Computer Graphics Final Assignment

– Putting it all together –

COMP 4302/6909

Due: March 27, 2024 (Wednesday) 11:59 PM

Group Self-Registration Deadline: March 22,2024 (Friday) 11:59 PM

Assignment goals: To demonstrate the skills and knowledge you have obtained through the course by showcasing a WebGL 3D graphics application based on a theme of your choice. Alternatively, you may produce a Three.js, Unity, or an Unreal demo that performs the same job as the WebGL assignment.

Available Themes:

You can submit one the following project types as your final project for the graphics course:

- A virtual-museum walk-through type of environment that allows for a virtual tour, for example of Downtown St Johns, or the Rooms Museum, or some art galleries from your hometown or country.
- A one or two player game or a 3D implementation of a basic 2D arcade game in a retro-style implementation, but that displays elements that are very specific to, or customized to the members of the group.
- A 3D simulation of a process that is completed in a factory style, in the type of a cinematic illustration of a process, or sequence of events, as in an animated scene (industrial simulation).
- A 3D system for walking through a virtual building (alternatively, a walkthrough of some part of the MUN campus), that includes some interactive elements.
- A sophisticated 3D interactive tutorial to illustrate concepts from the course (such as an edutainment-style demo such as the ones from the materials shown at the end of the slides).
- A museum of Graphics methods and algorithms, where each room presents a technique related to graphics, such as texture mapping, ray tracing, animation of characters, use of A.I., etc.

In case of doubt, it is advisable to check with the professor and discuss the project that you intend to do before the actual implementation to ensure it will meet the requirements and to help you assess the level of complexity of the project so that it is neither too simple nor too complex.

All submissions must have the following technical features:

1. A properly illuminated scenario all the way through. Two or more light sources must be activated to illuminate the scene.
    a. At least one of the lights must be switchable using keyboard input (pressing 'l' as a toggle button).
    b. The system must allow changing one light position and color using the keyboard.
    c. Varying materials, colors and use of textures is expected. For example, in a museum application with several rooms an extensive use of textures is expected.
2. A system for efficient, intuitive navigation must be presented, using the lookAt() function or a ModelView matrix equivalent, using both keyboard and mouse functionality.
3. Use of textures on the models. For instance, paintings in a museum would be textures.
4. At least one part of the program, where elements are moving in a hierarchical relationship with other elements, in the form of platforms carrying other objects, a solar system, or a pair of objects rotating around each other, illustrating the use of cumulative transformations.
5. A "novel component": A graphics technique that has not been part of the previous assignments. For instance, a method that might have been described as an "advanced topic", a research topic, or something that you found interesting, such as particle systems, game A.I., new shading effects, something out of the graphics research literature, such as collision detection between multiple objects, interactive model deformation, etc. An important consideration is that it must be appropriate with respect to the context in which it is presented. This is in the sense that it does not look completely different or conceptually detached from the other elements of the assignment, but it must be suitable to the theme you have chosen.
6. Video requirement: Produce a 5-minute video that is designed to be distributed to the public, where you explain how to achieve the features of your assignment. You should even feel free to post your submission directly on a free-access video platform and provide the link. The winners of the competition (up to three groups may be chosen) will earn 10% extra points, will be featured in class, and will earn bragging rights. A source of inspiration can be found in the Guerrilla CG videos:
    https://www.youtube.com/channel/UC5fpWfCQ95VFghlkH1RG70w

Project Presentation:

All projects submitted will have to include the video submission (required element), which could be published on the instructor's homepage, labelled as submissions for Comp 4302/6909 with identifying information and/or shown to other students during the last week of classes.

Evaluation Scheme:

Clarity and legibility of the source code, indentation, comments & readability.y.....	5%
Complete Documentation, instructions for deployment and video.....	15%
Successful use of textures	15%
User control over lighting conditions	10%
Effective interface and/or use of keyboard or mouse for navigation	15%
Overall degree of sophistication of the "novel component"	5%
Successful implementation of the "novel component"	5%
Appropriate integration of the "novel component"	5%
Relative quality and sophistication of the overall project (competitively graded) ...	25%

Further Details:

- The projects are to be finished individually or in groups of 2. Groups of 2 will need to self-register as a group on D2L. Individuals will need to self-register in a group that has no partners (an empty group) so that they become the only member in the group. It is advised individuals register in groups at the end of the list of groups. In addition, you can email the course instructor to confirm you wish to submit individually.

- Late submissions are allowed. For each day of a late submission 15% of the grade will be removed. You may submit your assignment up to 6 days late (in which case, the assignment grade would be regular grade minus 90%).

- All members of the group must present their own project in person. Projects will be presented in class during the last week of the course as per the course outline. The schedule of presentations will be posted on D2L. The class will be able to rank their favorite presentations during the last week of classes. Only those present in class may score other groups.

- For the competitive grading portion, a certain percentage of submissions will receive an a A (10-20%), B (20-30%), C (20-30%), D (5-20%), or F (0-10%), depending on their standing with respect to other submissions.

<u>Assignment Submission Declarations:</u>

- To avoid any confusion, in the final assignment you will want to explain to the TA very clearly what options you are submitting and the total grade you are aiming for.
- If you are using Unity or Unreal, you will also need to provide the link to the project source code you have completed if your project was created using Unity or Unreal. You can document this in the cover page of your assignment.
- If you use three.js, write a note to the TA about the fact that your implementation requires three.js with clear instructions on how to install and evaluate the submission.

- After your text responses, provide a separate page titled "References & Sources", documenting all external sources, including Web resources and AI programs or assistants (such as Chat GPT). Even if there were no external sources used by you, the declaration must be part of the assignment, and in such a case, it must contain at least this statement "No external resources except those provided by the instructor were used to produce this assignment."

- Failure to provide any of these declarations above will result in an invalid submission for the part where the documentation is missing. For example, if the whole submission is in a Game Engine and the Github or google drive link to the submission is not provided, then the whole submission will get 50% of what a properly documented submission would get (ordinary grade * 0.5).

What and How to Hand in:

You are handing in the source of your program, as well as any data files required for running your program. Your source code must contain sufficient internal documentation to facilitate grading. This includes your names, student numbers, a brief description of what the program does, which items you claim to have completed and a listing of known bugs, if any, at the top of the file. Send in your source program(s) through the Direct2Learn Dropbox as a single .zip file.

- A three-to-five-page documentation of your submission is required. The description must contain:
    - A cover page with name(s), student id(s), project title & submission date.
    - The description of your implementation features and characteristics.
    - A user manual, which lists the ways of interaction with the program submitted, including instructions on how to use the application.
    - A statement of acknowledgment of sources and references where you clearly specify which parts of your code were done by you, and which parts of the code come from third-party libraries (such as libraries to load models, sounds, open source code, etc.)
    - At least three screenshots of your application (showing the highlights of how it should look like when being reviewed) in not more than one page.
    - If your project contains special features that need to be checked out by the markers, include a mention to these features along with the instructions on how to test these feature for the markers

Documenting and referencing your sources:

To avoid potential plagiarism issues, always insert a disclaimer under a section called "Sources & References" explaining whether you adapted source code or used implementation information from any source, including the textbook's author website. Any source that is external to your own coding must be mentioned. All the source code submitted must be understood by all members of the group and each member must be

able to explain in detail his/her own part of the project if this is deemed necessary by the markers or the professor after submission.

Verifying your submission:

Once you have placed your assignment into the Dropbox, click on the link that indicates your submission has been done, download it, open it, revise the contents, test that the program you have submitted actually is the one intended (not the source files, for example), and if it does not correspond to what you want the TA to mark, you must resubmit again, until you are satisfied. The TA will only mark the last submission by any member of the group using the timestamps shown in the D2L folder.

Link to GitHub or Google drive repository for Game Engine submissions. In the case that you are submitting a Unity or Unreal project, it is not practical to have the whole project sent through D2L, as this could require a couple gigabytes of storage. In these cases, you are asked to provide instead a link to the Github or google drive repository where each part of the submission is found. To reduce the chances of having your work plagiarized by someone else, your links must be private and give only access to yourself and the following two GitHub accounts (instructor & TA): omerpas & AiurNanzatov or, for access to a google repository: omeruvia@mun.ca & aznanzatov@mun.ca