Kyle Raja

CS 330

2/19/2023

Project Reflection

*Justify development choices for your 3D scene. As you write, think about why you chose your selected objects. Also consider how you were able to program for the required functionality.*

I wanted to use a cube, cylinder, pyramid and sphere. However, I found it difficult to add multiple objects to my plane. As I added different objects, it would actually attach to other objects, not make them separate. I did some research online and the idea behind it is, creating an array and storing those objects inside that array. However, after creating my array, adding important things like texturing and lighting, did not work very well with the array. I decided to stick with the cube object, apply the texturing, and light source. Cube’s are a relatively easy object to create in the GLfloat verts[] array. The texturing looked good (wood style) on it so I also applied the texture to the plane, a nice wooden floor appearance. The object and plane were not very difficult to create, also the light source was not too complex. I found the greatest challenges were applying multiple objects and multiple light sources. I tried to create another light source on my cube, however I believe the light sources need to be in an array also to affect one object. Also what was strange is, my light source, a cube, also seemed to apply a plane for it also. I tried to remove the plane, however I believe it is using the verts[] array in the light source. I did find how to change the lighting of the source and set the light to a more neutral level so the texture would be visible.

*Explain how a user can navigate your 3D scene. As you compose your thoughts, discuss how you set up to control the virtual camera for your 3D scene using different input devices.*

I made sure to apply the 3D scene to my object. The user can use the A,S,D, and W keystrokes to move the camera left, right, up, and down. We can zoom into the object or zoom out. The mouse can also control the camera movement however, I found the keystrokes to be more reliable. This is, I believe, called the ProcessKeyboard and UMouseScrollCallback methods that control these camera movements. With these controls, we can view the plane 360 degrees. I made sure to apply the X, Y, Z coordinates to the 3D world.

*Explain the custom functions in your program that you are using to make your code more modular and organized. Ask yourself, what does the function you developed do and how is it reusable?*

One thing I really liked was the texture folder where I could store my .png files to texture the object. It made it very easy in the code to reference a different .png file if I wanted a different texture. I think keeping the resource folder on the main level above the module folders is a good idea in case I wanted to apply the texture on another module. Regarding the light source, I liked the gLightPosition and gLightColor functions in particular. It made things simple to alter how my light source behaved. Also gIsLampOrbiting was a nice variable as well in case I wanted to have my light source stop orbiting the object and just remain stationed. Probably one of the most powerful and useful functions was the GLfloat verts[] array. Having my objects in that array made things easier to manipulate the objects and plane. It’s good to have one area where the objects can be formed and textured. I also made sure to apply the X, Y, Z coordinates in the 3D scene.