NEW YORK UNIVERSITY—TISCH SCHOOL OF THE ARTS

CODE LAB: UNITY/C#

GAMES-GT 302-001
SPRING 2017

COURSE SYLLABUS

Professor: Matt Parker madparker@nyu.edu

Course Description

Unity is currently one of the premiere Game Engines, used for developing games for almost every platform imaginable. It boasts a powerful GUI (Graphical User Interface) that allows for easy configuration and setup of games and an asset store that provides thousands of resources for game development.

However, the heart Unity is the entity component model, which centers around game objects with components scripts added to them to control everything from display to behaviors to interaction. Using the C# programming language, we will explore how to use code that can go beyond the built in systems in Unity to create original and diverse games.

In addition to simply learning to program, students in this class will explore models and algorithms useful for developing games. We will discuss how platforms, libraries, frameworks, and engines affect game design, in both empowering and limiting ways. Finally, we will discuss the history of digital games, how new tools have democratized the process of game development, and the costs and benefits of those trends.

Course Objectives

By taking this course, the student will:

1) Develop a working knowledge of Unity
2) Gain an understanding of C#
3) Understand the conceptual foundations of computer programming
4) Grasp the logical context of games as interactive systems
5) Think like a coder, become one
6) Be introduced to the best practice techniques of game software development

Course Format

The course meets once per week. The lecture meeting will be used for lecture on key concepts, in-class lab time, discussion, and presentation of work. Students should expect to put in at least 5 hours per week outside of the class meetings.

Grading and Assignments

Students will have weekly assignments. All readings and assignments are due at the beginning of the next class. All assignments must be submitted to NYU Classes. There will also be a Midterm and Final game project and will be presented in class. Late work will not be accepted, unless expressly discussed with the instructor.

Final Grades will be determined according to the following breakdown:

Participation:	25%
Weekly Assignments:	30%
Midterm:	20%
Final:	25%

Prerequisites:

Students must have completed GAMES-GT-300 Code Lab 0 or have permission from the instructor.

Statement Of Academic Integrity:

Plagiarism is presenting someone else's work as though it were your own. More specifically, plagiarism is to present as your own: A sequence of words quoted without quotation marks from another writer or a paraphrased passage from another writer's work or facts, ideas or images composed by someone else.

In the context of this class, in particular, plagiarism is presenting someone else's digital game as your own. Working together to solve problems, using the internet as a resource to answer questions, etc is allowed, even encouraged, however, it should be done in the service of creating new original work, not duplicating existing games.

Accessibility

Academic accommodations are available for students with documented disabilities. Please contact the Moses Center for Students with Disabilities at 212 998-4980 for further information.

Attendance

Attending and arriving on time to all class sessions is required and expected. This includes all labs, recitations, and critiques. If you will be missing a class due to illness, or unavoidable personal circumstances, you must notify your professor in advance via email for the absence to be excused.

Unexcused absences and being late to class will lower your final grade. Three unexcused absences lower your final grade by a letter. Each subsequent unexcused absence will lower another letter grade. Two tardies will count as one unexcused absence. Arriving more than 15 minutes late to class will also count as an unexcused absence.

SCHEDULE

Below is the planned schedule for this class. It is subject to change at the instructors discretion.

Week 1

Introductions and course overview

- Go over Syllabus
- Class Schedule
- Intros
- Course Overview
- Sharing and Presenting work

- Game Engines vs. Frameworks vs. Programming Languages

- Unity Review

- Coordinate Systems
  - GameObjects
  - Components

- C# Review
  - Monobehavior
  - Variables (private, public, Internal)
  - Debug.Log

- Random
- Noise

Assignment:

- Make a Game:
  - Practice making a game that relies on code

Week 2

Version Control

- What is Version Control?
- Why is it useful?
- Teams, rolling back, merging
- Centralized (CVS, SVN) vs Decentralized (Mercurial, Git)

Create an Assembla
SmartSVN

In class version control nightmare exercise

Assignment:
Put first game into version control, continue development of first game, use version control

Week 3

Game Manager

Object.DontDestroyOnLoad
Static
Singletons
Const
Inputs
Intro to the Debugger!

"Other" Unity Functions
Fixed Update
Late Upate
Awake
etc

Assignment:
Unity Tutorials:
Properties:
http://unity3d.com/learn/tutorials/modules/intermediate/scripting/properties
Debug.Log & Draw Line:
https://www.youtube.com/watch?v=WV9lW3yr0Lw
Singleton:
http://unitypatterns.com/singletons/
Debugger and more:
https://www.youtube.com/watch?v=-D6qXaGSC2k

Unity Game:
Iterate on previous weeks game, properties, multiple scenes, static, and use debugging tools to help build it.

Week 4

File I/O and Strings

- PlayerPrefs
- StreamWriter
  - Construtor
  - Append to file
  - Write
  - WriteLine
  - Close
- StreamReader
  - Constructor
  - ReadLine
- Strings
  - Concatenating
  - Splitting

Assignment:

- Unity Tutorials:
  - Write to a file (Skip from to 5:25 to 7:45):
    https://www.youtube.com/watch?v=5soRZCJlIuk
  - Read from a file:
    https://www.youtube.com/watch?v=5soRZCJlIuk
- Make a Game:
  - Continue a previous game or make a new one. Add to it the ability to save some data to a file, then load it later. One simple example is a high score list.

Week 5

Simple Level Editor

- More on Strings
  - Length
  - Equals
  - LastIndexOf
  - Insert
  - ToCharArray

Making an ASCII Level
- Creating a file
- Reading the file into Unity
- Converting chars into Objects

Saving Level in Unity
- Getting Object Positions
- Saving them into a File
- Loading Objects into a new Scene

Assignment:
- Make a Game:
  - Continue a previous game or make a new one. Make a Scene where GameObjects are created at the start of the scene by being loaded from a text file.

Week 6

JSON & Properties

What is JSON?
Reading JSON
Writing JSON

Simple JSON: http://wiki.unity3d.com/index.php/SimpleJSON

Connecting to an external server, parsing info

Properties

- Unity Tutorials:
  - Properties: http://unity3d.com/learn/tutorials/modules/intermediate/scripting/properties

Week 7

Objects

Object Oriented Programming **vs** Entity Component Model
Object Oriented Programming **with** Entity Component Model
Objects inside of Objects
Arrays of Objects

Extending Monobehavior

Polymorphism
- Subclasses
- Overriding (virtual, override)
- Inheritance
- Casting

Making a class that does not extend MonoBehavior

Assignment:
- Begin Midterm Game:
  - Midterm game can be a continuation of previous work, but must show substantial additional work.

Week 8

Midterm Presentations

Assignment:
- No Homework!

Week 9

The 3rd Dimension

How Camera's work in 3D Software
Picking an Object in 3D
3D Rotation
Gimble Lock
Quaternion

https://www.youtube.com/watch?v=zc8b2Jo7mno

Assignment:
- Unity Tutorial:
- 3D Clock using Euler
  - Euler (gimbal lock) Explained: https://www.youtube.com/watch?v=zc8b2Jo7mno
  - Unity Quaternions: https://unity3d.com/learn/tutorials/modules/intermediate/scripting/quaternions
  - General Quaternions: http://quaternions.online/index.html
- Make a 3D Game!

Week 10

3D Math

Dot Product
Cross Product
Normals
Basic Matrix Transformations

Week 11

Data Structures
System.Collections.Generic
Lists
Dictionaries

Assignment:

Unity Tutorials:

List and Dictionaries:
http://unity3d.com/learn/tutorials/modules/intermediate/scripting/lists-and-dictionaries

Generics (Optional, Advanced):
http://unity3d.com/learn/tutorials/modules/intermediate/scripting/generics

Begin Final Game. Final game can be a continuation of previous work, but must show substantial additional work.

Week 12

Pathfinding

A* (Pronounced "A Star")

Assignment:

Unity Tutorial:

A* Pathfinding Project:
https://www.youtube.com/watch?feature=player_embedded&v=PUJSvd53v4k

Week 13

In Class Work on Final Presentations

Week 14

Final Presentations

Recommended Addition Readings:

C# Basics for Unity: https://www.youtube.com/playlist?list=PLa7_zxbnGOSbf5xtlkgPwp50z2ZshOdKu