TDT4195, Assignment 1

Task 1:

c)
The rendering in Illustration 1 is constructed from 12 vertices forming 5 distinct white triangles on a blue background. Three of the center vertices were reused between triangles.

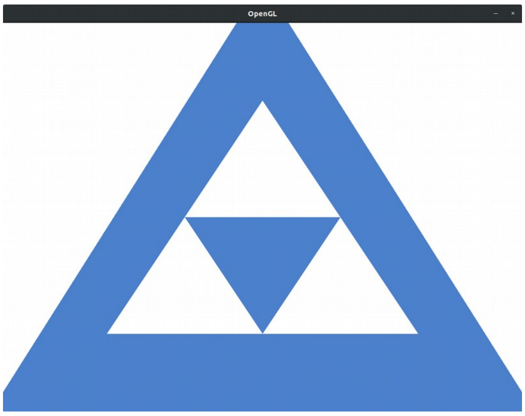


Illustration 1: 1c

Task 2:

a)
This phenomenon is known as clipping. It happens when a shape is defined outside of the screen coordinate system. Both the first and last point has their z-coordinate defined outside of the allowed range [-1,1]. This causes OpenGL to create new vertices at the intersections with the limits of the screen coordinates, which make up the “cutoff edge” for the polygon. The two short edges visible in Illustration 2 are examples of this. The purpose is to avoid rendering of shapes that should not be visible in the current viewport.

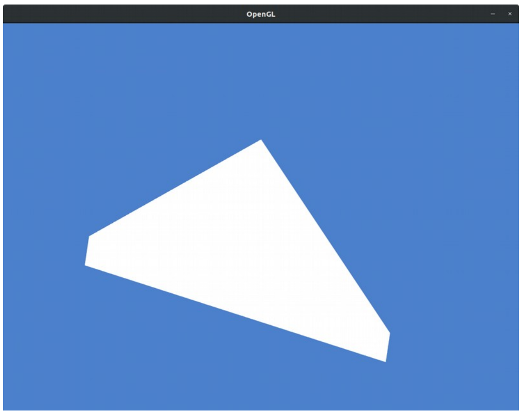


Illustration 2: 2a

b)
The triangle does not appear. This is because the back face of the triangle is facing towards the camera, and this is not rendered.
The points need to specified in a counter-clockwise fashion to make the triangle front face visible.

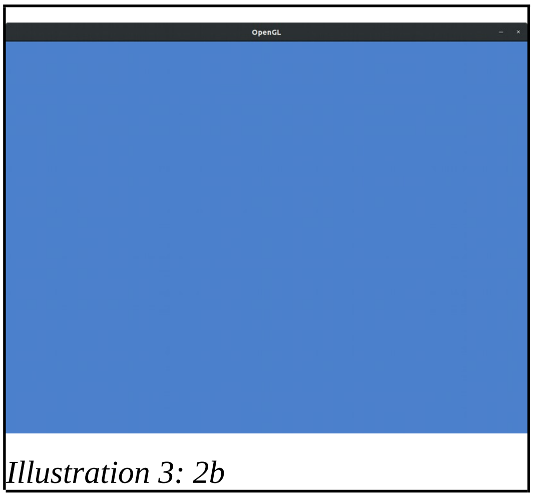


Illustration 3: 2b

c)

I) The depth buffer needs to reset each frame because each pixel is compared to the previous depth value of that pixel's location. If we do not reset the depth buffer, every pixel in a new frame would be compared to the most forward pixel with the same position in the last frame instead of just the pixels in the current frame.

II) The fragment shader may be executed multiple times for the same pixel if there are two partially overlapping shapes. The furthest shape would go through the entire pipeline first, and each pixel would get it's color value. When the closest shape then goes through the pipeline, some of the already shaded pixels would need to go through the fragment shader again because the second shape to go through the pipeline is closer and will overwrite them.

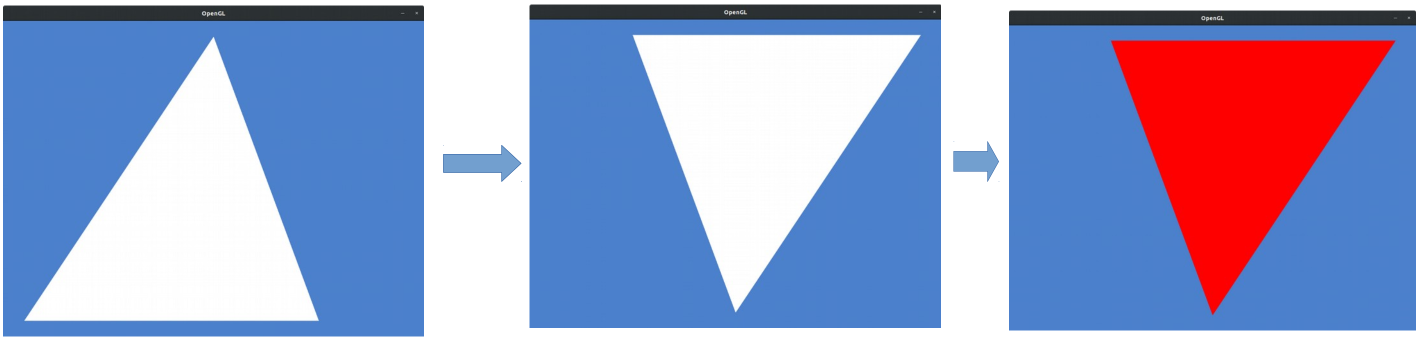
III) The two most common shader types are vertex shaders and fragment shaders.
Vertex shaders' responsibility is placing the vertices in the viewport. The fragment shaders' responsibility is providing each pixel in the output buffer with its correct color value.

IV) The main reason for using an index buffer to specify which vertices should be connected is vertex reuse. Most surfaces are connected, and the edge vertices of one triangle can be used in the neighboring ones as well. This greatly reduces the number of needed vertices, especially in large scenes with thousands or millions of triangles.

V) When there are multiple entry types in the buffer, if you are setting a pointer to an entry type which is not first in the buffer, you will need to specify the start index of your entry type.

d)
The mirroring was accomplished by simply inverting the sign of the x and y coordinates of each vertex in the vertex shader.

To change the color, I just set the values for green and blue from 1 to 0 in the fragment shader.



3)
a)