**MA 360 – Introduction of Computational and Data Science,**

**Mathematical Modeling and Simulation**

Instructor: Dr. H. Liu

**GENERAL INFORMATION**

MA 360 is a blended learning course that includes multimedia instructional materials, live problem-solving sessions, and team projects. The students will meet the instructor for three class hours per week in the first ten weeks. In the last five weeks, the focus of the coursework turns to team projects. The instructor will meet the class once a week and meet each team separately each week for the project helping sessions.

This is an in-person lecture course meeting at COAS 302 on M, W & F from 2:00 pm – 2:50 pm. However, zoom access  <https://erau.zoom.us/j/94433498540> is provided for those who are sick or other need such as traveling for Sports or Conferences, etc.

**In-person & on-zoom Office Hours M, W & F: 4–6:00 pm, *Office: COAS 301.16.***

***On-Zoom Office Hours T & Th*** <https://erau.zoom.us/j/96011124462>.

[***Linked Google Page for Signing Appointments***](https://docs.google.com/document/d/18qI1m2jttkfZAbWxs0snfsT034uxyyeMO3H-3zNIusE/edit?usp=sharing)***.***

You do not have to sign for in-person office hours on M, W & F. However, either in-person or on-zoom office hours, the signed students take precedence.

Prerequisite: Multivariate Calculus or Single Variable Calculus with approval from Instructor, and familiar with a programming language (e.g., Java, C or MATLAB)

The goal of this course is to learn how to use advanced mathematics language and computation tools to solve real-world problems. The topics of the course cover broad interdisciplinary problems whose solutions heavily depend on mathematical modeling and simulation. Students will gain hands-on experience on how to use software tools such as Numpy, NetLogo, and STELLA to model and simulate real-world projects.

**PERFORMANCE OBJECTIVES**

1. Introduce the major categories of mathematical models as well as their modeling languages and tools
2. Expose students to a broad variety of real-world applications of computational mathematics
3. Train students how to follow the mathematical modeling process to conceptualize problems, validate their models and verify their solutions
4. Improve students’ capability to make a judicious trade-off in their modeling assumptions to abstract complex problems
5. Provide students with hands-on experience in the use of computational software tools such as numpy, NetLogo and STELLA, etc. to model and simulate mathematical problems, and present visual representations of the problem space as well as alternative solutions.
6. Provide students with teamwork experience to solve problems beyond the scope of textbook exercises and typically beyond the scope of effort for one person.

**CONTENT**

Module 1: Matrix Algebra with emphasis on Eigen Vectors and Applications,

Module 2: Matrix Calculus and Error Minimization,

Module 3: Modeling Methodology and Dynamic System Models

Module 4, Agent-based Modeling & Simulation, and

Module 5, Stochastic Data Models & Simulation.

Module 6, Case Studies & Team projects

Lecture Notes, Data, Homework and Project Assignments in [Shared Google Folder](https://drive.google.com/drive/folders/1lRX1snVDMZavqnMdQlnwbCGD2uDFVv9x?usp=sharing)

<https://drive.google.com/drive/folders/1lRX1snVDMZavqnMdQlnwbCGD2uDFVv9x?usp=sharing>

Ebook of the Lecture Notes authored by the instructor is available online <http://modelsim.wordpress.com>.

Reference book: Angela B. Shiflet and George W. Shiflet, *Introduction to Computational Science, Modeling and Simulation for the Sciences*, Princeton University Press, 2nd Ed., 2014, ISBN: 9780691160719.

**GRADING**

|  |  |  |  |
| --- | --- | --- | --- |
| **Exam** | 30 % | **Homework** | 25 % |
| **Team Project** | 35 % | **Participation** | 10 % |

|  |  |  |  |
| --- | --- | --- | --- |
| **Course Calendar and Test Dates, Subject to Change** | | | |
| **Date \*** | **Modules** | **Content** | **Homework Due** |
|  | **Module 1** | **Matrix Algebra** |  |
| **January 12** |  | Introduction of Python & numpy |  |
| **14** |  | Comparing Numpy with Matlab for data processing |  |
| **13** | Overview | Introduction of different categories of math models |  |
| **19** | 1.1.1 | Vector, Matrix and their arithmetical operations |  |
| **21** | 1.2.1 | Linear Transformation and Inverse Matrix |  |
| **24** | 1.2.2, 1.2.3 | Inverse Matrix & Solving linear system equations |  |
| **26** | 1.3.1 | Eigen Vectors - 1 | Hw1 due |
| **28** | 1.3.2 | Eigen Vectors - 2 |  |
| **31** | 1.3.3 | Singular Value Decomposition |  |
| **Feb 2** | 1.3.4 | Principal Component Analysis |  |
| 4 | 1.3.5 | 25 Billion Eigenvector, Google Application 1 |  |
|  | **Module 2** | **Matrix Calculus & Linear Regression** | Hw 2 due |
| **7** | 2.1.1 | Matrix Calculus |  |
| **9** | 2.1.2 | Multivariate Approximations and Gradient Descendent |  |
| **11** | 2.1.3 | Optimizations and Lagrange Multiplier Method |  |
| **14** | 2.1.4 | Least Square & Linear Regression for multivariate functions |  |
| **16** | Review |  | Hw 3 due |
| **18** | **Test 1** |  |  |
|  | **Module 3** | **Compartmental Analysis and Dynamic Models** | |
| **23** | 3.1.2 | Compartmental Analysis |  |
| **25** | 3.1.3 | Model Verification and Validation, |  |
| **28** | 3.1.4 | Fluid mixing for multiple Tanks |  |
| **March 2** | 3.1.5 | Stella Overview and Simulation |  |
|  | **Module 4** | **Agent-based Modeling and Applications** | |
| **4** | 4.1 | Overview of States, Cells, and Environment of Agent |  |
| **7** | 4.2 | Introduction of Netlogo and Agentsheet |  |
| **9** | 4.3 | Ant Foraging Model and Simulation | Hw 4 due |
|  | **Module 5** | **Stochastic Models** | |
| **11** | 5.1 | Monte Carlo Simulation |  |
| **21** | 5.2 | Random Number from Various Distributions |  |
| **23** | 5.3 | Probable Cause – Markov Chain |  |
|  | **Module 6** | Modeling & Simulating Dynamic Systems – Case studies |  |
| **25** | Project Rubrics | Teams are formed, Teamwork Process, | Hw 5 due |
| **28** | 6.1.1 | Spring mass system and second order ODE |  |
| **30** | 3.1.2 | Modeling high order of ODE systems |  |
| **April 6** | 6.1.3 | Modeling and Simulation of Landing Systems |  |
| **13** | 6.2.1 | Nonlinear dynamic, Prey and Predator Models |  |
| **20** | 6.2.2 | Simulating population models with Stella & NetLogo |  |
| **27** | 6.2.3 | Simulating disease spread model with Stella & NetLogo |  |
| **May 4** | ***8 am - 10 am*  *Final Presentation and Final Report with Codes due*** | | |