CS330 Final Project Reflection

Kari L. Cheslock - 08/11/21

For my 3D scene, I chose objects from around my kitchen that I felt I could recreate using primitive shapes. Specifically, I chose a wooden spoon, a tomato, a lemon and a sugar canister made out of glass with a metal lid. I was able to recreate the wooden spoon using a cylinder and a box, the sugar canister was recreated with two stacked cylinders, and the tomato and lemon were recreated with spheres. The other reason I chose these objects was because the textures for each object would be unique and recognizable. I used a glass and a metal texture for the canister, a wood texture for the wooden spoon, a tomato texture for the tomato and a lemon texture for the lemon. I felt like the contrast between these different objects would make the objects recognizable when I recreated them in OpenGL. The other components of my scene were a plane that represented the table that the objects were resting on as well as a light source representing the overhead light in the room. I used the same metal texture for the table since the table in my original scene was also gray. The light source is a simple light cube, but I felt that this was sufficient to represent the overhead light in the room.

Navigation of this scene should be fairly intuitive, especially for anyone who has played any video games. The W key moves the camera forward within the scene, the S key moves the camera backward, the A key moves left, and the D key moves right. In addition to these keys, the Q key will move the camera up and the E key will move the camera down for added functionality. The user is also able to switch between perspective and orthographic projects by pressing the P key. Although these keys will allow the user to move around the entire scene, the code also contains functionality to move around the scene with the mouse. Swiping the mouse around moves the camera perspective around the scene. Also, the scroll wheel on the mouse will create a zooming in and out effect, similar to using the W and S keys. With all of these functionalities, the easiest way to navigate this scene would be to use a combination of the key presses and mouse movement. The mouse is certainly the fastest way to move around the scene; however, navigating with the W, A, S, and D keys will be better for the user to pinpoint the location of the camera to the desired location.

As far as modularizing the code, the main way that this code is broken up into reusable units is by having different header and cpp files for the spheres, cylinders and the plane that were rendered in the scene. The only other shape in the scene was a cube , which was easily rendered using an array of vertices and did not really need a separate header and cpp file. By having the code for generating a sphere, a cylinder and a plane in separate files, these files can then be easily reused to generate other scenes in other projects. The other main way that this code is modularized is by having each object generated in the scene, and all of its associated variables such as the VAOs, VBOs, and shaders, rendered by essentially copying and pasting the code and simply changing the variable names. Anyone looking at this code should be able to add or remove objects by following the same general pattern that has already been used. The last way that this code is modularized is by creating separate functions for the user to navigate the scene. The code has functions for processing input, meaning the various key presses that allow for navigation or changing perspectives, processing mouse movement, and generating the window in which the scene is rendered. Modularizing the code for navigating the scene is especially useful since this can be used for other projects without having to recreate it.

Overall, I am very happy with the way this project turned out. I feel like I could use this code as a template to create future projects and all I would need to do is modify the shapes and textures used to create a completely different scene. This project has served as a good starting point for learning OpenGL and for learning how to create more complex projects in the future. I feel like this project has taught me a lot of the basics that I need to create complex scenes, and I can go from here to learn the more complicated aspects of OpenGL and create bigger and more detailed worlds.