Faculty of Science

Course:	CSCI 3090U: Computer Graphics and Visualization
Assessment:	Final group project
Topic:	Animation, Simulation, Visualization, or Ray Tracer

Overview

This group project is designed for you to demonstrate the skills that you have learned in this course. The final project that you submit in the last week of classes will be a completed animation, simulation, or ray tracer program. You are expected to *work in a group of two students* when completing this project, but some exceptions can be made in rare cases, allowing for larger groups (with instructor permission) provided that the amount of work is increased accordingly.

***Note**: Any projects from individual students will not be accepted, except if special permission has been given by the instructor in advance.*

The entire project will be maintained in Git (e.g. GitHub). Essentially, your submission will consist of access (i.e. permission granted to the instructor and URL submitted) to your Git repository. The instructor will then use this access to download the latest version of your project, along with other information (e.g. commit logs) available through Git.

Detailed Instructions

Choosing a Topic

The project topic is, for the most part, up to you. Your primary goal is to implement as many of the course concepts as possible into your project. For students having trouble coming up with a project, here are a few ideas:

Animations

- Animated company logo
- A cinematic from a game
- Flocking birds, schooling fish, stampeding animals
- Kinematic simulation (e.g. humanoid walking, robotic arm/Canadarm)

A Basic Game

- A drivable car
- A flyable spaceship or airplane
- A ping pong or breakout-style game

Simulations (assuming you already understand the physics and math behind it)

- Rigid body simulation (e.g. a house of cards, rag doll, cloth, solar system)
- Soft body simulation (e.g. human body impact)
- Fluid dynamics (e.g. pouring water from one container into another)

Visualization

- Visualization of static data, e.g. in a carpet plot

- Visualization of dynamic data, animated
- Visualization of volumetric data

Ray Tracers or Similar

- Ray tracer with reflections and refractions implemented recursively
- Physically-based renderer
- Build your own rasterizer, e.g. using the GPU and/or parallelism within the CPU

Basic Requirements

For animations, games, and visualizations, topics such as lighting, shading, texturing should also be considered. Your application should run (on a reasonable machine) in near real-time.

If your project is a ray tracer, it should produce more realistic effects than the real-time graphics produced by OpenGL, DirectX, or Vulkan, but it doesn't have to run in real-time. Another challenge you could tackle with a ray tracer would be to compute ray-triangle intersections, and use that to render traditional triangle meshes (e.g. an .OBJ file), textures, etc.

For either type of project, generated geometry or paths would also make sense. Curves could be generated for animation paths, or used for extrusion/sweep operations to generate more complicated geometry. Similarly, procedural textures could be used in either type of project.

How to Submit

To submit this project, please push all your work to your Git repository, and submit access to that into the drop box for this assignment on Blackboard. It is your responsibility to ensure that all necessary files are included. If a file is missing from the repository, you do not get credit for it. Most groups will use GitHub, and thus can merely grant permissions to the instructor's GitHub account (`randyfortier`).

When your project has finished, submit the names of all group members and the GitHub URL to the Blackboard drop box.

***Note**: Do not submit a .zip file to Blackboard, containing all of your code. Use of git (either via GitHub or BitBucket) is required.*

Evaluation

When evaluating this project, the instructor will attempt to give a metric to the amount of work involved. This metric will be affected by the size of your group (i.e. what will be evaluated is the average work done per group member). The project will also be evaluated through several other metrics, including aesthetic/artistic components (where applicable), code quality/comments, and performance.

The entire project will be evaluated subjectively, so put your best effort into it if you'd like a good mark. No assignments were included in the course evaluation so that you would have sufficient time to dedicate to the final project. Therefore, use your time wisely and start your project early on.

***Note**: The project must include functional build scripts based on those used in the in-class examples or there will be a significant penalty applied.*

***Note**: Any instances of plagiarism will result in the student(s) receiving a mark of zero for the project, and further disciplinary action will be taken. Plagiarism includes, but is not limited to:*

- *Copying of (any amount of) work from the Internet, without proper citation*
- *Submitting a body of work, cited or not, that is primarily not your own work*
- *Copying of (any amount of) work from another student, past or present, without proper citation*
- *Allowing your own work to be copied by a fellow student*