INTERMEDIATE GAME DEVELOPMENT
(GAMES-UT 121-001)

TuTh 9:30-12:15 PM
2 Metrotech (MAGNET) ROOM 825

Instructor: Aaron Freedman <aefreedman@nyu.edu>, office hours by appointment
Jenny Jiao Hsia <jj1317@nyu.edu>
Teaching Assistant: Laurenz Riklin <lr1944@nyu.edu>

Course Description

This course reflects the various skills and disciplines that are brought together in modern game development: game design, programming, asset creation, and critical analysis. Classroom lectures and lab time will all be used to bring these different educational vectors together into a coherent whole; the workshop will be organized around a single, long-term, hands-on, game creation project. At the completion of this course, the student will be able to:

1. Describe typical work practice in game development.
2. Demonstrate competency through actual implementation of code and assets.
3. Work with a game engine, and understand the basics of how to build a game in the engine.

Course Website

github.com/aefreedman/SP2018_IntermediateGameDev
Additional instructions are detailed on the “Wiki” tab.

Course Hardware, Software, and Readings

All software is free, or has free student versions

1. A laptop
2. Unity3D Personal Edition
3. Autodesk Maya
4. Substance
5. SourceTree
6. *10PRINT*, by Nick Montfort, et al.

Learning Outcomes

Design, code, and asset creation as a unified discipline

- Iterative prototyping processes and troubleshooting, isolating bugs and problems.
- Code literacy, input and control structures (if / else / for / while), basic OO code patterns.
- Conceptualizing 3D space / raycasting / basic vector math, movement and collisions.
- Basic 3D polygon modeling and texturing workflows, and basic asset considerations.

Attendance Policy

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 before class via email** for the absence to be excused. Please be aware that you are still required to turn in work on time even if your absence is excused.

Unexcused absences and being late to class will lower your final grade. Excessive tardiness will also be counted as absences.

3+ unexcused absences lowers grade
2 tardies = 1 absence 15+ min late = 1 tardy

Assessable Tasks

Participation

Participation comprises active engagement in class activities, including but not limited to

- Group discussions and critiques
- Playtest sessions and code reviews
- Working on class material during lab periods

Your participation grade will also be affected if you engage in activities that are detrimental to or disruptive of the class environment. For example:

- Checking Facebook or playing Hearthstone
- Using headphones during non-work periods
- Working on assignments for other classes during class
- Showing up to class unprepared, e.g. not having Unity or Maya installed
- Sleeping in class

Weekly Assignments

Weekly assignments require the following deliverables, provided by links in the relevant pages on the course repository.

- A working WebGL build on a personal Itch.io project page
- A git repository for the project
- A devlog post on your dedicated Itch.io devlog project

Midterm Project, Individual

A short autobiographical game, with one or more gameplay elements, i.e. a gate, an obstacle, a challenge, or an activity.

Midterm Deliverables

- A working WebGL build on a personal Itch.io project page
- A git repository for the project

Final Project, Group

Work in groups of 3-4 students to build a small game with a secret theme.

Final Project Deliverables

- A working WebGL build on a personal Itch.io project page
- A git repository for the project
- One paragraph description of the game
- Three screenshots
- Documentation that each member of the team completed each of the following tasks
    - At least one code task
    - At least one asset creation task
    - At least one design task

Assignment Delivery & Late Assignment Policy

Deliverables must be provided to the location and in the format specified by the assignment, which in most cases (*but not all cases)* is on an Itch.io project with a link to that project in the course wiki.

Deliverables are due by **5PM on the evening before** the lecture session. This is to allow the course instructors enough time to check the deliverables before the class period, and also to encourage you to sleep before class. **Please note that the midterm and final projects have special due dates** (specified in the schedule, below)**.**

For weekly assignments, any late assignments will automatically receive no credit.

For the individual and group assignments, your project grade will be immediately reduced by ten percent of the total possible score. You will also lose an additional ten percent of the total possible score for each additional 24 hours that the project is late.

Schedule (subject to change)

*Homework assignments appear as bullets on the week they are *assigned.*

1/23 01: introductions, what is game dev, editor interface, exporting

- Devlog: “The Door Problem” by Liz England
- Develop: a door scene

1/30 ~02: software architecture, vector math, intro to Maya, intro to version control

- Devlog 10PRINT ch. 10
- Worksheets,
- Build a 2.5D treasure hunt game

2/06 ~03: Physics, Maya, more version control,

- Devlog 10PRINT ch. 25
- Build a Rube Goldberg machine with custom models

2/13 04: Scripting physics, using triggers,
Begin midterm

- Devlog midterm project idea
- Prototype midterm project

2/20 05: Raycasting, instantiation
In-class playtest

- Devlog midterm progress
- Iterate on midterm
- Install Substance Painter

2/27 ~06: Intro to 3D painting in Substance Painter
In-class playtest

- Homework: devlog midterm progress
- Iterate on your midterm project

3/06 07: playtest midterms
Midterm deliverables due 5PM before LAB
Midterm presentations during lab

- Devlog midterm project retrospective
- Read Vlambeer maze generator article

3/13 SPRING BREAK, NO CLASS

3/20 08: Midterm debrief, intro to procedural generation

- Devlog“What Do Prototypes Prototype”,
- Make a maze generator

3/27 09: Even more software architecture and version control
Begin Final project

- Devlog final project plan
- Prototype main game systems

4/03 10: How2CollabWitGit, coroutines
Code review

- Devlog final project progress
- Iterate on final project

4/10 ~11: intro to animation in Maya
Code review
In-class playtest

- Devlog final progress
- Iterate on final project

4/17 12: Game feel and “juiciness”
Code review

- Devlog final progress
- Iterate on final project

4/24 13: How do GPUs work, intro to shaders
In-class playtest
Code review

- Devlog final progress
- Iterate on final project

5/01 14: Lab-swap
In-class playtest
Code review

- Devlog final progress
- Iterate on final project

5/03 14.5: Presentations

Final deliverables due May 8, end of day

Grade Calculation

Students will be graded primarily on demonstrated process and technique. Students will be given grades based on a 100-point scale. Each assignment will be graded on a point scale, and these points will be added up to determine the final grade, according to the following:
98-100 A+ 92-97 A 90-91 A- 88-89 B+ 82-87 B etc.

The following are the components of the grade:

Attendance & participation	25 pts.
Homework	25 pts.
Midterm	20 pts.
Final	30 pts.
TOTAL	100 pts.

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. All writing tasks in this course will use this standard.

Please adhere to the following guidelines for all game development tasks (code, asset creation, etc.). These guidelines apply to both individual and collaborative work.

- You are expected to write your own code, within reason
- Copying or mimicking code from course materials or official documentation *is permitted*
- Using code from Unity's standard assets is permitted unless specified otherwise
- In cases where you have used code from an unofficial source, document such use within the file
- You may not use 3D assets from another class in your assignments for this class
- You may not use any assets downloaded from the internet, Unity Asset Store, or some other source without written permission
- Make sure to check licenses for any code that you're going to copy (just because it's online doesn't mean you can copy-paste it into your game)

In most cases, you will find yourself copy-pasting code from the web when you do not know how to do something. Document the source of your solution for your own sanity. The solution you found might not be correct.

Official Tisch policies on academic integrity are available here:
http://tisch.nyu.edu/student-affairs/important-resources/tisch-policies-and-handbooks

Health & Wellness

Your health and safety are a priority at NYU. If you experience any health or mental health issues during this course, we encourage you to utilize the support services of the 24/7 NYU Wellness Exchange 212-443-9999. Also, all students who may require an academic accommodation due to a qualified disability, physical or mental, please register with the Moses Center 212-998-4980. Please let your instructor know if you need help connecting to these resources.

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.