

CSE 3541, Final Lab Assignment

Learn Something, Have Fun

Autumn 2021, Prof. Roger Crawfis

This lab is an open assignment for you to learn more about something you are passionate about (or slightly interested in) related to the course topics: Computer Game and Animation Techniques. You may want to focus on a project that better prepares you for a follow-on course like CSE 5912 or CSE 5542. You may want a more technical and research-oriented project to enhance your graduate school application. You may want to create a game or animation to showcase on your portfolio. In any case, determine something that satisfies the requirements and a life goal (aka whatever you do, do it with a purpose).

It can be an individual project, or a group project of up to 3 people. Use Piazza to find your teammate! It is up to you to decide the content and the type of interactions in your application. Following a more complex (set of) tutorial and producing a polished product is fine. If the tutorial has a completed download, then document that and show how you used different models and/or extended the tutorial.

Requirements

Your package must be written in C#/Unity and it must contain a 3D scene. You are encouraged to use any asset from the Unity store. Please document them in the final submission. You will present / submit a proposal for class (and instructor) critique / help. Your final project will be evaluated based on the criteria:

1. Challenge - is the amount of work appropriate to 2 weeks of work outside of class? Is appropriate for the team size.
2. Results and Effort - is your implementation complete? If not, you can partially compensate by writing a short report (1 page or less) documenting difficulties encountered and any changes you had to make to your lab.
3. Timeliness - factored in as the lateness penalty.

Extra credit may be given at the discretion of the instructor based on results shown during the final demo time slot. Should you complete additional work beyond what you originally submitted by the deadline and demo it, partial credit for unearned points may also be awarded.

Schedule

Proposal

Each person/group should submit a project proposal as a PDF file by the due date (late proposals will receive a zero). The proposal should cover three aspects of this project: 1) a set of example images to demonstrate what you plan to create (pencil drawn); 2) the tools and techniques you plan to use to reach your goal (including amount of programming); 3) an initial set of tasks; 4) a project timeline; and 5) if a group project, you should also explain how the work will be split.

Short Presentation

Please prepare a short presentation to show your work. This presentation will be either a video recording, or a live talk probably during our regular class time as well as the final exam time slot. Please consider completing the presentation within the defined number of minutes. Show your result and explain what you did and what you learned. Although you still have time to refine your work before final submission, please make most of your system working so other people can give comments.

Final Submission

Please submit a demo video, and a PDF file by the due date (late submissions will receive a zero). If the file is too large, please consider providing a link for the instructor to download it. In the PDF file, please show: 1) the example images; 2) the techniques you used in the project and how the application works; 3) some image results; 4) use and credits of any prior labs, assets, or tutorials. If it is a group project, please also explain how the work was divided.

Ideas

1. Advanced Development (for advanced students and those wishing to go to grad school)
 - a. PCG or Game Research Task with Dr. Crawfis
 - i. Tiling Framework
 1. Automatic tileset generation
 - a. Terrain tiles
 - b. Dungeon tiles
 - c. Racetrack tiles
 - d. Maze tiles
 - e. Rooms and buildings
 - f. Occupancy grid tiles
 - ii. Spawner framework
 1. Use cases
 2. Object Pooling
 3. Tileset generation
 - iii. Board / Card games
 1. Animations for claim a space and winning a space
 2. Animations for winning and losing
 3. Better / worse AI players
 4. Board Generation
 5. Complete Game (w/GUI, High Score, animations, etc.)
 - a. 3D Tic-tac-toe
 - b. Toss Across
 - c. Treblecross
 - d. Ultimate Tic-tac-toe
 - e. Checkers
 6. Expanded framework
 - b. Advanced Technology
 - i. Kinematic demo
 1. Extension of Kinematica-Demos on github
 - ii. Ai-Planner, GOAP, Minimax trees
 - iii. VFX

- iv. Shader Graph
 - v. NavMesh Agents
 - vi. Inverse Kinematics
 - vii. HDRP lighting
 - viii. Terrain generation and tools
 - ix. Physics simulations
 - x. Ragdoll Physics
 - xi. WebGL or cell phone builds
2. Game Development
 - a. Project Tiny Samples
 - b. Quickstart-arcade
 - c. GUI
 - d. Cinemachine
 - e. Post-Processing
 - f. HDRP or URP
 3. Tech Demo
 - a. Opening Doors
 - i. <https://www.youtube.com/watch?v=MYKHRYaUIJM>
 - ii. <https://www.youtube.com/watch?v=fi4G6tMYeCw&list=PLqu5DuP51vTpvNGfDTf3fIZvYIHTb7Avp&index=36>
 - b. Virtual puppeteering
 - c. Use springs and physically based animation to create blobby or cloth-like objects.
 - d. Create a realistic urban crowd simulation that includes pedestrians and vehicles.
 - e. Procedurally create video game levels.
 - f. Procedurally create fluid surfaces.
 - g. Simulate fluid dynamics for particles or volumes (note: this can be a lot of work if you also want to do surface construction of volumetric data).
 - h. Create an articulated figure and procedurally animate it's linkages in an interesting way (forward kinematics).
 - i. Use (or implementation) of inverse kinematics algorithm
 - j. Machine Learning for X (game testing, PCG, learning to parkour, ...)
 - k. Combine and/or extend the features from the earlier labs.

References

<https://learn.unity.com/project/unity-for-animation-road-to-realtime-live-series?uv=2019.4>

<https://www.youtube.com/watch?v=DInV-jHm9rk>

<https://www.youtube.com/watch?v=XjMKbEIVNmg>

https://www.youtube.com/watch?v=F_Zn-B_8Uh0&list=RDCMUCgd3l8iA5zBYVa4sQ6-ONFw

<https://www.gabrielaguilarprod.com/product-page/unity-vfx-weapon-trails-project>

<https://github.com/Unity-Technologies/ai-planner-samples>

<https://github.com/Unity-Technologies/ProjectTinySamples>

<https://github.com/Unity-Technologies/VisualEffectGraph-Samples>

<https://github.com/Unity-Technologies/quickstart-arcade/tree/dev/Assets/Templates>

<https://github.com/Unity-Technologies/quickstart-arcade/tree/dev/Assets/Templates>

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

<https://github.com/Unity-Technologies/NavMeshComponents>

<https://www.youtube.com/watch?v=6xs0Saff940&list=PLImQaTpSAdsBUM6-deTd4A82UeT1T2E4Q&index=1>

<https://www.youtube.com/playlist?list=PLKUARKaoYQT1H1jOpLmuowo1d03sGrf3T>

https://www.youtube.com/watch?v=jUdx_Nj4Xk0&list=PLFt_AvWsXI0fnA91TcmkRyhhixX9CO3Lw&index=24

<https://www.youtube.com/playlist?list=PLPV2KyIb3jR4u5jX8za5iU1cqnQPmbzG0>

<https://www.youtube.com/watch?v=PRrClxTifB4>

<https://www.youtube.com/watch?v=4Kj6YUPLWCw>

<https://www.youtube.com/watch?v=qmdC7kydWa4>