courteous** and **professional** manner at all times. Any communications, either in person, or online in forum posts and email, that do not follow the basic precepts of common courtesy and professionalism will not be answered, and in extreme cases will be reported to university authorities. Carleton University's expectations of student behaviour online can be found at [this link](#).

### 14. Undergraduate Academic Advisor

---

The Undergraduate Advisor for the School of Computer Science is available in Room 5302 HP; or by email at [scs.ug.advisor@cunet.carleton.ca](mailto: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 the Writing Tutorial Services.

### 15. SCS Computer Laboratory

---

Students taking a COMP course can access the SCS computer labs. The lab schedule and locations can be found at [this link](#). All SCS computer lab and technical support information can be found at [this link](#). Technical support staff may be contacted in-person or virtually, see [this link](#) for details.

### 16. University Policies

---

**Academic Calendar.** For information about Carleton's academic year, including registration and withdrawal dates, see Carleton's [Academic Calendar](#).

**Academic Accommodations.** Carleton is committed to providing academic accessibility for all individuals. Please review the academic accommodation available to students at [this link](#).

**Student Academic Integrity Policy:** 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 suspended or expelled from the University. Examples of punishable offences include plagiarism and unauthorized collaboration. Any such reported offences will be reviewed by the office of the Dean of Science. More 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. More information and standard sanction guidelines can be found [here](#).

**Unauthorized 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".

**Unauthorized Student Recordings and Use of Instructor Recordings:** Unauthorized student recording of classroom or other academic activities (including advising sessions or office hours) is prohibited. Unauthorized recording is unethical and may also be a violation of University policy. Students requesting the use of assistive technology as an accommodation should contact the [Paul Menton Centre](#). Unauthorized use of classroom recordings – including distributing or posting them – is also prohibited. Under the University's [Copyright Policy](#), faculty own the copyright to instructional materials – including those resources created specifically for the purposes of instruction, such as lectures slides, lecture notes, and presentations. Students cannot copy, reproduce, display, or distribute these materials or otherwise circulate these materials without the instructor's written permission. Students who engage in unauthorized recording, unauthorized use of a recording, or unauthorized distribution of instructional materials will be referred to the appropriate University office for follow-up.