Course Outline

COMP 208

Computer Programming for Physical Sciences and Engineering

Winter 2020

section 001: TR	16:05-17:25	WONG 1020
section 002: WF	14:35-15:55	ENGMC 304

Instructor:

Jonathan Campbell

Office:	MC 324	
Office Hours:	TBA	
Email:	jonathan.campbell@mcgill.ca	(see email policy below)

Teaching Assistants (T.A.)

TAs will be available for office hours on the third floor of the Trottier building, room 3090. They can answer your questions about course material as well as assignments. Each TA's office hours will be posted on myCourses.

Overview

Welcome to COMP-208! Please read this document carefully and keep it for reference throughout the term. This course introduces students to computer programming and is intended for those with little or no background in the subject. No knowledge of computer science in general is necessary or expected. On the other hand, basic computer skills such as browsing the Web, sending email and other such fundamental tasks will be necessary in this course.

The course uses the Python programming language. Python is an example of a programming language (as are Java, C++, and many others). A large part of this course will focus on the basic building blocks of programming, which provide the foundations to learning other languages such as Java or C++.

Learning how to program is not easy; it is not a set of facts that one can simply memorize. In principle, a computer program is simply a set of instructions that tells a computer to perform a task. However, finding the right set of instructions can be quite challenging. For that, one has to learn how to structure a larger problem into small subsets, and then find

the solution to each particular subset. This course aims to teach students a way of thinking that will enable them to build non-trivial programs.

Primary Learning Objectives

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

- Design and describe precise, unambiguous instructions that can be used by a computer to solve a problem or perform a task;
- Translate these instructions into a language that a computer can understand (Python);
- Write programs that solve complex problems by decomposing them into simpler subproblems;
- Apply programming style and structure conventions to make programs easy to understand, debug and modify;
- Learn independently about new programming-language features and libraries by reading documentation and by experimenting;
- Solve basic problems in scientific computing, including numerical methods such as root finding, numerical integration, and systems of linear equations.

What This Course is Not About

This course is not about how to use a computer. It will not teach you how to send email, browse the Web, create word processing documents or spreadsheets, setup and configure a computer, use specific software applications (except those needed to complete coursework), design Web pages, or deal with operating system or hardware problems. However, the course offers introductory tutorials that provide instruction in aspects of computer usage necessary to complete coursework.

Prerequisites

Prerequisite: MATH 141 (`https://www.mcgill.ca/study/2019-2020/courses/math-141`) or equivalent

Co-requisite: MATH 133 (`https://www.mcgill.ca/study/2019-2020/courses/math-133`) or equivalent

Note that we will be assuming knowledge of certain concepts in these courses, including derivation, integration, and basic linear algebra.

Course restrictions

Credit can be given only for one of COMP 202, COMP 204, or COMP 208. COMP 208 cannot be taken for credit with or after COMP 250 or COMP 206.

COMP 202 is intended as a general introductory course, while COMP 208 is intended for students with sufficient math background and in (non-life) science or engineering fields.

Required software

Typically when programmers write code they used what is called an integrated development environment (IDE) to write programs. IDEs provide an editor that allows you to type your program, commands to compile and run it, and many other useful tools, all in one application.

We will be using an IDE called Thonny, which you can download at `https://thonny.org`. We will upload a document to myCourses with instructions to get it running, and the TAs and instructor will be available during office hours if you have trouble installing it.

There are many others out there, so if you prefer another IDE (such as Spyder or PyCharm) or using a text editor (such as Atom or Gedit) you are welcome to do so. Note that if you use a different editor, the teaching staff may not be familiar with your choice of editor.

Recommended textbooks

There is no primary textbook. We will be using slides, which will be made available on myCourses, and may reference other resources (some of which are listed below) from time to time.

If you would like to consult a free, online textbook, you can use the following:

Think Python 2e, by Allen B. Downey. Available at no cost under the terms of the Creative Commons Attribution-NonCommercial 3.0 Unported License at `https://greenteapress.com/wp/think-python-2e/`.

Other references

- How to Think Like a Computer Scientist: Interactive Edition, by Brad Miller and David Ranum. This is an interactive textbook based on the above work by Jeffrey Elkner, Allen B. Downey, and Chris Meyers. Available at no cost at `https://runestone.academy/runestone/books/published/thinkcspy/index.html`.
- The Python Tutorial. You can browse or download this from Python Software Foundation's Web site at `https://docs.python.org/3/tutorial/`.

- The Python 3 Documentation. You can also browse or download this from Python Software Foundation's Web site at `https://docs.python.org/3/`.

Contacting instructor and teaching assistants

Please post all your questions about the course (including assignments and the midterm/final) on the myCourses message boards, so that everyone can see both the questions and the answers. You may freely answer other students' questions as well, with one important exception: you may not provide solution code (although you are permitted to provide one or two lines of code to illustrate a point). Email the instructor if and only if you have an urgent and important personal matter. Email TAs only if you have a question about your assignment grade (their email will be part of your assignment feedback).

Students are expected to monitor both their McGill email account and myCourses for course-related news and information. Please subscribe now to myCourses Announcements, if you haven't done so already.

You can also see a TA or the instructor during their office hours for help, or ask a student working at the CSUS Help Desk in Trottier (on the 3rd floor; see here for more details: `https://helpdesk.mcgill-csus.ca/faq`).

MyCourses discussion board rules

Please help out by answering each other's questions on the Discussion Board. The instructor and TAs will try to moderate the Discussion Board. But the Discussion Board works best when students help each other out. When posting to the Discussion Board, please obey the following rules. Postings that do not conform to these rules may be deleted.

- Choose the appropriate folder (Topic).
- Use the search feature to see if your question has been asked before.
- Choose a suitable subject line, so that readers know what the post is about.
- If you have multiple questions that are unrelated, then use multiple postings so people can more easily follow the thread.
- Proofread before posting. Take an extra minute to ensure that what you write makes sense.
- If you would like your posting to be deleted, just add a request within the thread.
- It is nice for you to post a thank you note on myCourses when someone helps you out. However, please keep in mind that everyone subscribed gets notified, which can be a bit annoying if everyone sends them. So please use some discretion.

- Do not post long excerpts of code on the discussion boards. Please post only short excerpts (one or two lines) or error messages.

Grading Scheme and Deadline Policy

The final grades will be calculated using the following percentage breakdown:

- **Four Assignments: 35 % total,**
  (first assignment worth 8 %; others worth 9 %)
- **Midterm Examination: 20 %**
- **Final Examination: 45 %**
- **Participation: bonus 1 %**

Assignments

There will be four assignments consisting of the writing of Python programs. It is very important that you complete all assignments, as this is the best way to learn the material. By working hard on the assignments, you will gain essential experience needed to solve problems on the midterm and final examinations.

To receive full grades, assignments (as well as all other course work) MUST represent your own personal efforts (see the section on Plagiarism Policy and Assignments below).

Assignment submission will take place on myCourses. The instructor and TAs will discuss how to use it during the lectures and tutorials, but every student is responsible for verifying that their submissions are successful. If you believe the content of your myCourses submission box is different from what you have submitted, you must e-mail your section instructor within 5 days of the assignment deadline in question to provide evidence of your correct submission.

Assignment marks will also be posted on myCourses. It is your responsibility to check that the marks are correct and to notify your section instructor of any errors or missing marks. If you believe that your assignment was graded incorrectly, you should first email the TA who marked your assignment. Their email should be in the feedback left on your assignment. If you and the TA cannot resolve the discussion, then you should contact the instructor.

Late Policy: Late assignments will be deducted 10 % each day or fraction thereof for which they are late, including weekend days and holidays; that is, assignments that are between 0 and 24 hours late will be deducted 10 %, assignments that are between 24 and 48 hours late will be deducted 20 %, and so on. Assignments submitted more than 2 days after the deadline will not be accepted, nor graded, and will therefore receive a grade of 0.

The instructor reserves the right to modify the lateness policy for a particular assignment; any such modifications will be clearly indicated at the beginning of the relevant assignment

specifications. Plan appropriately and do not submit to myCourses only minutes before the assignment deadline. Take care: programming assignments are notoriously time-consuming and individual exceptions to the lateness policy will not be granted without appropriate justification submitted in writing and supported by documentary evidence.

Midterm

The midterm examination will take place in the evening at the following date and time:

Thursday, February 27, 2020 starting at 6:05 p.m.

Room assignments will be announced in class and posted on myCourses when it is closer to the date.

Participation

There will be a bonus 1 % given to students who participate in at least 60 % of the polling questions during lectures.

Each lecture may have one or more polling questions posed during class time. During class, students will respond to questions from the instructor using a personal device (smartphone, tablet, or laptop). Students should come to class with their devices charged and connected to the Internet.

Polling will be available through `www.mcgill.ca/polling`. To participate in polling sessions, students must first register for an account by clicking on Register Your Account at `www.mcgill.ca/polling` and logging in with their McGill username and password, then following the prompts. For more information, please visit the Getting Started for Students section at `www.mcgill.ca/polling`.

For any technical problems with polling, please contact the IT Service Desk: `http://www.mcgill.ca/it/get-started-it/need-help`.

If you do not have a phone, tablet, or laptop to use to respond to polling questions, please contact the instructor immediately in order for appropriate arrangements to be made.

To maintain a safe and respectful classroom environment, please ensure that any polling responses you submit are appropriate and relevant to the question asked. Please note that unless the poll is labelled as anonymous, your responses are identifiable to the instructor. Please see the Code of Student Conduct and Disciplinary Procedures.

If a student answers at least 60 % of these questions throughout the term, they will receive the bonus 1 %. Answers will not be graded; simply participating in the poll is sufficient.

Supplemental/deferred final exam

In exceptional situations, students may write a supplemental examination. However, the ability to do so is not automatic, and depends on your exact situation; contact your Student Affairs Office for further information.

The Supplemental/Deferred exam will cover the same material as the Final Exam and it will replace the Final Exam grade, with the same grading policy as stated at the beginning of this section. For information on Supplemental Exams, see `https://www.mcgill.ca/science/student/general/exams/supplemental`.

Final grade

There are many factors that determine your final grade, including how hard you work, how talented you are in this subject, how much time you have available because of other commitments, what your academic background is, what your health situation or family situation is, etc. However, we do not consider these factors when we calculate your final course grade. Rather, we/Excel calculate your final grade according to the grading scheme specified above.

When we calculate your final course grade, we will use a formula that rounds off to the nearest integer. If your grade is 84.4 then it rounds to 84 and you get an A-, whereas if it is 84.6 then it rounds to 85 and you get an A. If your grade is 84.5, our formula will round it up to 85. The same round off procedure holds for low grades. If your calculated final course grade is 49.4 then it rounds to 49 which is an F. We draw a very hard line on this, so if you don't want to fail then you should stay far away from that line.

If you receive a grade of D, F or J, you will not be given the opportunity to complete additional work to upgrade your grade.

Plagiarism policy

McGill University values academic integrity. Therefore all students must understand the meaning and consequences of cheating, plagiarism, and other academic offenses under the Code of Student Conduct and Disciplinary Procedures (see `www.mcgill.ca/integrity/` for more information).

Plagiarism Policy and Assignments

You must include your name and McGill ID number at the top of each source code file that you implement and submit. By doing so, you are certifying that the program or module is entirely your own, and represents only the result of your own efforts.

Work submitted for this course must represent your own efforts. Assignments must be done

individually; you must not work in groups. Do not rely on friends or tutors to do your work for you. You must not copy any other person's work in any manner (electronically or otherwise), even if this work is in the public domain or you have permission from its author to use it and/or modify it in your own work (obviously, this prohibition does not apply to source code supplied by the instructor explicitly for this purpose). Furthermore, you must not give a copy of your work to any other person.

The plagiarism policy is not meant to discourage interaction or discussion among students. You are encouraged to discuss assignment questions with the instructor, TAs, and your fellow students. However, there is a difference between discussing ideas and working in groups or copying the solution from someone else. A good rule of thumb is that when you discuss assignments with your fellow students, you should not leave the discussion with written notes. Also, when you write your solution to an assignment, you should do it on your own.

Getting Help and Partial Credit

Students who require assistance with their assignments should see a TA or instructor during their office hours. If you have only partially finished an assignment, document the parts that do not work, and submit what you managed to complete for partial credit.

Plagiarism detection

We will be using automated software similarity detection tools to compare your assignment submissions to that of all other students registered in the course, and these tools are very effective at what they have been designed for. However, note that the main use of these tools is to determine which submissions should be manually checked for similarity by an instructor or TA; we will not accuse anyone of copying or working in groups based solely on the output of these tools.

You may also be asked to present and explain your assignment submissions to an instructor at any time.

Students who put their name on any code that are not entirely their own work will be referred to the appropriate university official of the student's Faculty (e.g., Engineering, Science, etc.) who will assess the need for disciplinary action. For more details on the process that follows, see for example Section III Articles A.37 (p. 10) and A.48 (p. 13) of the Code of Student Conduct and Disciplinary Procedures: `https://www.mcgill.ca/secretariat/files/secretariat/code_of_student_conduct_and_disciplinary_procedures.pdf`.

Plagiarism and polling questions

Since polling records are used to compute a portion of course grades, responding as someone other than yourself is considered an academic offense. During class, possession of more than one response device or using the credentials of another student will be interpreted as intent

to commit an academic offense. Please refer to McGill's policy on Academic Integrity and Code of Conduct for more details.

Land acknowledgement

McGill University is on land which has long served as a site of meeting and exchange amongst Indigenous peoples, including the Haudenosaunee and Anishinabeg nations. We acknowledge and thank the diverse Indigenous people whose footsteps have marked this territory on which people of the world now gather. Please see here for more details: `https://www.mcgill.ca/edu4all/other-equity-resources/traditional-territories`.

Miscellaneous Policies

Collaboration on assignments (allowed, but see details)

We encourage you to discuss the assignment problems with each other. There is no better way to learn than through discussion with your peers. We also encourage you to help each other out with debugging problems, especially with the mechanics of debugging. Also, we encourage you to pose questions on the Discussion Boards and to answer each other's questions there too.

However, there are limits to this collaboration. You can give hints (and the TAs and instructor will give hints sometimes if you ask). However, your discussion should not go so far that you are revealing the solutions to each other. And you must never copy code from each other.

Re-grading assignments

Mistakes can occur when grading assignments. Not surprisingly, requests for re-grading are always in situations in which students feel they received fewer points than deserved, rather than more points than deserved. With that upward tendency in mind, please note that if you wish the instructor or the TAs to re-grade a question on an exam or assignment, we will do so. However, to avoid upward grade ratcheting, *we reserve the right to re-grade other questions as well.*

McGill Language Policy

In accordance with McGill University's Charter of Students' Rights, students in this course have the right to submit in English or in French any written work that is to be graded. See here for more details: `https://www.mcgill.ca/study/2019-2020/university_regulations_and_resources/undergraduate/gi_lang_policy`.

Office for Students with Disabilities

If you have a disability and require accommodations, the Office for Students With Disabilities (`https://www.mcgill.ca/osd/`) is here to help you sort those out. OSD liaises with the instructor on your behalf to ensure that your accommodations are met.

Pregnancy and Caregiving

Students who are pregnant and/or caring for a dependent also often may find it helpful to receive academic accommodations. McGill's guidelines for accommodations for students who are pregnant and/or caring for a dependent may be found at `https://www.mcgill.ca/study/2019-2020/university_regulations_and_resources/graduate/gi_accommodation_pregnancy_caring_dependants`.

Scent Free Environment

This classroom and associated office hours are a scent free environment. Please refrain from wearing perfume, cologne and body spray in these spaces out of respect for people with neurological & respiratory issues who may be affected by these scents.

Campus Computer Laboratories

Using the SOCS computer laboratory facilities: All students registered in COMP-208 may use the SOCS computer laboratory facilities to do their work regardless of the program in which they are registered. These facilities are located on the third floor of the Trottier building.

Refer to `https://www.cs.mcgill.ca/about/facilities/` for more information on the SOCS computer laboratory facilities.

Other computer laboratory facilities: You may also use other computer laboratory facilities on campus to do your work. Most facilities are available to all McGill students, but there are facilities which grant usage privileges only to students registered in a course or program offered by the faculty or department which manages the facility.

Students should contact the work area of their choice to inquire about access requirements, opening hours, or any further information such as software availability.

Course content

Note that minor changes in content, reading material, and times for tutorials and assignments may occur. It is your responsibility to attend class and be aware of what content is being covered.

Tutorials

Throughout the term, there will be several (optional) tutorials. These will be designed to help you with the material and assignments, and to give you a chance to ask questions in a smaller environment than lectures. Further information will be posted on myCourses. It is not necessary to register for tutorials.

The tutorials will review and practice material presented in class. For example, a tutorial might cover the while and for statements to ensure that everyone is caught up. There will also be special tutorials for Python setup and midterm/final exam review.

Approximate schedule of topics

The following schedule is only approximate and may/will change depending on how the semester unfolds.

Week #	Topics
1	Introduction
2	Python basics, variables, conditionals
3	Turtle, lists, for loops
4	While loops, tuples, list comprehensions, functions
5	Functions, dictionaries
6	Sets, exceptions, file IO
7	Libraries, search algorithms
8	Sorting algorithms, Matplotlib (Midterm this week)
9	OOP, NumPy
10	NumPy, image analysis
11	Numerical integration, solving systems of equations
12	Finding roots of non-linear equations, regression, curve-fitting
13	Random numbers & walks, Monte Carlo methods