

# CSYE 7270

## Virtual Environments and Real-Time 3D

### Course Syllabus

#### Course Information

Professor: Nik Bear Brown

Email: [ni.brown@neu.edu](mailto:ni.brown@neu.edu)

Office: 505A Dana Hall

Office: Zoom Only

Note: I am also a master's student at Northeastern. Do not send e-mail to my student e-mail [brown.ni@husky.neu.edu](mailto:brown.ni@husky.neu.edu) I almost never read that e-mail.

The class sessions will be hybrid through Zoom and in-class.

Course website: Canvas

The class sessions will be hybrid through Zoom and in-class. You have the choice to attend in class or through Zoom. They are synchronous. You are expected to attend class during the class time.

#### Course Prerequisites

Programming in some language like python.

#### Course Description

Examines how to program and design to create interactive experiences and immersive virtual environments in real-time. Students learn to create interactive real-time experiences and applications to areas such as virtual and augmented reality, games, data visualization, medicine, industrial design, and film.

Introduces real-time 3D rendering engines and the different subsystems used to create 2D and 3D worlds, including rendering, animation, collision, physics, audio, trigger systems, shading, control logic, behavior trees, and simple artificial intelligence. Offers students an opportunity to learn the inner workings of real-time graphics engines and how to use libraries, such as physics and graphics libraries, to develop interactive real-time virtual environments. Discusses graphics pipeline, scene graph, level design, behavior scripting, object-oriented game design, artificial intelligence, analytics, and scripting languages.

#### Learning Objectives

This course examines how to program for virtual environments and real-time 3D in Unreal Engine or Unity 3D. Introduces the different subsystems used to create 2D and 3D worlds, including rendering, animation, collision, physics, audio, trigger systems, shading, game logic, behavior trees, and simple artificial intelligence. Offers students an opportunity to learn the inner workings of game and graphics engines and how to use libraries such as physics and graphics libraries to develop virtual environments. Discusses graphics pipeline, scene graph, level design, behavior scripting, object-oriented game design, world editors, and scripting languages. Includes servers and mobile programming (Android and iPhone) will also be introduced.

- There will be weekly assignments creating small games or tools, 3D simulations or virtual environments.
- There will be biweekly in-class progress presentations.
- There will be a group project and an individual project.

The first half of the class will primarily involve the mechanics of programming for program for games, virtual environments and real-time 3D in Unreal Engine or Unity 3D. The focus is “hands on” learning by building a series of small projects. Students will be given a simple skeleton and then expected to enhance/extend the skeleton for each assignment.

The second half of the class will primarily involve social/network gaming, virtual reality and augmented reality programming.

Learning objectives for the course are:

- Learn how to program in C++ and C# in a clear, robust, efficient, and safe manner.
- Game engine rendering
- Animation
- Game engine collisions
- Game engine physics
- Game engine audio
- Game engine trigger systems
- Game engine shaders
- Augmented reality
- Virtual reality
- Social/network gaming
- Mobile gaming
- Game servers
- Basic Game AI
- Basic Game Analytics

## Weekly Schedule

Week 1	Game Design Document or GDD
Week 2	Unity Game Engine Basics
Week 3	Unity Game Engine Basics
Week 4	Unreal Engine Basics

Week 5	Unreal Engine Basics
Week 6	Shaders & Materials
Week 7	Shaders & Materials
Week 8	Shaders & Materials
Week 9	Particle Effects
Week 10	Animation
Week 11	Animation
Week 12	Audio
Week 13	Profiling/Optimization
Week 14	Game AI Basics
Week 15	Final Projects

## Course Software

For games and real-time 3D students will their choice of one of two game engines Unreal Engine or Unity 3D

If your choice is Unreal Engine, you need to download and install the free and open-source Unreal Engine <https://www.unrealengine.com/en-US/>

If your choice is Unity 3D you need to download and install the free and closed-source Unity 3D <https://store.unity.com/download>

We will cover both extensively the first couple of weeks of class so students understand the differences and make the right choice for them.

For VFX all students will be using the free version of SideFX: Houdini <https://www.sidefx.com/>

While optional the free the free and open-source 3D creation suite Blender would be helpful for all students. <https://www.blender.org/download/>

## Course GitHub

The course GitHub (for all lectures, assignments and projects):

<https://github.com/nikbearbrown/YouTube>

## nikbearbrown YouTube channel

Over the course of the semester I'll be making and putting additional data science and machine learning related video's on my YouTube channel.

<https://www.youtube.com/user/nikbearbrown>

The purpose of these videos is to put additional advanced content as well as supplemental content to provide additional coverage of the material in the course. Suggestions for topics for additional videos are always welcome.

## Teaching assistants

The Teaching assistants are:

TBA

Programming questions should first go to the TA's. If they can't answer them then the TA's will forward the questions to the Professor.

## Learning Assessment

Achievement of learning outcomes will be assessed and graded through:

- Quizzes
- Completion of assignments
- Completion of term projects

## Reaching out for help

A student can always reach out for help to the Professor, Nik Bear Brown [ni.brown@neu.edu](mailto:ni.brown@neu.edu). In an online course, it's important that a student reaches out early should he/she run into any issues.

## Grading Policies

A point system is used. Everything that you are expected to turn in has points. Points can range from 1 point to 1000 points Assignments get a 10% deduction for each day they are late rounded up. Exams cannot be made up unless arrangements are made before the exam.

I expect to use the following grading scale at the end of the semester. You should not expect a curve to be applied; but I reserve the right to use one.

Score	Grade
93 – 100	A
90 – 92	A-

88 – 89	B+
83 – 87	B
80 – 82	B-
78 – 79	C+
73 – 77	C
70 – 72	C-
60 – 69	D
<60	F

Typically grades will end up roughly 25% A, 25% A-, 25% B+, 20% B, 5% less than B but that depends on students' performance.

## Canvas

You will submit your assignments via Canvas and Github. Click the title of assignment (Canvas -> assignment -> <Title of Assignment>), to go to the submission page. You will know your score on an assignment, project or test via Canvas. Canvas only represents only the raw scores. Not normalized or curved grades. A jupyter notebook file ALONG with either a .DOC or .PDF rendering of that jupyter notebook file must be submitted with each assignment.

Your name MUST be part of your submission, for example Sanchez Rick Assignment 1.zip

Multiple files must be zipped. No .RAR, .bz, .7z or other extensions.

Assignment file names MUST start with students last name then first name OR the groups name and include the class number and assignment number.

Assignment MUST estimate the percentage of code written by the student and that which came from external sources.

Assignment MUST specify a license at the bottom of each notebook turned in.

All code must adhere to a style guide and state which guide was used.

## Due dates

Due dates for assignments at midnight on due date of the assignment.

Five percent (i.e. 5%) is deducted for each day an assignment is late. Solutions will be posted the following Monday. Assignments will receive NO CREDIT if submitted after the solutions are posted. Any extensions MUST be granted via e-mail and with a specific new due date.

## Course Materials

## ML-Agents

The Unity Machine Learning Agents Toolkit (ML-Agents) is an open-source Unity plugin that enables games and simulations to serve as environments for training intelligent agents. Agents can be trained using reinforcement learning, imitation learning, neuro-evolution, or other machine learning methods through a simple-to-use Python API. We also provide implementations (based on TensorFlow) of state-of-the-art algorithms to enable game developers and hobbyists to easily train intelligent agents for 2D, 3D and VR/AR games. These trained agents can be used for multiple purposes, including controlling NPC behavior (in a variety of settings such as multi-agent and adversarial), automated testing of game builds and evaluating different game design decisions pre-release. The ML-Agents toolkit is mutually beneficial for both game developers and AI researchers as it provides a central platform where advances in AI can be evaluated on Unity's rich environments and then made accessible to the wider research and game developer communities.

<https://github.com/Unity-Technologies/ml-agents>

Most textbooks related to Unity 3D and Unreal Engine are all available for free to NEU students via SpringerLink (<http://link.springer.com/>). there are no required textbooks.

Further Unity 3D and Unreal Engine have many tutorials on their websites. The class is very hands-on and we will be extending and adapting the Unity 3D and Unreal Engine learning materials.

## Participation Policy

Participation in discussions is an important aspect on the class. It is important that both students and instructional staff help foster an environment in which students feel safe asking questions, posing their opinions, and sharing their work for critique. If at any time you feel this environment is being threatened—by other students, the TA, or the professor—speak up and make your concerns heard. If you feel uncomfortable broaching this topic with the professor, you should feel free to voice your concerns to the Dean's office.

## Collaboration Policies

Students are strongly encouraged to collaborate through discussing strategies for completing assignments, talking about the readings before class, and studying for the exams. However, all work that you turn in to me with your name on it must be in your own words or coded in your own style. Directly copied code or text from any other source **MUST** be cited. In any case, you must write up your solutions, in your own words. Furthermore, if you did collaborate on any problem, you must clearly list all of the collaborators in your submission. Handing in the same work for more than one course without explicit permission is forbidden.

Feel free to discuss general strategies, but any written work or code should be your own, in your own words/style. If you have collaborated on ideas leading up to the final solution, give each other credit on what you turn in, clearly labeling who contributed what ideas. Individuals should be able to explain the function of every aspect of group-produced work. Not understanding what plagiarism is does not

constitute an excuse for committing it. You should familiarize yourself with the University's policies on academic dishonesty at the beginning of the semester. If you have any doubts whatsoever about whether you are breaking the rules – ask!

Any submitted work violating the collaboration policies WILL BE GIVEN A ZERO even if “by mistake.” Multiple mistakes *will be sent to OSCCR for disciplinary review.*

To reiterate: **plagiarism and cheating are strictly forbidden. No excuses, no exceptions.** *All incidents of plagiarism and cheating will be sent to OSCCR for disciplinary review.*

## Assignment Late Policy

Assignments are due by 11:59pm on the due date marked on the schedule. It is your responsibility to determine whether or not it is worth spending the extra time on an assignment vs. turning in incomplete work for partial credit without penalty. Any exceptions to this policy (e.g. long-term illness or family emergencies) must be approved by the professor.

Assignments will receive NO CREDIT if submitted after the solutions are posted. Any extensions MUST be granted via e-mail and with a specific new due date.

Only ONE extension will be granted per semester.

## Student Resources

**Special Accommodations/ADA:** In accordance with the Americans with Disabilities Act (ADA 1990), Northeastern University seeks to provide equal access to its programs, services, and activities. If you will need accommodations in this class, please contact the Disability Resource Center ([www.northeastern.edu/drc/](http://www.northeastern.edu/drc/)) *as soon as possible* to make appropriate arrangements, and please provide the course instructors with any necessary documentation. The University requires that you provide documentation of your disabilities to the DRC so that they may identify what accommodations are required, and arrange with the instructor to provide those on your behalf, as needed.

**Academic Integrity:** All students must adhere to the university's Academic Integrity Policy, which can be found on the website of the Office of Student Conduct and Conflict Resolution (OSCCR), at <http://www.northeastern.edu/osccr/academicintegrity/index.html>. Please be particularly aware of the policy regarding plagiarism. As you probably know, plagiarism involves *representing anyone else's words or ideas as your own*. It doesn't matter where you got these ideas—from a book, on the web, from a fellow-student, from your mother. It doesn't matter whether you quote the source directly or paraphrase it; if you are not the originator of the words or ideas, *you must state clearly and specifically where they came from*. Please consult an instructor if you have any confusion or concerns when preparing any of the assignments so that together. You can also consult the guide “Avoiding Plagiarism” on the NU Library Website at [http://www.lib.neu.edu/online\\_research/help/avoiding\\_plagiarism/](http://www.lib.neu.edu/online_research/help/avoiding_plagiarism/). If an academic integrity

concern arises, one of the instructors will speak with you about it; if the discussion does not resolve the concern, we will refer the matter to OSCCR.

**Writing Center:** The Northeastern University Writing Center, housed in the Department of English within the College of Social Sciences and Humanities, is open to any member of the Northeastern community and exists to help any level writer, from any academic discipline, become a better writer. You can book face-to-face, online, or same day appointments in two locations: 412 Holmes Hall and 136 Snell Library (behind Argo Tea). For more information or to book an appointment, please visit <http://www.northeastern.edu/writingcenter/>.