CMSC 201 Section 60 Syllabus
Fall 2022

Section 1: Course Information

Course Number	CMSC 201 - Section 60 (plus discussions 61 & 62)
Course Name	Introduction to Computer Science I
Location:	Lecture: ITE 229 Discussion:
Term	Fall 2022
Instructor	Lecture: Alfred W. Arsenault
Contact Information	Email: arsenaul@umbc.edu Cell phone: 410-746-4731
Office Hours	ITE 349; after class. Other times by arrangement
Textbook	None required; online resources will be identified
Final Exam	Monday, December 19, 3:30-5:30 pm

Section 2: Course Overview

An introduction to computer science through problem solving and computer programming. Selected topics in computer science are introduced through programming projects in the Python language running under a UNIX operating system. The core material for this course includes functions, strings, loops, and files. Programming techniques covered by this course include modularity, abstraction, top-down design, specifications, documentation, debugging, and testing. No prior programming experience is required.
Note: the purpose of this class is to teach the principles of computer science, not to teach students how to code in the Python language. Because of that focus, use of certain Python coding conventions will be required and use of other Python coding conventions will be prohibited.

Section 3: Course Objectives

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

1. Solve programming problems using a modern coding language such as Python.
2. Define key concepts in programming including loops, lists, functions, and selection structures.
3. Make use of problem-solving skills, especially in the use of computers to solve real-world problems.
4. Explain and apply the proper steps in developing and creating a computer program.
5. Demonstrate the ability to debug a program so it runs successfully and solves the problem.
6. Use UMBC's UNIX system to create, test, and execute Python programs.
7. Transfer the skills learned to achieve success in future courses, projects, and employment.

Section 4: Grading

There will be approximately 1,000 points available during this semester. Standard grading is that 900 points (90%) earns a course grade of A; 800 points earns a course grade of B; 700 points earns a course grade of C; 600 points earns a course grade of D; and below 600 points results in a course grade of F.
The chart below shows how points are planned to be allocated. This may change during the semester due to circumstances that occur beyond our control. If that occurs you will be notified of those changes.

	Quantity	Points Per Assignment	Total
Homework 0	1	10	10
Homeworks 1-6	6	40	240
Projects	3	100	300
Labs	12	10	100 (drop up to 2)
Midterms	2	100	200

Final Exam	1	150	150
Total			1000

Section 5: Course Structure

The course consists of both lectures and discussion sections. Students are responsible for attending both lectures and their own proper discussion section.

Lectures and Attendance

Lectures are Monday and Wednesday afternoons from 4:00 to 5:15 pm in Room ITE 229.
Lectures will generally not be recorded, but notes will be provided to the students, typically ahead of the class.
Attendance will not normally be taken, but as this is a fairly small class, the instructor expects to know fairly early in the semester who the students are and who regularly shows up.
Should circumstances require that a lecture be presented on-line, students will be notified of how to join the on-line session. All on-line lectures will be recorded and made available to students on the instructor's YouTube channel.

Discussion Sections

Discussion sections are led by Teaching Assistants, who are typically undergraduate Computer Science majors. Discussion sections give students an opportunity tp practice the programming skills covered in class in a controlled environment.
Students are assigned to one specific discussion section. Discussions meet once a week.
Section 61 will meet Monday morning, from 11:00 to 11:50 in Engineering 021A.
Section 62 will meet Wednesday morning from 11:00 to 11:50 in Engineering 021A.
Students are responsible for attending their own, assigned, discussion section meeting.
Attendance ***will*** be taken and attendance in discussion ***is*** part of the student's laboratory grade.
If you have to miss a discussion section for a valid reason, please contact your Teaching Assistant or the Instructor about making up the work.

Office Hours:

The course instructor will be available for office horus after each lecture. Other times can be arranged between students needing assistance and the instructor.

Release of Projects and Homeworks

Release of Projects and Homeworks: Release of these assignments will be done through blackboard announcements and probably in the announcement channel on discord as well. They will be links to google documents with the assignment.

Submission will be done through the GL system. We will show you how to log into the system on your first day of class.

Submissions will be due 11:59:59 on Mondays in general, but the due date and time will be listed on the assignment as well. The system is automatic, so late assignments will not be accepted.

If you need an extension, ask before the due date and time. Projects are very challenging to new programmers, and you should expect to take multiple days of work, generally from 5-10 hours sometimes up to 20 hours if you are having difficulty. You have two weeks to complete these projects so you need to seek help early, start early, and work until you get stuck, then put it away until the next day, and repeat.

Structure of Midterms and Final

There will be three examinations this semester: 2 midterms and a final exam. All tests are “comprehensive” in the sense that the programming tools we learn this semester build on each other sequentially, and thus you can’t “ignore” or “forget” the early semester concepts when working later in the semester.

Exams this semester will be given on Blackboard. Students will login to the course Blackboard page at the regular class time and complete the midterm exam within 75 minutes. (For students with approved accommodations see Section 9..) We do NOT use the lockdown browser. Exams are essentially “open book” in that students have access to search engines and programming tools during the exam. The tests will focus on what the student is able to understand and use to solve problems, NOT on what the student is able to memorize or search for during the exam.

Sample exams will be provided to the students approximately one week before each exam. These will contain the same number and type of questions as the actual exams, and allow the student to not only verify knowledge of the material but also verify the student’s ability to complete the exam in the requisite time.

Section 6: Course Policies

Late Work: No late work will be accepted in this course . All assignments must be submitted by 11:59:59 PM on the day due. The lab assignments are to be done during your weekly discussion session, so attendance is mandatory.

Course Preparedness: You are responsible for all material covered in the lecture, even if it is not in the course slides or web pages. You are responsible for the material in the course slides or web pages, even if it is not covered during lecture.

Section 7: Academic and Technology Resources

The following are the key resources for the students to use this semester:

a. Course github repository: https://github.com/Al-Arsenault201/Fall_2022
   This contains class lecture notes, code samples, and reference material
b. Instructor's YouTube channel: https://www.youtube.com/channel/UCHTFm94enHC_6NHSxZQMiYA
   Contains videos showing certain concepts that need detailed explanation and/or demonstration. ALSO: if any lectures must be provided online, they will be recorded and will be available here
c. Discord server: details TBD

UMBC Tutoring Help:

You can also visit the Learning Resources Center (LRC), where you can find tutoring for CMSC 104, CMSC 201, CMSC 202, and CMSC 203 by appointment. Each appointment is 50 minutes once a week, with a small group of other students taking the same course. To sign up for CMSC tutoring, fill out their enrollment form.

For technology support, you can contact the Technology Support Center (TSC) on the first floor of the Albin O.Kuhn Library. For more information, call 410-455-3838 or check out the website: http://doit.umbc.edu/tsc /

General Python resources:

Python interpreters: https://www.python.org/
Make sure you have an approved version of the Python interpreter if you're working on your own computer

PyCharm IDE: https://www.jetbrains.com/pycharm/
A good, free IDE to use in developing, testing and debugging your code. Get the free community edition, OR use your UMBC ID and get the professional version free for a one-year period.

Section 8: Mental Health Resources

Diminished mental health can interfere with optimal academic performance. The source of symptoms might be related to your course work; if so, please speak with your instructor. However, problems with other parts of your life can also contribute to decreased academic performance. UMBC provides cost-free and confidential mental health services through the Counseling Center to help you manage personal challenges that threaten your personal or academic well-being.

Remember, getting help is a smart and courageous thing to do -- for yourself and for those who care about you. For more resources get the Just in Case mental health resources Mobile and Web App. This app can be accessed on this web page: http://counseling.umbc.edu/justincase

The UMBC Counseling Center is in the Student Development & Success Center (between Chesapeake and Susquehanna Halls). Phone: 410-455-2472. Hours: Monday-Friday 8:30am-5:00pm.

Section 9: Students with Accommodations

Students who qualify for accommodations must register with UMBC's SDS office. SDS will then make arrangements with the instructor so that the student will have a fair opportunity to perform all work. Examples of accommodations provided can be that the student receives extra time to take an exam; or takes the exam in SDS spaces rather than on-line.

Section 10: Academic Integrity

CMSC 201 is a difficult course for many students, and requires a substantial amount of time and effort outside of the classroom. Many of you are learning to code for the first time, and will need to learn new ways of thinking about a problem, new methods for solving a problem, and new techniques for analyzing a problem. Doing the assignments, finding (and fixing) errors and bugs, and improving your coding skills are 100% necessary for you to succeed in computer science.

For this reason, CMSC 201 has very strict rules about academic integrity and student collaboration on all assignments. Cases of academic dishonesty will be dealt with *severely*. If your assignment is found to be "substantially similar" to that of another student, both you and the other student will receive a **grade of 0** for that assignment. Furthermore, all parties concerned will have their prior assignments more closely examined for cheating. A second incident will result in a grade of 'F' for the semester.

We will be using special software to check for cheating. The software is quite sophisticated and has surprised many students in the past. There is no difficulty in comparing every pair of assignments, or comparing assignments submitted to other sections of this course, from previous semesters, or code found online.

Nonetheless, being able to collaborate effectively with other programmers is also an important skill, and we want students to start cultivating it early. To that purpose, we will allow collaboration on some assignments; this will be clearly stated both on the assignment page and in the individual assignment documents. Some will be marked "individual work only," which means you are only allowed to solicit help from the TAs, instructors, and tutors. Other assignments will be marked "collaboration allowed," which means that you will be allowed to work with other current CMSC 201 students (even those in other sections). However, in order to foster individual understanding of the material, there are still restrictions on what collaboration "means" in CMSC 201. See Section 13 of the syllabus for more details.

Regardless of if an assignment allows collaboration or not, there are some basic rules and restrictions that you should never violate in completing your work. If you have questions about what is acceptable, please contact an instructor or TA. What follows is a *non-exhaustive* list of restrictions for completing your assignments in this course.

- **You may not download or obtain anyone else's work.**

  o You should think carefully about the assignment, and the assignment you turn in should be entirely a product of your own understanding of the material.

  o You may not google or search for the solution to an assignment, even if it's "only for reference," even if you put it aside before programming, and even if that code is not from another student.

  o You may not copy code other than that provided in the course materials (slides, book, labs, etc.). o You may **not** purchase or otherwise contract someone else to do the assignment (in whole or in part) for you. If we find that you have done so, it will result in an automatic 'F' in the course. (This includes paying a tutor to solve your assignment.)

- **You may not share or upload the work you do on this course's assignments (homeworks, projects, labs, etc.).**

  o You may not email code, in whole or in part. Do not even email code to course staff!

  o You may not post screenshots of your code, in whole or in part.

  o You may not post code to public repositories or forums, in whole or in part.

  o You may not allow anyone to access your files. This means **properly protecting your work**! Do not leave your computer unlocked if you step away; do not

allow someone to copy code from your monitor; do not give your password to another student.

- **You will be held to UMBC's Undergraduate Student Academic Conduct Policy.**
  - The details of the policy can be found here: http://www.umbc.edu/policies/pdfs/iii-1.10.03.pdf

- **You should come to office hours for assistance.**
  - Come early and often! The day an assignment is due will be very busy!
  - You may go to any office hours, including those held by a TA or instructor other than your own.
  - Part of the learning process of Computer Science is getting stuck – the TAs are there to help answer your questions, and to teach you how to find your own solutions.
  - Make sure you have a specific question, and can explain to the TA what it is you're having trouble understanding and/or what techniques you've already tried to solve your problem.

Another good rule of thumb is that you should never touch someone else's keyboard It can be easy to get carried away -- you just want to help them "fix one thing," but what you end up doing is typing your code into their assignment.

To give you a better idea of the difference between individual work, collaboration, and violating the Academic Integrity policy, we've created a chart with some examples. As always, if you have questions or are unsure if an action would violate the Academic Integrity policy, please ask a TA or instructor.

Action	Allowed for Individual Work	Allowed for Collaborative Work
Getting help from an instructor or TA	**Allowed**	**Allowed**
Consulting the official Python documentation You can find it at https://docs.python.org/release/3.3.2/	**Allowed**	**Allowed**
Discussing course topics covered by assignments	**Allowed**	**Allowed**

Creating, sharing, or copying notes about course topics	**Allowed**	**Allowed**
Getting or receiving help with using GL or the UMBC Linux system	**Allowed**	**Allowed**
Comparing output from your assignments As long as you do not look at each other's code when it's individual.	**Allowed**	**Allowed**
Discussing how to test your code We don't always tell you every little thing that could go wrong with your code, so working together on this is a great idea.	**Allowed**	**Allowed**
Borrowing verbatim from the course slides or book You don't need to cite your sources if you use the book or slides.	**Allowed**	**Allowed**
Working together on practice problems or Interactivities	**Allowed**	**Allowed**
Helping someone else debug their code	Not allowed	**Allowed**
Planning a general design for your program	Not allowed	**Allowed**
Brainstorming general solutions to the assignment	Not allowed	**Allowed**
Explaining why you made a specific design decision in your code	Not allowed	**Allowed**
Looking at someone else's code on their screen, with their permission	Not allowed	**Allowed**
Looking at someone else's code without their permission	Not allowed	Not allowed
Looking at someone else's code on your screen	Not allowed	Not allowed
Copying someone else's code	Not allowed	Not allowed
Two (or more) people writing a single solution to an assignment	Not allowed	Not allowed
Collaborating with someone who is not a current CMSC 201 student (This includes siblings, parents, and students from last semester.)	Not allowed	Not allowed
Giving (or receiving) a detailed explanation of a solution	Not allowed	Not allowed
Looking for solutions or help online	Not allowed	Not allowed

Purchasing solutions (This includes paying a tutor to help solve your assignment.)	Not allowed	Not allowed

Academic Integrity – UMBC

Statement of Values for Student Academic Integrity at UMBC

In February 2001, the Faculty Senate affirmed the importance of our values and practices by adopting the Statement of Values for Student Academic Integrity that is placed on most course syllabi:

Academic integrity is an important value at UMBC. By enrolling in this course, each student assumes the responsibilities of an active participant in UMBC's scholarly community in which everyone's academic work and behavior are held to the highest standards of honesty. Cheating, fabrication, plagiarism, and helping others to commit these acts are all forms of academic dishonesty, and they are wrong. Academic misconduct could result in disciplinary action that may include, but is not limited to, suspension or dismissal.

The purposes of higher education are the learning students and faculty undertake, the knowledge and thinking skills developed, and the enhancement of personal qualities that enable students to be strong contributing members of society. In a competitive world, it is essential that all members of the UMBC community uphold a standard that places integrity of each student's honestly earned achievements above higher grades or easier work dishonestly sought.

All members of the UMBC community are expected to make a commitment to academic honesty in their own actions and with others. Academic misconduct can result in disciplinary action that may include suspension or dismissal. The following are examples of academic misconduct that are not tolerated at UMBC:

- **Cheating:** Using or attempting to use unauthorized material, information, study aids, or another person's work in any academic exercise.

- **Fabrication:** Falsification or invention of any information or citation in an academic exercise.

- **Facilitating academic misconduct:** Helping or attempting to help another student commit an act of academic misconduct.

- **Plagiarism:** Knowingly, or by carelessness or negligence, representing as one's own, in any academic exercise, the intellectual or creative work of someone else.

- **Dishonesty:** Lack of truthfulness or sincerity when interacting with the faculty member regarding an academic exercise

To this end, UMBC undergraduate students also adopted the following Undergraduate Honor Statement as it describes the high standards to which everyone in the community will be held:

I hereby assume the responsibilities of an engaged member in a scholarly and civic community in which academic work and behavior are held to the highest standards of honesty. It is my active participation that affirms these principles and gives them true meaning as well as value in my education. I realize that by committing acts of dishonesty I hurt myself and place an indelible mark on the reputation of UMBC. Therefore, I will not cheat, fabricate materials, plagiarize, or help another to undertake such acts of academic dishonesty, nor will I protect those who engage in acts of academic dishonesty.

For more information on the topic of Academic Integrity, visit: https://academicconduct.umbc.edu/