The 3D scene I chose to recreate was a bunch of different candies scattered across a countertop. Using openGL I created the countertop, a lollypop, gummy worm, hard candy, and a sour cube candy. I used the plane object to create the countertop. To create the lollypop, I used a cylinder shape as the stick with a sphere on top for the candy. When I created the gummy worm, I was not sure how to turn the cylinder shape on its side, so I created multiple sphere shapes to create a similar effect. The hard candy was just a sphere shape scaled smaller than the lollypop. Lastly, I created a cube for the sour cube candy. I textured all the objects to follow along with the colors of the candy that was shown in the original photo.

A user can navigate the 3D scene using a keyboard and mouse. The A key is programmed to move the camera left, S key is backwards, D key is right, W key is forward, Q key is up, and E key is down. The mouse is used to move the camera in different directions to look around the scene. By scrolling down on the mouse wheel will slow the camera speed down, and scrolling up will speed it up. To use different input devices for this, the input devices would need to have the keys mapped and directional for the mouse. For example, this can be done using a controller by mapping the A, S, W, D, Q, E keys to different buttons on the controller. The navigation of the camera that is normally the mouse can be used as a joystick on the controller.

Some of the functions used in my code are processInput, mouse\_callback, scroll\_callback, and loadTexture. The processInput function processes the A, S, W, D, Q, and E key when they are pressed and released. This tells the camera what to do when those keys are pressed. The mouse\_callback function is used for the mouse movements. The scroll\_callback function is used for the scroll wheel movements. The loadTexture function tells the program how to display the textures on all the objects in the 3D scene. These functions make the code more modular and organized because it puts the different parts of the code in sections that are labeled so it is easy to understand and make changes. For example, if I wanted to change the A key to the F key, I would know to go to the processInput function. This makes the code more easily updatable as well, since it is organized in a way that it is easy to find where the different processes for the code are. It can also be reusable. For example, you can take the mouse\_callback function and use it for a different program that uses the mouse. This will bring functionality to other codes as well which will increase development speeds for future projects.