MATH299M
Visualization through Mathematica
1 Credit
Fall 2018

Administrative

Instructor:	Ajeet Gary	agary@terpmail.umd.edu
Office Hours:	TBD	MATH3115
Course supervisor:	Dr. William Goldman	wmg@math.umd.edu

Course Website
TBD

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. 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 physics, economics, calculus and more, 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
MATH140/141 (or equivalent)
Strongly recommended pre- or co-requisite: MATH241
Coding: No coding background is required, and Mathematica isn't exactly a coding language. However, having coding experience will make you more comfortable learning Mathematica.
Math: The more math you know the more you'll be able to immediately get out of this class. Even if you only know introductory calculus though 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.

Class Time
Fridays 2-2:50pm

Course Materials
The text for this course will be a Mathematica Handbook PDF that I've created and will distribute electronically.

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

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 two weeks after it is assigned. Models will be submitted electronically on ELMS as a .nb file. They will generally be quite short, not demanding more than an hour outside of class per week. 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). At the end of the semester everyone will briefly present their projects.
There will be a small point deduction for assignments turned in late.

Grading

Weekly Models (12)	30%	(Lowest 3 assignments will be dropped)
Class Attendance	15%	
Project 1	10%	
Project 2	10%	
Final Project	35%	

There will also be many opportunities for extra credit.

Collaboration
Students are allowed to work together on all projects 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.

Schedule (Subject to change)

Week 1: 8/27 – 9/2	**Introduction and Basics: Variables, Sets, Table, Flatten** **Model C1: Generating Rows of Pascal's Triangle** **Model 1.1: Generating Fibonacci Sequence** **Model Ex1: Multiplication Table**
Week 2: 9/3 – 9/9	**Basics II: Conditionals, Functions and Defining Functions** **Model C2: Converting Rectangular and Polar Coordinates** **Model 1.2: Show Convergence of Fibonacci Sequence Ratio**
Week 3: 9/10 – 9/16	**Manipulate I: Manipulate around Function and Table** **Model C3: Manipulating Polynomial by its Roots** **Model 1.3: Repeated Fractions with Manipulate**
Week 4: 9/17 – 9/23	**Graphics: Points, Lines, Rectangles** **Model C4: Visual Proof - Regular Polygons approach Circle** **Project 1: Draw Fibonacci number Rectangles**
Week 5: 9/24 – 9/30	**Plotting I: Plot on explicit and implicit functions** **Model C5: Manipulating Coefficients of Polynomials** **Model 2.1: Plot Sine and Cosine waves with Amplitude and Phase**

Week 6: 10/1 – 10/7	**Basics III: Solve, Derivatives, Integrals, Simplify, Expand** Model C6: Graphing first N Derivatives of a Function Model 2.2: Calculate terms of Taylor Expansion Model Ex2: Probabilistic Interpretation of the Wave Function
Week 7: 10/8 – 10/14	**Plotting II and Manipulate II: Plot and Manipulate options** Model C7: Plot Sum of Sine Waves Project 2: Visualizing nth-Order Taylor Approximations
Week 8: 10/15 – 10/21	**Plotting IV: RegionPlot** Model C8: Visualizing De Morgan's Laws with Venn Diagrams Model 3.1: Integral of a Function as Area under Curve
Week 9: 10/22 – 10/28	**Plotting V: Plot3D** Model C9: Visualizing the Fundamental Theorem of Algebra Model 3.2: Draw Paraboloids
Week 10: 10/29 – 11/4	**Plotting VI: ContourPlot** Model C10: Visualizing Trig Functions on Complex Plane Model 3.3: Draw Conic Sections from Level Curves of Paraboloids Model Ex3: Electromagnetic Field Lines around a Dipole
Week 11: 11/5 – 11/11	**Plotting VII: ParametricPlot, ParametricPlot3D** Model C11: Creating a Sphere, Cylinder and Torus Model 3.4: Draw a Double Helix and Cone
Week 12: 11/12 – 11/18	**Plotting VIII: PolarPlot, PolarPlot3D** Model C12: Visual Proof of the Pythagorean Identity $Sin^2+Cos^2=1$ Model 3.5: Draw Polar Flowers Model Ex4: 3D Electron Orbits
Week 13: 11/19 – 11/25	**Final Project Decision Day! N students = N different projects**
Week 14: 11/26 – 12/2	**Manipulate III: Locators and LocatorAutoCreate**
Week 15: 12/3 – 12/9	**Evaluation Control: Evaluate and Hold**
Final Exam Week	**Showing off the Power of Mathematica through Examples:** Fractals, Relativistic Beaming, Barn-Ladder Paradox, Moebius Transformations, Dynamical Systems, Chaos, and more

In Black is the topic of the week (the tools we'll be learning)
In Green is the in-class example that we'll do together
In Blue is the assignment assigned that day which will be due in two weeks
In Purple are Extra Credit projects (more of these will be added)

Course Policies
The standard course policies apply: http://ugst.umd.edu/courserelatedpolicies.html