Michael Barbuzano

CS-330

8/14/2023

Final Project

The first object I wanted to implement from the image I chose early on in this course was the mug. I think the mug was a good complex object to work with but proved to be somewhat challenging. I created a cylinder for the mug’s base and then utilized a half torus to represent the handle of the mug. Although I think that the mug came out well, there are many things I could have improved on. I chose to apply a grey texture to the plane that all of the objects sit on, I intended it to appear more like granite, but I think I made some mistakes with my shader functions. The cutting board was a simple 3d object to create as it is just a wide cuboid that sits on top of the plane and adds a lot to the scene. I also implemented two cans of soup that were in the original image, as they were fairly easy to create and add to the scene. Finally, I added the cereal box that was present in the original image as that was also a fairly easy shape to replicate in this project. One object I omitted from the original image was the hot sauce bottle, as its shape is a bit obscure and would require multiple objects to create, I already created an object compiled from more than one basic shape, so I decided to omit the hot sauce bottle.

In the 3d scene I created, the user can traverse around with the W,A,S,D,Q, and E keys. The w key moves the camera forward, based on where the camera is currently facing, the A key, moves the camera to the left of its current position, the S key moves the camera backwards, and the D key moved the camera to the right. The Q and E keys control the height of the camera, the Q key moving the camera upwards and the E key lowering the camera. The user can move the mouse cursor to change where the camera is facing. The scroll mechanic on the mouse allows the user to increase and decrease the speed at which the camera moves.

The most reusable function in the code for this project is the URender function. I have not worked with OpenGL or 3d models before this course, but utilizing a single function for all of the rendering of a project seems efficient to me. I can imagine large projects might want to separate rendering functions, so that not all objects are rendered at the same time and can be called separately in the main function. But in this project, it made it very easy to add new objects as I would create the function declaring the vertices of the object, then I would add a “draw call” to the single URender function. This allowed me to make major changes to the 3d scene without changing any of the code in the main function. With the use of the URender function and in line code comments, I was able to organize all the 3d models “draw calls” in an efficient manner and add many changes to the 3d scene without creating a mess in the main function.