CSYE 7270: Virtual Environments and Real-Time 3D

Course Syllabus

Course Information

- **Professor:** Nik Bear Brown
- **Email:** ni.brown@neu.edu
- **Office:** 505A Dana Hall
- **Office Hours:** 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. I almost never read that e-mail.

Class Format: In-class sessions only (NO ZOOM)
Course Website: Canvas

You are expected to attend class during the scheduled 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 (GDD)
- In-class instruction

Week 2

- Unity Game Engine Basics

Week 3

- Unity Game Engine Basics
- **Generative AI Focus:**
    - Tools: Midjourney, DALL-E 3
    - Topics: Creating game sprites, simple textures, and UI elements
    - Integration: Importing AI-generated assets into Unity projects

Week 4

- Unreal Engine Basics

Week 5

- Unreal Engine Basics
- **Generative AI Focus:**
    - Tools: Stable Diffusion, ControlNet
    - Topics: Creating environmental textures and concept art
    - Integration: Material workflow with AI-generated assets in Unreal

Week 6

- Shaders & Materials

Week 7

- Shaders & Materials
- **Generative AI Focus:**
    - Tools: Adobe Firefly, RunwayML
    - Topics: Procedural texture generation, normal maps from AI images
    - Integration: Creating custom material libraries with AI assistance

Week 8

- Shaders & Materials

Week 9

- Particle Effects
- **Generative AI Focus:**
    - Tools: Nvidia Canvas, Leonardo.AI
    - Topics: Creating particle textures and effect sheets
    - Integration: Enhancing VFX systems with AI-generated elements

Week 10

- Animation
- In-class instruction
- Friday: Show and Tell - Students present progress

Week 11

- Animation
- **Generative AI Focus:**
    - Tools: VEO 3, Runway Gen-2, Pika Labs
    - Topics: AI-assisted animation creation, motion synthesis
    - Integration: Converting AI videos to animation sequences
- Friday: Show and Tell - Students present progress

Week 12

- Audio

- In-class instruction
- Friday: Show and Tell - Students present progress

Week 13

- Profiling/Optimization
- **Generative AI Focus:**
    - Tools: Suno AI, Audacity
    - Topics: Creating game soundtracks, sound effects, and ambient audio
    - Integration: Optimizing AI-generated audio assets for game performance
- Friday: Show and Tell - Students present progress

Week 14

- Game AI Basics
- In-class instruction
- Friday: Show and Tell - Students present progress

Week 15

- Final Projects
- Presentation and evaluation
- Final Show and Tell - Complete project demonstrations

Course Software

For games and real-time 3D, students will choose one of two game engines:

- **Unreal Engine**: Download and install the free and open-source Unreal Engine https://www.unrealengine.com/en-US/
- **Unity 3D**: 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 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

nikbearbrown YouTube Channel

Over the course of the semester, I'll be making and putting additional data science and machine learning related videos 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 TAs. If they can't answer them, then the TAs 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. 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 for grading. Every assignment, project, and exam you are expected to complete is assigned a point value, ranging from 1 to 1000 points.

- Late submissions incur a 10% deduction per day, rounded up.
- Exams cannot be made up unless prior arrangements have been made.

AI-Based Grading Approach

Due to the widespread use of Generative AI, grading is adjusted as follows:

Students scoring 80% or higher (absolute scale) will be graded relatively based on class performance:

- Top 25% (rounded to the nearest integer): A
- Next 25%: A-
- Next 25%: B+
- Final 25%: B

Students scoring below 80% (absolute scale) will be graded based on the traditional absolute grading scale below:

Score	Grade
78–79	C+
73–77	C
70–72	C-
60–69	D
Below 60	F

Important: Students below 80% cannot earn a grade higher than B-, even if the relative curve would otherwise place them higher.

Notes:

- The instructor reserves the right to make minor adjustments for fairness.
- Students are encouraged to complete all work independently and ethically, especially in light of Generative AI use.

Canvas

You will submit your assignments via Canvas. 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

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.

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.

AI Policy for Coursework

Use of AI in Assignments

The use of AI tools (such as ChatGPT, Claude, GitHub Copilot, etc.) is permitted in this course, with the following requirements:

- All AI usage must be properly cited
    - Include the name of the AI tool used
    - Specify which portions of your work were AI-assisted
    - Describe how the AI was used (e.g., generating code, editing text, brainstorming ideas)

Demonstration of Understanding

- Students must be able to explain any AI-generated content in their submissions
- Teaching Assistants or the Professor may ask students to walk through and explain any part of their work
- Inability to demonstrate understanding of submitted work may result in grade penalties

Academic Integrity

- Students are responsible for all submitted work, regardless of how it was generated
- AI should be used as a tool to enhance learning, not to bypass it
- Using AI without citation constitutes academic dishonesty

Quality Expectations and Grading

Due to the widespread availability of AI tools, a relative grading component (20% of total points) will be applied to all assignments worth 100 points or more. This component evaluates your work compared to peers, with emphasis on:

- Originality and creativity beyond AI-generated content
- Depth of understanding demonstrated
- Customization and personalization of solutions
- Real-world applicability

Quality Score Breakdown (20 points)

- **Bottom 25% (5 points)**
    - Meets basic requirements but lacks depth or real-world relevance
    - Basic implementation with minimal customization
    - Limited error handling and agent interaction
    - Superficial documentation and demonstration
- **26-50th percentile (10 points)**
    - Solid implementation with thoughtful agent design
    - Functional tools with appropriate integration
    - Basic error handling and memory implementation

    - Clear documentation and demonstration
- **51-75th percentile (15 points)**
    - Strong technical implementation with clear real-world applications
    - Sophisticated agent interactions and tool integration
    - Comprehensive error handling and memory management
    - Professional documentation and compelling demonstration
    - Evidence of testing and performance optimization
- **Top 25% (20 points)**
    - Exceptional project demonstrating innovation and technical excellence
    - Novel application with demonstrable value
    - Advanced agent orchestration with sophisticated decision-making
    - Unique custom tool that significantly enhances capabilities
    - Production-ready implementation with attention to scalability
    - Outstanding documentation and presentation

Best Practices for AI Use

- **Use AI as a learning tool**
    - Ask for explanations of concepts you don't understand
    - Request alternative approaches to problems
    - Use AI to review your work and suggest improvements
- **Maintain your voice and perspective**
    - Edit and refine AI-generated content to reflect your understanding
    - Add your own insights and observations
    - Ensure the final work represents your learning and knowledge
- **Document your process**
    - Keep a record of your prompts and the AI's responses
    - Note any modifications you made to AI-generated content

- Be prepared to discuss your collaboration with AI during evaluation

By following this policy, you can ethically leverage AI tools while ensuring they enhance rather than replace your learning experience in this course.

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.

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/) 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/. 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/.