CDS 230 Section 001: Modeling and Simulation I

Time: TR 3:00 – 4:15 PM
Location: Innovation Hall 134
Course format: In-person

Instructor: Carlos Cruz (ccruz1@gmu.edu)
Office: Planetary Hall 117
Office hours: By appointment only

Teaching assistant: In Hoi Kim(ikim21@gmu.edu)
Office hours: R 1:30 – 2:30 PM, RH 249
Google phone: TBD

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: 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. 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. Python is, arguably, the most popular high level programming and the lectures will emphasize essential programming concepts using Python and applying those coding skills to convert problem descriptions into functional programs.

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: There will be **10 graded problem sets** consisting of multiple programming assignments. Each problem set will be due at the start of class a week after assigned, generally **Thursday at 3:00 pm**, unless otherwise indicated by the instructor. I will drop the problem set with the lowest score. There **will** also be 8 **"quizzes"**, each about 10-20 minutes long, available online and after the Thursday class, whose total will count as 1 problem set.

Exams: There will be two in-class exams: **one midterm** and **the final**. Exams are closed book/notes unless specified otherwise by instructor. The final exam will be a cumulative assessment of everything in the course.

The grade distribution is as follows:.

Assignments (A)	50%
Midterm(M)	20%
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. The total number of points will be computed as follows:

$$Total = N_A * 50 + N_M * 20 + N_F * 30 + N_X * 10 \tag{1}$$

where N is the *weighted* number of points earned in each category and N_X is the total number of extra credit points that will be given during exams and some assignments, totaling *up* to 10% above the total. The final letter grade will be based on the following scale:

Grade	Total
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

General course policies

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

Rules for assignments:

- All assignments will be turned in through Blackboard (`https://mymason.gmu.edu`). Do not email them or they will not get graded!
- **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**. Each student gets **three** excuses that allows them to turn in an assignment

24 hours late. After using up your 3 exceptions and you submit late, your maximum score lowers by 20%. Regardless, the latest you can turn in work is 24 hours after the posted deadline, no exceptions.

Note that:

  - Blackboard being unavailable is not an excuse for late submissions. If such an event occurs, email me (and the TA) immediately.
  - Computer failure will not be cause for an extension. Make sure you backup your work!

- If you submit **all** assignments in time, you will get a 1% extra credit reward at the end of the semester.
- **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)
- If the code fails to execute (i.e. if it produces a **syntax** error) you will receive at most 50% and may get some points back if the grader can *quickly* spot the error(s) and fix your code.
- 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.**

Completing coursework: Completing courseworks requires that you complete all assignments and do the required reading from the course notes. You should expect to spend 8-10 hours per week on this course, including the 2.5 hours of lecture time. A student who cannot complete coursework or an exam due to an incapacitating illness or other compelling reason can apply for an extension of time. Note that such extensions are evaluated on a case by case basis and there is no guarantee that an extension will be granted.

Some frequently asked questions:

- **What software do we need?**. We will use Python v3.7 or later as distributed via Anaconda. Students may install Python from other distributions as long as the version number matches.
- **What are the required prerequisites for this course?**. The recommended prerequisite is CDS130 which introduces you to basic programming concepts. Therefore, you are expected to have basic computing skills as well as familiarity with your computer's operating system.
- **Is there a course website?**. All the course work, syllabus, assignments, scripts, course notes, readings, etc, will be distributed through the blackboard course website. Make sure you visit the course website regularly, familiarize yourself with its contents and follow all announcements.

If you have any questions please do not hesitate to contact me.

Policies for courses in a continuing pandemic

- **Course modality**: This course has a face-to-face course modality. All students taking courses with a face-to-face component are required to follow the university's public health and safety precautions and procedures outlined on the university Safe Return to Campus webpage (https://www2.gmu.edu/safe-return-campus). Similarly, all students in face-to-face (and hybrid courses) must also complete the Mason COVID Health Check daily, seven days a week. The COVID Health Check system uses a color code system **and students will receive either a Green, Yellow, or Red email response**. Only students who receive a "green" notification are permitted to attend courses with a face-to-face component. **If you suspect that you are sick or have been directed to self-isolate, please quarantine or get testing. Faculty are allowed to ask you to show them that you have received a Green email and are thereby permitted to be in class**.

- **Masks**: Mason's policy requires that everyone wear a mask in indoor settings. That will make it inconvenient and perhaps difficult to communicate during lectures, but we'll have to obey the rules.

- **Illness**: Students who become seriously ill should contact the instructor as soon as possible. Notifications made days or weeks *after* the illness will be difficult to justify and accommodations may be difficult or not possible at all. So, it is important that, even under difficult circumstances, you exercise personal responsibility. Note that you may be need to provide a doctor's note in order to be granted extensions or accommodations.

General university policies

Course Materials: All course materials posted to Blackboard or other course site are private to this class, that includes course notes and assignments. Also, by federal law, any materials that identify specific students (via their name, voice, or image) must not be shared with anyone not enrolled in this class.

Privacy: In order to comply with student privacy laws, faculty and students need to use their GMU email accounts when corresponding with each other.

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.

- Plagiarism, cheating, and falsification will not be tolerated at any time and in any form.

- 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.

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. **Make sure that you provide the instructor with a copy of the Office of Disability Services accommodation determination as soon as possible**. 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[1] course schedule

Week 0 (1/25-1/27) ..... Introduction (Ch 1)
Week 1 (2/1-2/3) ..... Variables, Data Types (Ch 2)
Week 2 (2/8-2/10) ..... Control Flow, Collections I (Ch 3-4)
Week 3 (2/15-2/17) ..... Collections II, Functions (Ch 5-6)
Week 4 (2/22-2/24) ..... File IO (Ch 7-8)
Week 5 (3/1-3/3) ..... Arrays, Numpy (Ch 9-10)
Week 6 (3/8-3/10) ..... Numpy, Matplotlib (Ch 10-11)
Week 7 (3/15-3/17) ..... **Spring recess**
Week 8 (3/22-3/24) ..... **Midterm on 3/22**, OOP (Ch 13)
Week 9 (3/29-3/31) ..... Introduction to Modeling (Ch 14)
Week 10 (4/5-4/7) ..... Discrete Time Models (Ch 15)
Week 11 (4/12-4/14) ..... Continuous Time Models (Ch 16)
Week 12 (4/19-4/21) ..... Stochastic models (Ch 17)
Week 13 (4/26-4/28) ..... Stochastic models (Ch 17)
Week 14 (5/3-5/5) ..... Machine Learning (Ch 18)
Week 15 (5/10-5/12) ..... **Final exam on 5/12** (Reading day on 5/10)

Chapters (e.g. Ch 1) refer to course notes chapters.

Important dates

First day of classes ..... Jan 24
Drop Deadline ..... Feb 7
Midterm exam (3:00-4:15) ..... Mar 22
Last day of classes ..... May 7
Final exam (1:30-4:15) ..... May 12

[1] Disclaimer: The instructor reserves the right to modify this syllabus at any time during the course to improve the learning experience and classroom environment. The pacing of the course and the list of covered topics may also be adjusted in response to student progress.