Winter 2022: CSI4130
Assignment 1
Due: Saturday, Mars 12th, 2022, 11:55 pm in Virtual Campus
University of Ottawa - Université d'Ottawa

1   Animation of 2D Shapes [9 in total]

In this assignment, you will build a 2D shape from a collection of triangles and implement a pre-determined animation in Javascript with WebGL. You will also create a similar animation in Three.js. The goal is to familiarize yourself with the elementary modelling and rendering processfor computer graphics. **This is an individual assignment.**

2   WebGL

For this part of the assignment, you are working directly with WebGL. You are not allowed to use any library except gl-matrix.js and the helper files from the lab webgl-utils.js webgl-debug.js cuon-utils.js in your solution. You are advised to base your solution on the lab ''Colored Tetrahedron''.

2.1   Shape [3]

The current example uses a tetrahedron that is spinning. Change the shape to 2D and use your initials (You only need to use two letters independent of middle-names or multiple last names). Note that you can simply duplicate the setup for each letter and use two draw calls to accomplish drawing one letter after the other. This is made particularly convenient with VAOs in WebGL2. The structure of your program would then look as follow.

Setup:
Get first VAO
Define vertices and triangles for your first letter
Set the vertex buffer objects for your first letter
Disable first VAO
Get second VAO
Define vertices and triangles for your second letter
Set the vertex buffer object for your second letter
Disable second VAO

Drawing: Enable
first VAODraw
first letter
Disable first VAO
Enable second VAO
Draw second letter
Disable second VAO

2.2 Animation [1]

The spinning animation of the Tetrahedron is to be changed such that the initials spin clockwise around the center of your viewport. Please keep your initials upright at all times.

3 Three

Your animation will have to follow a very similar approach to the Solar animation lab. To this assignment, you are asked to use basic geometric shapes directly available in `Three.js` such as boxes, cylinders, and spheres.

You are not allowed to use any other library except `Three.js` and `dat.gui.js`.

3.1 Desk Lamp [3]

Build a simple stick figure resembling a desktop lamp. If you like to be inspired watch the first Pixar short called Luxo Jr from 1986. Your stick figure must be built as a scenegraph in `Three.js` with at least separate primitives, i.e., cylinders, spheres, or ellipses. An example of desk lamp is shown in Figure 1. The minimum number of primitives are one for the support, lower arm, upper arm, two for the lamp shade and one for the bulb plus 3 spheres for the three respective joints. Your program must show the desk lamp in a pose like Figure 1 and from a corresponding viewpoint. The desk lamp shown in the picture is just a simple example, feel free to make your lamp fancier. Provide controls with corresponding sliders in `dat.gui` to control the arm joints and the lamp head by the user. The lamp head must be able to move up/down and right/left. Place the lamp in the middle of the window.

3.2 Camera Control [2]

Use `dat.gui` sliders for two angles to control the pose of the camera and for the position in 3D where the camera is directed to (the *at* position). The angles must be rotating the camera in a sphere around the *at* point. Select the radius of the sphere large enough such that it is outside your scene. In other words, your perspective camera will always need to look at the *at* point and your rotational controls will need to move your camera on a sphere around the *at* point. In order to make the control more intuitive, use `dat.gui` to turn on the display of a small sphere shown at the *at* point.
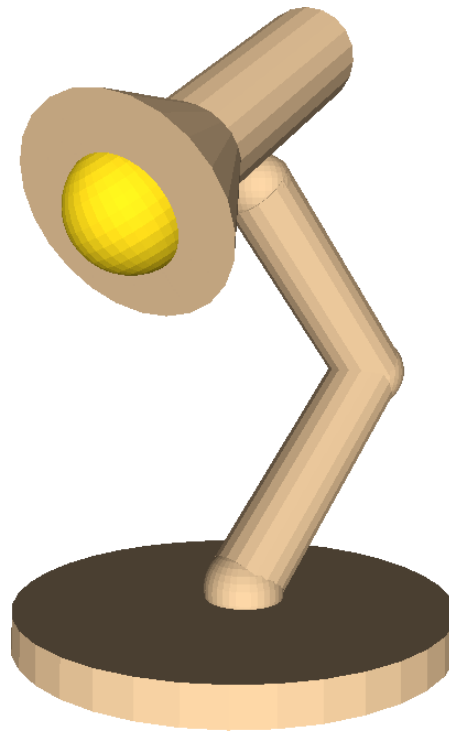
Figure 1: Sketch of a Desk Lamp.

3.3   Bonus: Animation [3]

Use a similar strategy than in the Solar example to make the arms and lamp head move for the lamp to jump. For inspiration, you may have a look at Luxo Jr. by Pixar. Use `dat.gui` to give the user the option to switch the animation on and off. This is a bonus question and hence marks will be given only for successful (i.e., convincing) attempts.

4   Submission

Your assignment submission must consist of your Javascript and html files. As you are working with the current version of `Three.js` and `dat.gui.js`, you will not submit these.

Filename
initials.js
initials.html
desklamp.js
desklamp.html

You must submit the files listed above and no library files via Virtual Campus.