Carleton University
School of Computer Science

COMP 2401B
Intro to Systems Programming

Course Outline (W2025)

Course Information:

Instructor: Mark Lanthier (MarkLanthier@cunet.carleton.ca)

Lectures: Mondays & Wednesdays **4:05pm** – 5:25pm
Tutorials: May be done at any time.
Location: ONLINE Via zoom (ID posted on Brightspace)
Office Hours: Fridays 1:00pm - 3:00pm
Course Website: https://brightspace.carleton.ca/d2l/home

Lab/Course Coordinator:

Vojislav Radonjic (VojislavRadonjic@cmail.carleton.ca)

Teaching Assistants:

Bowen Zhang (BowenZhang9@cmail.carleton.ca)
Andrew Dang (AndrewDang3@cmail.carleton.ca)

Important dates and deadlines can be found here: https://carleton.ca/registrar/registration/dates/academic-dates/, including class suspension for fall, winter breaks, and statutory holidays.

Course Calendar 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.

Includes: Experiential Learning Activity
Precludes additional credit for SYSC 2006.
Prerequisite(s): (COMP 1006 or COMP 1406 or SYSC 2004) with a **minimum grade of C-**.
Lectures three hours a week, tutorial one and a half hours a week.

You MUST **earn a C- or better in this course** to take some required 2nd/3rd year courses!

⚠ Course Objectives & Overview:

A main objective of this course is to teach you the C programming language. The code that you write may be lower-level code than you are used to. Many students struggle with pointers and understanding what goes on behind the scenes when you create variables and call functions. This course will give you a thorough understanding of how variables and stored and accessed and how the computer memory can be managed carefully and correctly. These are concepts that you did not need to worry about in JAVA or Python as it was all handled for you. This course also has the objective of giving you some familiarity with using the Linux operating system. You will use a Virtual Machine and issue direct low-level commands with the underlying operating system. The course will cover the following topics:

- Introduction to Systems Programming
- Working within a Linux environment

- Programming using the C language
- Data Representation (Bit representation, Primitive data types, Compound data types, Pointers)
- Memory Management (Stack and heap, Dynamic memory allocation)
- Arrays and Linked lists
- Concurrent Computing (Concurrent systems, Processes (signals, sockets) and Threads)
- Program Structure (I/O, Procedural program design and organization, Using libraries)
- File I/O, X11 windows and Graphics, Shell scripts

* To take this course, a grade of **C-** or above MUST have been earned in **COMP1406/COMP1006**.

Learning Materials and Other Course/Lab-Related Resources:

Students are not required to purchase textbooks or other learning materials for this course. There is no textbook assigned to this course. Instead, there is a free in-depth set of course notes and coding examples available on Brightspace. All materials created for this course (i.e., course notes, coding examples, lecture recordings, tutorials, tutorial code, assignments, assignment code bases, marking schemes, tests, exams, and test solutions) 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 **copyright violation** and is **strictly prohibited**.

You will be programming in the Linux environment using **VirtualBox**. Go to this website and follow the instructions to get it all set up: https://carleton.ca/scs/technical-support/virtual-machines/

Once you install the VirtualBox, you need to install the Virtual Machine. The Virtual Machine that we will use is named **COMP2401-F24.ova**. You need to download that file and then when you run the VirtualBox, you can select **Add...** from the **Machine** menu to add the virtual machine. One added, you simply click on the virtual machine that you added (on the left side of the window) and then press the **Start** button. The username is **`student`** and the password is **`student`**.

Evaluation:

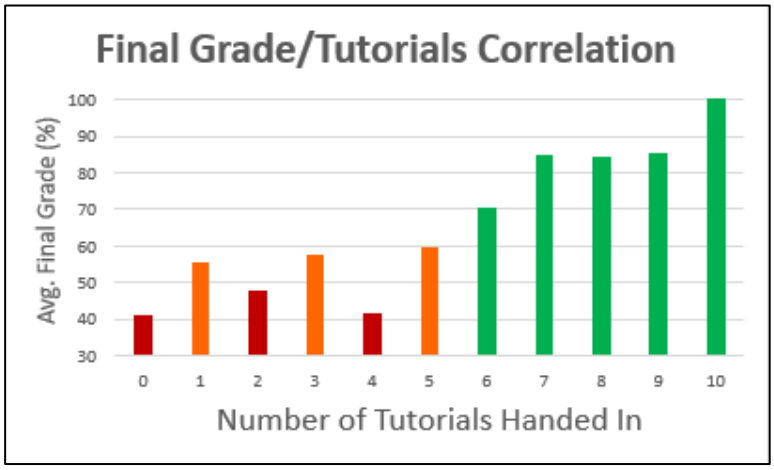
Component	Weight	Details	Due Date
Tutorials (optional)	**10% bonus**	1% each	Weekly, starting Monday, Jan 6th
Assignments	**60%**	best 5 of 6 at 12% each	Biweekly, usually due Tuesdays at 11pm
In-Class Tests	**20%**	best 2 of 3 at 10% each	Jan 27, Feb 26, Mar 24
Final Exam	**20%**	no flexibility, do well	(to be announced)

In-class Tests:

There will be 3 **online** tests during the class timeslot. You must write the test during that time, otherwise you will receive a grade of zero. If you miss one test, the other two will count for your grade. If you miss two tests ... one will end up with a zero grade. If you miss all three tests ... then you will likely need to withdraw from the course to avoid a grade of **F**. **Please note that tests and examinations in this course will use a remote proctoring service provided by Scheduling and Examination Services.** You can find more information at https://carleton.ca/ses/e-proctoring/.

Tutorials:

This course section is offering **optional** self-guided tutorials. You do not have to do them, but if you "complete" one, it adds **up to 1%** bonus to your final grade. The tutorials are an excellent companion to the course notes in that it walks you through the course material with additional examples. They have been designed to highlight typical problems that occur when coding in C and as a result, they will help you to become a better C programmer. If you want to do well in the course, do ALL tutorials (see chart for grade correlation).



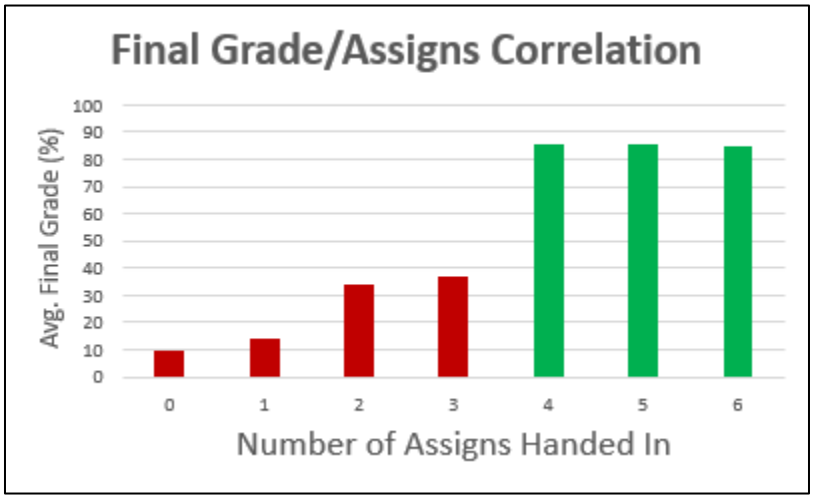
The tutorials will be available on Brightspace as quizzes and will take a varying amount of time ranging roughly from 75 minutes to 110 minutes. The tutorials will only be available during a week-long range, so you must do them sometime within the designated time range. However, you do not have to do the tutorial all at once. You may pause the tutorial and continue at another time as long as you submit by the due date.

The tutorials are self-guided. You will NOT need any help from the TAs. You will simply follow the instructions and do everything it indicates. **You will ONLY get a grade for the tutorials if you submit them (within the time range allotted) AND if you spend enough time on them. If you attempt to simply race through a tutorial by selecting the answers without taking the time to do the tutorial properly, you WILL receive 0 for that tutorial**. Brightspace times your answers for each question. A TA will be checking to confirm that you spent enough time on each question of the tutorial, and marks will be deducted based on your question timing.

Assignments:

There will be assignments in this course which will be available on the course web page.

Lateness: Assignments must be handed in before or on the due date and time. For every **15 minute interval** that an assignment is late, there will be a deduction of **1%** off the grade. Hence, for example, an assignment that is late by **3** hours and **5** minutes will have a deduction of **13%**. After **25** hours ... the deduction is **100%** and the assignment will not even be graded. **Being "sick" on the day an assignment is due is not a valid excuse for not handing anything in.** You have 13 days to do each assignment (a little less for the last assignment). If you do not start your assignment the first week ... and then get too sick to work on it the last week ... you will end up with zero. So, start early and submit partially completed versions so that if you get sick, you will at least have part of it done and submitted. Keep in mind that being sick, does not give you permission to miss an assignment. The chart here shows the grade correlation according to the number of assignments handed in.



Submission: Assignments will be submitted on Brightspace. DO NOT email your assignments to any TAs. If you are at home trying to submit and you cannot get it to work, you will be late with your assignment and will be subject to late submission deductions. So … make sure that your internet connection is reliable. If not, maybe a family member could let you connect to a **WiFi hotspot** through their phone so that you can connect and submit. If you get your grade back and realized that you did not submit the proper files … you cannot re-submit anything again to get remarked, despite

how long you spent on the assignment. Please, therefore, verify that what you handed in is what you wanted to hand in by downloading it yourself right after you hand it in. Lastly, if you submit any version AFTER the due date, it will be late with deductions accordingly.

Backups: Keep a backup of your work, perhaps on a USB flash drive or by sending yourself an email with your assignment files attached.

Marking Issues: It is your responsibility to ensure that your marks on Brightspace are correct. Keep in mind that TAs can make mistakes. All complaints regarding assignment marks should be brought to the attention of **the TA who marked it**. Only if the TA does not address the problem to your satisfaction should you bring the matter to the instructor. This MUST be done **no later than two weeks after the assignment has been graded**. After this time, no remarking will be done.

Cheating and Plagiarism (University Policies):

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 expelled from the program or University. Examples of punishable offences include: plagiarism and unauthorized co-operation or collaboration. 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. Standard penalty guidelines can be found here.

Unauthorized Co-operation or 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". Your assignment code **WILL BE COMPARED** with others in the course.

You are NOT allowed to:

- **copy & edit any portion of another student's code, nor from any websites**, but you may use code from the course notes.
- **give your code (or portions of it) to another student**.
- **work on your assignment with other students**, nor with friends, parents, relatives, etc..
- **post full or partial assignment solutions** on discussion boards or websites (e.g., github, facebook, etc..).
- **share assignment or quiz specifications or post them online** (to sites like Chegg, CourseHero, OneClass, etc.).

You are NEVER permitted to post, share, or upload course materials without explicit permission from your instructor.

If you are unsure of the expectations regarding academic integrity, 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 webpage: https://science.carleton.ca/students/academic-integrity/.

All of the assessed activities in this course were designed to be completed by an individual working alone. Unless it is explicitly stated otherwise, **the use of any AI system will be considered academic misconduct**. This includes, but is not limited to, chatbots (e.g., ChatGPT, Google Gemini, Microsoft Copilot,etc..), research assistants (e.g., Elicit), and image generators (e.g., Stable Diffusion, Dall-E), etc..

Undergraduate Academic Advisor

The Undergraduate Advisor for the School of Computer Science is available in **Room 5302HP**; 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.

Requests for Academic Accommodation

You may need special arrangements to meet your academic obligations during the term. Please go here to read about requests for special accommodations: https://students.carleton.ca/course-outline/

Here are some of the reasons you may request special accommodations … most of these have specific deadline for requesting accommodation. It is the student's responsibility to make the need known to the instructor within the stated deadlines: **Religious obligation**, **Pregnancy obligation**, **Students with Disabilities**, **Survivors of Sexual Violence** , **Student Activities**