MATH299M/CMSC389W
Visualization through Mathematica
Fall 2018 1 Credit STICs course

Administrative

Instructor:	Ajeet Gary	agary@terpmail.umd.edu
Office Hours:	MW 12:00PM-1:00PM	MATH3115
Course supervisor:	Dr. William Goldman	wmg@math.umd.edu

Course Website
ELMS Page – At the end I want to post the Final Projects here: http://egl.math.umd.edu/

Overview
This course is designed to teach how to use the most common and useful features of Wolfram Mathematica, an extremely powerful technical computing system that can be used to model a wide range of problems. Students will also learn important techniques and styles that will equip them with a profound mastery of Mathematica. Plotting functions in several ways, making models that can be manipulated in real time by the user, and efficiently computing solutions to complicated equations are among the things we'll cover. We'll use these skills to model various structures in math, computer science and physics, and for the final project every student will pick something relevant to their major (or interest otherwise) to model, whether that be in physics, math, engineering, economics, or anything else mathematical in nature. Over the course of learning these tools students will encounter profound examples of what Mathematica can do, seeing first-hand that creating models that can be manipulated in real time helps greatly in understanding the underlying symmetries and properties of a problem.

Prerequisites
For Math Majors: MATH140/141 (or equivalent)
For CS Majors: CMSC216 and CMSC250
Strongly recommended pre- or co-requisite: MATH241
Coding: No coding background is required. However, having coding experience will put you in a more comfortable position learning Mathematica.
Math: The more math you know the more you'll be able to immediately get out of this class. Though even if you only know introductory calculus there are still tons of uses in calculus, pre-calc and algebra, and as you learn more math you'll be able to apply these tools to more and more topics and problems.
Physics: Some of the most fascinating models that I've personally made are ones that model physics problems. Math is the language of physics, and so having powerful math visualization tools is a major advantage in deeply understanding the results of physics problems.
If you are a Math major, you can sign up for MATH299M whose prerequisites are MATH140 and MATH141, if you are a Computer Science major and would like credit towards your upper-level courses, sign up for CMSC389W. Three 1 credit CS electives (Winter classes or STICs classes) count towards 1 upper level elective for the CS major. This course, as it is cross listed, does indeed count towards this. MATH299M and CMSC389W are the same course, just listed under both departments.
Class Time
Fridays 2-2:50pm in EGR 1104

Course Materials
The text for this course will be a series Mathematica notebooks that will be released weekly, they will be interactive files that define the tools we will be learning that week.
Mathematica has excellent online documentation: http://reference.wolfram.com/language/
As a UMD student you have a free subscription to Wolfram Mathematica that you can download from TERPware: https://terpware.umd.edu/Windows/title/1837

Homework Assignments
All Assignments will be coding projects, most of them small.
Each week there will be a small Mathematica coding assignment on the material that we covered in lecture, each project will be due one week after it is assigned. Models will be submitted electronically on ELMS as a .nb file.
Students must complete 10 out of these 15 assignments. If more than 10 are completed then the highest 10 scores will be taken, and some sort of extra credit will be awarded for completing the extra models. Assignments turned in two or more weeks past the deadline will receive a maximum score of 90%.
The assignment prompts will in general be somewhat open-ended. I will ask to see something visualized, and it will be up to you what it exactly looks like and how it is implemented!

Projects
The two projects will be given after the first and second months of class, and for each you will have until the next project is assigned to complete them. Since students will be coming from different backgrounds, for Project 2 there are three options, Project 2a, 2b and 2c, in the fields of vector calculus, linear algebra and harmonic analysis, respectively, of which only one needs to be completed.

Final Projects
The last month students will be working on their Final Project. Each student will have their own topic of choice (I'll help you pick something that will be cool to model and relevant to your interests!). During our final exam slot (time to be determined) instead of taking a final exam, we will be doing short presentations of everyone's final projects!

Grading

Weekly Models	45%	10 of the assigned 15 must be completed (including EC)
Project 1	10%	
Project 2	10%	
Final Project	35%	

There will also be many opportunities for extra credit.

… < 80 <= B- < 83 <= B < 87 <= B+ < 90 <= A- < 93 <= A < 97 <= A+

Collaboration
Students are allowed to work together on all assignments except the Final Project. If you copy other students' code all semester then on the Final Project, which will be *unique for each student*, you'll have a hard time. At the bottom of each assignment you should list the names of the students you collaborated with.

Schedule (Subject to change)

Week 1: 8/27 – 9/2	**Introduction and Basics: Variables, Lists, Functions, Manipulate** Model C1: Investigating Infinite Sums Model H1: Asymptotic Complexity Exposition
Week 2: 9/3 – 9/9	**Basics II: Defining Functions, Implicit Functions, Table, Map** Model C2: Converting Between Rectangular and Polar Coordinates Model H2: Convergence of Fibonacci Sequence Ratio Model Ex1: Truth Tables
Week 3: 9/10 – 9/16	**Plotting I: Plot, Options and Manipulate controls** Model C3: Manipulating Coefficients of Polynomials Model H3: Constructive/Destructive Interference with Sine Waves
Week 4: 9/17 – 9/23	**Computation: Solve, Limit, Derivative, Integral, Simplify, Expand, Numeric** Model C4: Graphing first N Derivatives of a Function Model H4: Calculate terms of Taylor Expansion Project 1: Visualizing Taylor Approximations Model Ex2: Radius of Convergence on the Real Line
Week 5: 9/24 – 9/30	**CS-Style Scripting: Conditionals, Loops, Block, Module** Model C5: Merge Sort Model H5: Gaussian Row Reduction
Week 6: 10/1 – 10/7	**Graphics: Graphics, Points, Lines, Shapes, Show, GraphicsGrid, Graphics3D** Model C6: Geometric Circle Area Proof Model H6: Visual Proof of the Baudhayana Theorem Model Ex3: Visualizing the Solar System Planetary Orbits
Week 7: 10/8 – 10/14	**Plotting II: RegionPlot, ContourPlot, DensityPlot** Model C7: Radius of Convergence of the Complex Geometric Series Model H7.1: Integral of a Function as Area under Curve Model H7.2: Gravitational Field populated with several bodies Model Ex4: The Mandelbrot Set
Week 8: 10/15 – 10/21	**Plotting III: PolarPlot, ParametricPlot** Model C8.1: Visual Proof of the Pythagorean Identity $Sin^2+Cos^2=1$ Model C8.2: Visualizing Homotopic Equivalence Model H8.1: Visualizing Trig Functions on the Complex Plane Model H8.2: Visualizing Integrals on Washers and Wedges Project 2a: Partial Derivatives and Gradient Visualization Project 2b: Dot Product and Cross product Visualization Project 2c: Fourier Analysis Visualization
Week 9: 10/22 – 10/28	**Plotting IV: Plot3D, ContourPlot3D, ParametricPlot3D** Model C9.1: Investigating the RGB Color Model Model C9.2: Visualizing Non-Orientable Surfaces Model H9.1: Visualizing Trig Functions on the Complex Plane (Part II) Model H9.2: Drawing Conic Sections as Level Sets of Paraboloids Model Ex5: Create Spikey
Week 10: 10/29 – 11/4 (Ajeet will be away)	**Manipulate II: Locators, LocatorAutoCreate, Intro. to Dynamic** Model C10.1: Interactive Polygon Model C10.2: Visualizing Linear Transformations of R^2 Model H10: Poles of Complex Functions
Week 11: 11/5 – 11/11	**Parallelization: Parallelize, Kernels, Timing, Memory**

	Model C11: Parallelizing Finding Glitch Primes Model H11: Parallelization Super-Computing Competition!
Week 12: 11/12 – 11/18	**Development: Import, Export, Packages, Initialization Cells, .m, .mx** Model C12.1: Cardinal Numbers Package Model C12.2: Snapshot Export Widget Model H12: Create Bases Package Model Ex6: Screenshot Portfolio Choosing Final Projects!
Week 13: 11/19 – 11/25 *Thanksgiving Break*	**Advanced Dynamic: Proper use of Dynamic and Dynamic Module** Model C13.1: Modeling Particles in a Box Model C13.2: Pong Model H13: Writing Manipulate in Dynamic
Week 14: 11/26 – 12/2	**Precomputing: The Art of Pre-Computation** Model H14: Precomputation Update to Past Assignment
Week 15: 12/3 – 12/9 **Last Class of Semester**	**Evaluation Control: Evaluate, Hold, Tuning and Debugging** Model C15: Evaluation Control in Model with Change of Variables Model H15: Stable Algorithm for Solving Linear Systems
Final Exam Week	**Showing off the Power of Mathematica through Examples:** Fractals, Relativistic Beaming, Barn-Ladder Paradox, Möbius Transformations, Dynamical Systems, Chaos, Hyperbolic Geometry, Interpolation Techniques, Geometric Constructions, and more

In Black is the topic of the week (the tools we'll be learning)
In Green are the in-class example(s) that we'll look at
In Blue is the homework assigned that day which will be due the next week
In Purple are Extra Credit projects
In Pink are when the Projects are assigned

Final Project ideas:

- **Visualizing Homotopic Equivalence**
- **Constructions from Euclid's Elements**
- **3D Electron Orbits using the Wave Function**
- **Create a Complex Variables Package**
- **Stable Two-Body Orbits**
- **Cryptocurrency Momentum Strategy Stock Analysis**
- **Evolution of Heat in 3D-Space by the Heat Equation**
- **The Stereographic Projection**
- **Visual Proof of Kepler's Laws of Planetary Motion**
- **Electromagnetic Field Lines around a Dipole**
- **Visualizing Fractional Derivatives**
- **Hybrid Plots Package: ParametricContourPlot3D, GraphicsPlot, PlotND, …**
- **Constructing Conic Sections**
- **…**

Course Policies
The standard course policies apply: http://ugst.umd.edu/courserelatedpolicies.html
Students must also fill out the Usage Agreement (uploaded on ELMS) regarding distribution of course materials.