**Reflection**

I attempted to recreate the original 3D scene as well as I could. Some issues I ran into were creating a bowl, so for that one shape I left it as a sphere. Another issue I ran into was the ruler was actually supposed to be hallow through the middle and I was not able to figure out a way to create a hollow triangular prism. To move through the 3D scene, you use the WASD keys and the mouse. You can also press the ESC key to exit the program. Some of the steps I took to make my code more modular was using the Cylinder object by just changing the input parameters like number of sides and radius to create different shapes. Another modular piece of code is how the different textures are declared and called for each shape.

As you can see from the following snippet of code you can see how the code between the Ruler object and Plane object are similar and utilize the declared textures, positioning, and the Cylinder object:

//ruler

glActiveTexture(GL\_TEXTURE0);

glBindTexture(GL\_TEXTURE\_2D, texture3);

glBindVertexArray(VAO2);

model = glm::mat4(1.0f); // make sure to initialize matrix to identity matrix first

model = glm::translate(model, glm::vec3(-1.0f, -1.38f, 0.0f));

angle = 273.0f;

model = glm::rotate(model, glm::radians(angle), glm::vec3(1.0f, 0.3f, 0.5f));

ourShader.setMat4("model", model);

static\_meshes\_3D::Cylinder C2(0.25, 3, 1.5, true, true, true);

C2.render();

//plane

glActiveTexture(GL\_TEXTURE0);

glBindTexture(GL\_TEXTURE\_2D, texture5);

glBindVertexArray(VAO2);

model = glm::mat4(1.0f); // make sure to initialize matrix to identity matrix first

model = glm::translate(model, glm::vec3(0.0f, -2.0f, 0.0f));

ourShader.setMat4("model", model);

static\_meshes\_3D::Cylinder C3(6, 4, 0.1, true, true, true);

C3.render();

Below is the source image and the final product:



