Design Decisions

This project was very challenging for me to complete. Some of the requirements I was not able to complete, and I know I will get points off for this. This includes applying textures to the objects and the lighting for the scene. I also struggled with creating any 3D object besides a cube and a pyramid. I could not figure out how to draw the cone, cylinder, and spheres that I needed to create my complex objects. The plane was created using a cube with the y-coordinate being the same for each triangle to create a flat plane. The iPad was created using a cube and then scaling the cube to look like the iPad shape. I would have wanted to put spheres on the corners to make the iPad more rounded like in the photo. The Rubik’s cube was created using a cube. The air pods were also created using a cube and I would have wanted to have spheres in the corners to make it more rounded. The pencil was created by having a scaled cube as the body and a pyramid as the tip. I would have wanted to have a cone as the tip and a cylinder as the body.

The scene has intuitive controls for navigating the camera with movements that you would use in basic computer games. First the camera has controls to go forwards, backwards, left, and right. These controls are W for forwards, S for backwards, D for right, and A for left. There is also and option to go up and down. Q is pressed to go up and E is pressed to go down. The mouse is used in the scene to allow what is being seen to change without the position of the camera changing. The mouse cursor changes the orientation of the camera in the up, down, left, and right direction. The scroll wheel on the mouse allows the user to zoom in or out on the scene. There is also a button for ending the program which is set to Escape.

There are various functions used within the program to develop the overall scene. After the use of the namespace, shader files, and the main file we get to out functions used to draw the objects. For each object I used a render function for each to make it more organized to see what object is being rendered when and where. It was getting confusing for me to have it all in one render file as I could not get multiple objects to render in one render function. Each render function also has a destroy render function. After the render functions there is the mesh functions for each object. I too made each object its own function to make it easier to see each object details. I also did this because I could not create multiple shapes using one mesh and creating multiple mesh functions for each object worked. Each mesh function also has a destroy mesh function. After these functions we have the shader function that create the shaders we have and compile them while giving error messages if they were not create successfully. This is followed by a function to destroy the shader program. We then have the function to process the input from the keyboard to control the camera movement and end the program. There is then a function to resize the window in the event that the user changes the window, and this makes it so the object displayed with be correct with the change. The next two functions control the movement of the mouse and the scroll wheel to control the camera. The camera functions utilize the camera header file to be able to process the input.