CMSC 388D

Prerequisites: C- or better in CMSC132
Credits: 1
Time and location: TBD

Instructor:

Dr. David Mount

TA:

Sina Mirnejad
Saadiq Shaik

Course Description:

This course provides basic insight into the design process of a Java game engine, 2D graphics, and the mathematics behind it all. CMSC 389<TBD> is primarily intended to focus on 2D elements of a game engine, but will discuss ideas which can be expanded upon and applied to a 3D environment.

The course will demonstrate how good programming principles, such as object-oriented design, can be applied in the context of game development. Also the students will develop a complete game, which could be a nice addition to a programming portfolio.

Resources:

https://www.youtube.com/watch?v=4iPEjFUZNsw&list=PL7dwpoQd3a8j6C9p5LqHzYFSkii6iWPZF
https://www.youtube.com/watch?v=VE7ezYCTPe4&list=PL8CAB66181A502179

Topics Covered:

W	Topic	Description	Projects
1	Game loop	Brief exploration of the provided flag project code, and making the basic game loop	
2	Basic Object	Structure of a generic Game-Object class and how to integrate it into the game loop.	Game of life project
3	Input detection	Introducing the libraries such as "java.awt", and "java.swing"	
4	Input management	Create a central system to process the input.	
5	Event handling	Create a central system to record and process the events created by game objects.	Rogue project part 1
6	Asset loading	Setting up a system to load assets	
7	Drawing, Sprites, and animations	Integrate the original animated sprites into the flag projects code.	
8	Asset caching	Creating a central system that provides assets, for all other objects	Rogue project part 2
9	Layering	Upgrading the draw system to draw in the same order as the layer of the object	
10	Fixing the frame rate	Separating the frame and logic, and allowing the developer to set the rate both loops are traversed	Final Project
11	Fixing the frame rate		
12	Camera	Creating a camera system	
13	2.5 D	Changing the base vectors to make 2.5D perspective.	
14	2.5 D		
	Extra topics if Class advanced faster than expected		
	Pathfinding	132 covers dijkstra's algorithm, we can first implement it, then easily change it to A*.	

Grading:

Grades will be maintained on the CS Department grades server. Students are responsible for all material discussed in lecture and posted on the class repository, including announcements, deadlines, policies, etc.

Final course grade will be determined according to the following percentages:

Projects(60%):

These projects are applications made using the Game engines developed by the student to showcase students' understanding and progress in implementation of course material.

Assignments(10%):

Assignments demonstrate students' understanding of most recent material covered.

Final Project(30%):

Final project is showcase of final state of the students semester long project. Unlike other projects it's not themed. This project is meant to demonstrate the limits and capabilities of students game engine.

Office hours:

TBD

<always welcome to contact trough elms>

Excused Absence and Academic Accommodations:

See the section titled "Attendance, Absences, or Missed Assignments" available at Course Related Policies.

Disability Support Accommodations:

See the section titled "Accessibility" available at Course Related Policies.

Academic Integrity:

Note that academic dishonesty includes not only cheating, fabrication, and plagiarism, but also includes helping other students commit acts of academic dishonesty by allowing them to obtain copies of your work. In short, all submitted work must be your own. Cases of academic dishonesty will be pursued to the fullest extent possible as stipulated by the Office of Student Conduct. It is very important for you to be aware of the consequences of cheating, fabrication, facilitation, and plagiarism. For more information on the Code of Academic Integrity or the Student Honor Council, please visit http://www.shc.umd.edu.

Course Evaluations:

If you have a suggestion for improving this class, don't hesitate to tell the instructor or TAs during the semester. At the end of the semester, please don't forget to provide your feedback using the campus-wide CourseEvalUM system. Your comments will help make this class better.