Dipesh Patel

CS – 330

Final Project Reflection

For this project we were to create or choose a scene with multiple objects to recreate in Visual Studio using OpenGL. The object I chose for this project varied in size, shape, and materials. This allowed a number of shapes and material types to be used in OpenGL to represent the objects. They are also common objects that someone might have lying around, so they are easy recognize. Most of the object chosen are also complex compound object, so they consist of multiple shapes to make the object. Once the shapes were identified it was a matter of implementing them in the project.

First I started with the knife object. That required making two cube shapes. By manipulating the vertex placement I was able to replicate the basic shape of the object. Then I made a plane shape to represent the table the objects were placed on. The next object I created was the flashlight. I started from the bottom and worked my way up creating cylinders of various sizes.

The watch was the next object I created. I started with the watch face. To create the watch face I created two cylinders, one to represent the lower part of the body and one to represent the upper part of the body. Next I created the watch bands using cube shapes. I used one vertex array to create both bands they would be easier to move and manipulate.

I continued with the charger next. I started with a cylinder for the centerpiece to get the size correct. Next I made a new shape, the torus. I had to rotate, scale, and position it according to the position and size of the cylinder. I worked on the charger cable next using one vertex array for the whole cable but rendering each separate section individually. I started from the section closest to the body of the charger and worked my way out creating each section. Lastly, I created the booklet for the user manual. It was a simple cube shape.

We needed to moved through the scene, so we could reposition our view and help with the creation and positioning of the objects. In order to move through the scene we needed to create a camera. First we needed to create movement for the camera. Here we use a first-person type system of the W, A, S, D keys for the forward, left, right, back movements. Next was the camera rotation. This was mapped to the mouse movement, so the camera rotates in the direction the mouse is moved. We needed to add extra movement to make it a bit easier to navigate the scene, up and down movement. For that the Q and E were bound to the vertical positive and negative movement. Next we need a way to adjust the rate the camera moves. This way done by binding the mouse scroll up and down to the camera movement speed. When you scroll up the camera movement rate goes up and when you scroll down the rate goes down.

The code was split into sections based on function. Texture, lighting, vertices, etc were split up for easier modification. Two of the functions that were modularized were the cylinder and torus creation. The main functions for these are located outside the main file. The function is called with parameters to create the shape in the main file and can be repeatedly used. Another function that has been modularized is the fragment shader code and lighting calculations. They can be repeatedly used without having to recreate the code over and over again. With any of these functions, they can be moved outside the main file to their own file and called within the main file without any loss in functionality and could be used in other projects as well.