CDS 230: Modeling and Simulation I

Fall 2019

Instructor:	Carlos Cruz	**Time:**	T-R 15:00 – 16:15
Email:	ccruz1@gmu.edu	**Place:**	Exploratory Hall 1004.

Office Hours: After class, or by appointment, or post your questions in the forum provided for this purpose.

Objectives: 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 class will combine lectures with hands-on programming emphasizing the art of converting problem descriptions into functional programs. Student participation will be highly encouraged.

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

Participation
A large portion of the material for this course will be presented during the lectures, so regular attendance will be essential. Furthermore, quizzes will be administered during lectures. There will be a **CDS230 Discussion Board** available through Blackboard where students will be able to interact and engage in discussions regarding lecture material and class assignments. **You may not post homework solutions in the forum!**

Course Pages:

1. Course page available through **Blackboard**
2. https://github.com/cds-230/fall-2019

Main References: There is no required text for this class. The following are references that you may find helpful.

- Allen B. Downey, *Modeling and Simulation in Python*. This is an Open Access book available here: http://greenteapress.com/modsimpy/ModSimPy3.pdf. It has the same objectives as this class but with a different approach.
- Brad Miller and David Ranum, *How to think like a Computer Scientist*, available here: https://runestone.academy/runestone/books/published/thinkcspy/index.html. This online book includes an interactive workspace that you may find helpful.
- Class notes and other readings will be provided throughout the semester.

Grading Policy:

Quizzes	10%
Homework	40%
Midterm	20%
Final	30%
Total	100%

The letter grade is based on:

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

There is no curve in this class. All the points earned will be counted and converted to a grade according to the grading scale above. If you do all the assignments and ask for help when the going gets rough then you will get a good grade. "F"s and "D"s are generally awarded to students that fail to do their work and give up on the course material. It is important that you **do not fall behind** in this class.

Quizzes: Quizzes will be administered during class and will consist of simple questions and/or exercises. Quiz questions will be graded as either correct or incorrect and, for the purposes of grading, each will be worth 1 point.

Homework: There will be 12 homework sets in this course, roughly one per week. Each homework will consist of 5-10 short programming tasks. The top 10 homework scores will be used to compute your grade and the bottom two will be discarded.

- **Homework assignments will be submitted as a Python file**. I will provide a template that you can use to submit your homework.
- The Python files you will submit **must** run and produce some output. If a particular homework problem fails to execute, i.e produces an error, then that problem will not get any credit.
- Partial credit may be given only if the answers are **reasonable** or close to the correct answer.

- All homework is due before class one week after they are assigned (Tuesday at 3PM). **Late homework will be not be accepted**.
- All assignments will be turned in through Blackboard. Do not email them or they will not get graded!
- **Ask questions**! Do not hesitate to contact me to discuss a homework problem or any aspect of the course. I will be available for about 30 minutes after every class. **I will not be able to respond to questions on Saturday and Sunday**.

Academic Honesty:

- You may work together with other classmates to discuss the homework assignment, but in the end **you must write your own codes and documents**.
- It is fine (and encouraged) to use other books, internet resources, your friends, etc. to get solution ideas, but **you may not copy-paste or transcribe code and submit as your own**. Get an idea, sleep on it and then generate the answer yourself based on your own understanding.
- **If you use outside resources then you are required to cite these sources**.
- **If there appears to be evidence of cheating or plagiarism, then honor code actions may be initiated**.

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) to determine the accommodations you need.

Tentative Course Outline:

Week 1 (8/27-8/29) ..... Introduction
Week 2 (9/3-9/5) ..... Variables, Data Types
Week 3 (9/10-9/12)) ..... Collections I, Conditions and Loops,
Week 4 (9/17-9/20)) ..... Collections II, Functions
Week 5 (9/24-9/26)) ..... Python Arrays
Week 6 (10/1-10/3) ..... Numpy
Week 7 (10/8-10/10) ..... File IO, Matplotlib
Week 8 (10/17) ..... Midterm
Week 9 (10/22-10/24) ..... Population growth
Week 10 (10/30-10/31) ..... Spreading of diseases
Week 11 (11/5-11/7) ..... Discrete time models
Week 12 (11/12-11/14) ..... Continuous time models
Week 13 (11/19-11/22) ..... Random numbers
Week 14 (11/26) ..... Monte Carlo sumulation
Week 14 (11/28) ..... Thanksgiving recess
Week 13 (12/3-12/5) ..... Continuous field models
Week 16 (12/12) ..... Final

Important Dates:

First day of classes	Aug 26
Drop Deadline	Sep 17
Student self-withdrawal (100% tuition liability)	Sep 18 - Sep 30
Selective withdrawal (100% tuition liability)	Oct 1 - Oct 29
Fall break	Oct 14: Tuesday classes do not meet.
Midterm exam	Oct 17
Thanksgiving recess	Nov 27 - Dec 1: No classes.
Last day of classes	Dec 7
Final exam	Dec 12