Ashley Santo

Final Project

The objects I chose to model were a water bottle, a spice jar, a pen and a book. The reason I chose these objects because I felt they had simple, basic shapes but also provided enough of a challenge for this project since they contain different shapes and textures as well. I will say, I thought I was understanding the modules pretty well the first couple weeks but towards the later modules, I ended up with exceptions being thrown that I could not solve and unfortunately had to leave. So, my project didn’t exactly turn out how I envisioned it but I think I did my best to get the gist of where I wanted to go with the design if I could have solved the error. For my objects, the water bottle was the only one where I used one primitive shape which was a large cylinder. The rest of the items all used multiple primitive shapes. The book used planes for the front and back covers and a cube for the pages, the pen used a cube for the body and a pyramid for the tip, and lastly, the spice jar used a cube for the glass and a small cylinder for the lid. OpenGL utilizes vertices that the program could then link together to create basic, flat triangles. Each vertex needed to have three attributes: the x, y, and z positional coordinates, texture coordinates and light coordinates. The x, y, and z coordinates give the position the vertex should be in, the texture coordinates define how the texture should wrap the object and the light coordinates define how the light should be reflecting off the objects. Once all the triangles are defined, they are then connected to create the multiple primitive objects seen in my project: planes, cubes, cylinders and a pyramid. The most difficult part was trying to draw a cylinder out of triangles. I think I had the right idea where I created two flat “circular” shapes out of six triangles and then connected them with planes but I don’t know if it executed as cleanly as I was hoping for. The rest of the objects were much easier to map out.

The base code for the camera file was borrowed from a previous assignment. I simply edited the camera header file to include the necessary movement. On program start up, the camera is set to an initial position that the user can then control to move around the scene. The input devices are mouse and keyboard. The user can use the WASD keys to move forward, back, left and right while the Q and E keys move the camera up and down. As the user moves around the scene, the keystrokes will communicate to OpenGL how it should update the viewport. There was supposed to be an easy switch between projections that I was having trouble figuring out but did leave as a comment in the code. The idea behind it was to have the projection initially set to perspective and then when the user enters P, it would change to the orthographic projection instead. The orthographic would allow the user to see the scene in 2D versus in perspective that is set in 3D.

The main customizable function in my code are the vertices and indices. These are used to map every point needed to create a shape and can be reused in different projects with a simple edit to the actual coordinates to match whichever item is being modeled. I organized this with labels that show where each point is located to simplify the process of figuring out how to draw the shapes. This made it easy to simply copy the code from one cube and paste it for another with changes to the coordinates. It made it much easier to visualize and map out. It also makes it easier to understand how the textures should look on the object and how the light should be hitting it. For example, by having a cube easily labeled top and bottom, I know how the light should hit it and how the textures should be wrapped. The next custom function would be the camera. The separate header file makes it extremely simple to reuse code from this project to another and just remap the controls and projections. Lastly, were the textures themselves. I think I showed how simple it was to add textures into the program with a couple name changes as well as file locations. I think most of the code within this project could easily be reusable, if not, at least a starting area for someone trying to learn and grow with OpenGL.