COSC 471

Prof. Ziying Tang

Project 3

In this programming assignment, I am required to draw a solid triangle on the screen by rasterizing it. The previous assignment drew a wireframe triangle, which didn't seem very interesting to me. To rasterize the triangle, I need to create a 2-dimensional bounding box for the triangle, traverse all the pixels in this bounding box, and check whether the center point of each pixel is inside the triangle. If the pixel is inside the triangle, I compare the interpolated depth value at its location with the corresponding value in the depth buffer. If the current pixel point is closer to the camera, I set the pixel color and update the depth buffer.

To accomplish this task, I need to modify the function rasterize\_triangle() to execute the triangle rasterization algorithm. I also need to modify the function insideTriangle() to check whether a point is inside the triangle. Fortunately, I don't need to deal with the rotation transformation in this assignment, so I can simply return an identity matrix for the model transformation.

Since I only know the depth values at the three vertices of the triangle, I need to use interpolation to get the depth value for the pixels inside the triangle. The interpolated depth value is stored in the variable z\_interpolated, which is provided for me. I need to initialize the depth buffer and pay attention to the sign of z values. The z values are reversed to ensure that they are all positive numbers, and the larger the value, the further away from the viewpoint.

A picture containing graphical user interface

Description automatically generated