CDS 230 Section 003: Modeling and Simulation I

Time: TR 3:00 – 4:15 PM
Location: Synchronous online (Zoom lectures)

Instructor: Carlos Cruz (ccruz1@gmu.edu)
Office hours: Fri 4:00 – 5:00 PM
Backup instructor: Hamdi Kavak (hkavak@gmu.edu)

Teaching assistant: Rebecca Burris (rburris2@masonlive.gmu.edu)
Office hours: Tue 4:30 – 5:30 PM
Google phone: (540) 318-3413

Prerequisites: CDS 130 or permission of instructor. Also, basic computer skills, including familiarity with text editors and the command line interface.

Textbook: There is no required text for this class but customized course notes and additional readings will be provided during class.

About the course

Course description: Modeling and simulation is one of the key technologies in the 21^{st} century and is central in engineering, the natural and social sciences, medicine, economics, and many other fields. This course aims to provide students with an introduction to modeling and simulation concepts using the **Python** programming language and to gain confidence in writing computer programs that solve scientific problems. The online lectures will emphasize essential programming concepts using Python and applying those coding skills to convert problem descriptions into functional programs. All Zoom lectures in this class will be recorded automatically for offline access.

Course expectations: This course requires that you learn to solve scientific and mathematical problems using the Python programming language. This will in turn require a *new* **computational thinking** approach, an approach that picks out the essential details of a problem, how to formulate the problem in ways a computer can understand, and how to follow a problem-solving process in ways that the process can be automated. This is generally not an easy task and it requires discipline and hard work.

Computational thinking skills can only be acquired through problem solving and therefore, it will be very essential that you try to do all the assignments and keep up with the assigned readings. In fact, *all* assignments and reading are mandatory unless otherwise specified. One word of advice would be to **begin your assignments early**. Students that wait for the last day - or hour - to begin working on the assignments do not do well. Also, for some of you CDS 230 will be a challenging course and so, if you get stuck, **ask for help**, as soon as possible. You may contact me or the TA in private. I will generally respond to emails very quickly, less than one day, but may take longer on weekends.

Finally, note that CDS-230 does not require a textbook but instead uses customized course notes that are complementary to the online lectures.

Grading and course requirements

Assignments: Computational problem solving will be one of the most important skills acquired in this course. There will be **7 problem sets** in this course. Each problem set will be due at the start of class a week after assigned, **Thursday at 3:00 pm**, unless otherwise indicated by the instructor. There will also be approximately **5 online "quizzes"**, each about 10-20 minutes long that will **require** Respondus Monitor with a webcam. The "quizzes" will be available after the Thursday session and will be due on **Sunday at midnight**.

Exams: There will be **one midterm** and **the final** exam. Both will **require** Respondus Monitor with a webcam. The final exam will be a cumulative assessment of everything in the course. Dates and times for the exams are given in the **Important Dates** section below.
Grading Policy:

Problem Sets (PS)	40%
Quizzes (Q)	5%
Midterm(M)	25%
Final (F)	30%
Total (T)	100%

There is no curve in this class. All the points earned will be counted and converted to a grade as specified in the grading scale given below. Note that I may assign **extra credit** problems in the assignments or exams which will be weighted accordingly. The total number of points T will be computed as follows:

$$T = N_{PS} * 40 + N_Q * 5 + N_M * 25 + N_F * 30 + N_X \tag{1}$$

where N is the *weighted* number of points earned in each category. The final letter grade will be based on the following scale (see sample calculation at the end of this document):

Grade	T
A+	> 96.7
A	93.3-96.6
A-	90.0-93.2
B+	86.7-89.9
B	83.3-86.6
B-	80.0-83.2
C+	76.7-79.9
C	73.3-76.6
C-	70.0-73.2
D	60.0-69.9
F	< 60.0

Hypothetical grade calculation:
Consider a student that gets 200 out of 235 possible points in the problem sets; 40 out of 42 possible points in the quizzes; 104 out of 120 possible points in the midterm; 150 out of 175 possible points in the final and zero extra credit. Then, using (1)

$$T = (200/235) * 40 + (40/42) * 5 + (104/120) * 25 + (150/175) * 30 = 86.2 \tag{2}$$

Therefore, the student will get a B

General course policies

Class participation: A large portion of the material for this course will be presented during the online lectures, **so participation will be essential, but not mandatory**.

Rules for assignments:

- **Assignments submitted after the due date will not be graded. Late assignments will be accepted after the due date only in extenuating circumstances, which you must notify me at least one day in advance**. If I make an exception, then you will get an extra day with a 10% deduction.
- **Assignments will be submitted as one or several Python files**. I will explain during class how to submit your assignments.
- The files you will submit **must** run and produce some output, so it will be your responsibility to **test** your code before submitting.
- Partial credit may be given even if the answers are wrong or give incorrect results as long as a reasonable effort has been made and code has been tested.
- No credit will be given if it is clear that no effort was made (for example just restating the problem) or if the code fails to execute (i.e. if it produces a **syntax** error).
- If you're stuck and can't get help, write as much code as you can and write comments within your code explaining where you're stuck.
- **You may NOT ask me or the TA if your code is correct or not.**
- All assignments will be turned in through Blackboard. Do not email them or they will not get graded!

Policies for online classes

- Activities and assignments in this course will use the Blackboard learning system, available at `https://mymason.gmu.edu`. Students are required to have regular, reliable access to a computer with an updated operating system (recommended: Windows 10 or Mac OSX 10.13 or higher) and a stable broadband Internet connection (cable modem, DSL, satellite broadband, etc., with a consistent 1.5 Mbps [megabits per second] download speed or higher. You can check your speed settings using the speed test on this website: `https://www.speedtest.net/`). For more information visit the ITS website (you can start here: `https://its.gmu.edu/help-support/getting-started/getting-started-for-students/`).
- Activities and assignments in this course will regularly use web-conferencing software (Blackboard Collaborate / Zoom). In addition to the requirements above, students are required to have a device with a functional camera and microphone. In an emergency, students can connect through a telephone call, but video connection is the expected norm.
- This course requires the use of LockDown Browser and a webcam for online exams. The webcam can be built into your computer (internal webcam) or can be the type of webcam that plugs in with a USB cable (external webcam).

University policies

Academic honesty:

All members of the Mason community are expected to uphold the principles of scholarly ethics. On admission to Mason, students agree to comply with the requirements of the GMU Honor Code and System (`https://oai.gmu.edu/mason-honor-code/full-honor-code-document/`). Similarly, graduating students are bound by the ethical requirements of the professional communities they join.

- To uphold the rigor of the course and the value of your degree, the Honor Code will be rigorously enforced. The instructor will use several manual and automated means to detect cheating in all work submitted by students. Keep in mind it is extremely easy to detect cheating with logic and code.
- The penalty for cheating, plagiarism, lying, and stealing will always be far worse than a zero grade, to ensure it is not worth taking the chance. Any instance of misconduct that is detected will be referred to the Office of Academic Integrity (OAI) and will most certainly translate into a lowered grade at a minimum and may result in course failure (a final grade of F).
- All assignments and assessments, including computer programs and associated outputs, submitted for grading must represent the student's own work. It will be interpreted that when any assignment or assessment is submitted that the student is stating that 100% of the work is their own.
- Definitions:
  - **Collaboration**: Students are encouraged to discuss problems and methods with each other. However, when submitting any assignment/assessment, all work must be the student's own work.
  - **Cheating**: The use of an unauthorized source to develop portions of an assignment/assessment. Also, copying someone else's work and submitting it as the student's own work.
  - **Falsification**: The use of data in an assignment that is either not developed per the instructions in the assignment/assessment, or is offered without any supporting work. **If you use outside resources then you are required to cite these sources**.
  - **Plagiarism**: When students copy or quote someone else's work without proper citations or credit to the author of the work. Copying from sources like CHEGG.com and CourseHero.com, and other similar internet resources is considered plagiarism.
- Plagiarism, cheating, and falsification will not be tolerated at any time and in any form. If the instructor determines that a student has committed plagiarism, cheating or falsification in the submission of an assignment or assessment, the instructor shall:
  - For the first incident, the student will be given a zero for the entire assignment or assessment, issued a letter of warning from the instructor, and report the incident to the Chair of the Computational and Data Sciences Department and the MASON Honor Committee for further adjudication and additional remedial action/sanctions. Note that the student cannot receive credit for the course until the MASON Honor Committee process has concluded.
  - For a second incident, the sanctions from the 1st incident will be repeated and the student will be assigned an automatic grade of "F" for the course. Additionally, the 2nd incident's referral to the MASON Honor Committee will recommend suspension from MASON.

Disability statement:

If you have a documented learning disability or other condition that may affect academic performance you should make sure this documentation is on file with Office of Disability Services (`http://ods.gmu.edu`, 703-993-2474) to determine the accommodations you need. Provide the Instructor with a copy of the Office of Disability Services accommodation determination prior to receiving any accommodations. The Instructor will closely protect this information as private and will not share the information with anyone other than the class assistants unless authorized in writing by the student or the Office of Disability Services.

Tentative course schedule

Week 1 (1/26-1/28)	Introduction (Ch 1)
Week 2 (2/2-2/4)	Variables, Data Types (Ch 2)
Week 3 (2/9-2/12)	Control Flow, Collections I (Ch 3-4)
Week 4 (2/16-2/18)	Collections II, Functions (Ch 5-6)
Week 5 (2/23-2/25)	File IO (Ch 7-8)
Week 6 (3/2-3/4)	Arrays, Numpy (Ch 9-10)
Week 7 (3/9-3/11)	Numpy, Matplotlib (Ch 10-11)
Week 8 (3/16-3/18)	**Midterm exam on 3/16**, Exam review
Week 9 (3/23-3/25)	Object Oriented Programming
Week 10 (3/30-4/1)	Introduction to Modeling
Week 11 (4/6-4/8)	Discrete Time Models
Week 12 (4/13-4/15)	Continuous Time Models
Week 13 (4/20-4/22)	Stochastic models
Week 14 (4/27-4/29)	Machine Learning
Week 16 (5/6)	**Final exam**

Important dates

First day of classes	Jan 25
Drop Deadline	Feb 16
Midterm exam (3:00-4:15)	Mar 16
Last day of class	Apr 29
Final exam (1:30-4:15)	May 6