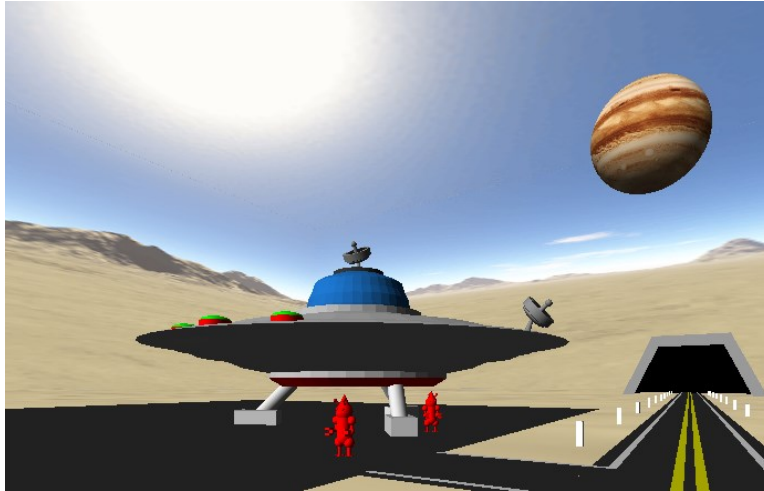COSC363: Computer Graphics
Assignment 1
Alien World
(Max. Marks = 20;   Due date: <u>28 Mar 2024</u>)

1. The Scene

On a remote planet, a few aliens are preparing a spaceship for liftoff.

2. Minimum Requirements (Max. 7 Marks)

In this assignment, you will develop an OpenGL program to display your version of the above scene. The minimum requirements are outlined below.

2.1 A model of a spaceship: This need not be a highly complex model. It could be constructed using a set of GLUT objects. The spaceship should display some animation (e.g. rotating parts, flashing lights).

2.2 Alien: Alien models may also be constructed using GLUT objects. They need not necessarily have humanoid characteristics. Each Alien should perform some action such as continuously moving along a path, pushing an object, etc.

2.3 Lighting: The scene should have proper lighting so that the models and animations are clearly visible.

2.4 Camera: The user should be able to move camera through the scene using the standard set of directional keys: E.g. Up arrow: move camera forward in the current direction, Down arrow: move camera backward in the current direction, keeping the look direction same, Left arrow: Turn left by 5 degs, Right arrow: Turn right by 5 degs.

2.5 Textures: At least three different textures must be used in the scene. It is not necessary to texture-map all surfaces. Please do not use very large images (>10MB) as textures. A skybox texture set, if used, will be counted as a single texture.

3. Extra Features and Marks (Max. 8 Marks)

The assignment will be marked out of 20, with 7 marks allocated for minimum requirements, 3 marks for the overall rendering quality, and 2 marks for a report (see below). With just the basic features given in 2.1–2.5 above, your assignment would earn at most 12 marks out of 20. You will need to add a few more features to the scene to get additional marks. Some possible features, and the approximate marks they would each gain if implemented correctly, are listed below.

3.1 Planar shadows cast by at least two objects (1 mark)

3.2 A spotlight on a moving/rotating object (2 marks). The movement of the spotlight should be clearly visible.

3.3 A Sky box (1 mark)

3.4 A custom-built sweep surface (a surface of revolution or extrusion) constructed using quad strips (2 marks). This will require evidence in the report detailing how the surface was generated.

3.5 A surface shape generated using a mathematical formula (e.g. paraboloid). (1-2 marks)

3.6 Physics models (e.g. gravity, trajectory of a projectile). Please give relevant equations in your report. (1-2 marks depending on the complexity of the model).

3.7 Spaceship liftoff: If implemented, spaceship liftoff must be defined as an animation initiated by pressing the space bar on the keyboard. (1 mark)

3.8 Particle systems (2-4 marks)

The marks associated with each feature is indicative of the time and/or effort required to implement that feature. The above list should not be taken as a complete list of the only features that can be implemented.

4. Rendering Quality (Max. 3 Marks)

In the context of this assignment, the term "rendering quality" refers to the visual aesthetic quality of the models and animations displayed by the program. Slow and jittery animations, incorrectly rendered models and shadows, improper movement of the camera, and improper illumination may cause marks to be deducted for poor rendering quality.

You will gain marks only for models/animations that are correctly and clearly displayed on the screen. A particular feature may be implemented correctly in code, but if the corresponding output is not clearly seen on the screen, you will not receive marks for that feature.

5. Report (Max. marks: 2, Max. number of pages: 4)

The report should include the following:

- Your name and student number
- A brief description of the scene (1 paragraph).
- At least two screenshots showing important aspects of the scene or animation
- An itemised list of *all* extra features implemented.
- Images of diagrams, sketches used for designing models (e.g. the base curve of a sweep surface).
- Equations if any, used for designing surfaces and animations.
- A full list of control functions (keyboard, mouse, special keys) defined for interacting with the scene.
- Build commands or instructions for compiling and running the program. If the program was developed using Visual Studio on a Windows platform, please mention that in your report.
- References to external sources of textures, models, algorithms, equations etc., used in your work.
- **Important!** The report must contain the following declaration. **Assignment submissions without the declaration will not be accepted**.

Declaration
I declare that this assignment submission represents my own work (except for allowed material provided in the course), and that ideas or extracts from other sources are properly acknowledged in the report. I have not allowed anyone to copy my work with the intention of passing it off as their own work.
Name:
Student ID:
Date:

You may include more than 4 pages in the report, only if absolutely necessary. Please submit your report as a single **PDF** file.

6. Models and Animations

The object models in the scene may be constructed using any of the following methods:

- By transforming and combining a set of (at least three) GLUT/GLU objects such as spheres, cubes, cylinders etc. to form a composite model.
- By using a set of polygonal surfaces comprising of quads and triangles.

- By using surface generation methods such as sweep surfaces, surfaces of revolution etc.

You are not required to design highly complex mesh models for this assignment. Using objects modelled using design packages such as 3DS Max, Blender, Lightwave, etc., or downloading pre-built mesh models from the Internet will not give you any extra points.

You may use parts of lab code and resources (models, images, image loading functions) provided in the course. Models and animations developed in the lab will give you marks only if significant changes or enhancements have been made to them.

7. Program Development:

- Please do not use source codes of programs from online repositories, tutorials, and books.

- Develop your program in C/C++ language using only OpenGL 2 API. Please do not use OpenGL 4 code (vertex/fragment shaders etc.), or extensions (e.g. ARB, EXT etc.) that are not part of the standard OpenGL-2 API.

- **Please make sure that your program can be compiled and run on CSSE lab machines.**

- Your submission will <u>not</u> be marked for code readability or optimization. However, you are encouraged to prepare well documented and readable code.

8. Assignment Submission

Submit electronically (using *Learn*), the source code and all supplementary files (models, images etc.) needed to run the program. Please also submit your report in PDF format. The files should all be packaged together and submitted as a single .zip file.

This is not a group project. Your assignment submission must represent your own individual work. In particular, students are not permitted to share program source code in any way.

9. Miscellaneous

1. Check regularly on the *Learn* system forums for spec updates and clarifications.
2. Standard departmental regulations regarding dishonest practices apply.