Xuan(James) Zhai-CS5382-HW1-0

1) Line 25 is used to specify the color of the triangle.

a) fColor = vec4( 0.0, 0.0, 1.0, 1.0 );

b) fColor = vec4( 1.0, 1.0, 0.0, 1.0 );

c) This line is executed on the GPU because it’s a fragment shader function.

2) Line 154 is used to specify the color of the background.

a) gl.clearColor( 0.0, 0.0, 0.0, 1.0 );

b) If I comment out line 157, the background color will become white(blank). I think that’s become it needs to clear the buffer for using the preset value specified at line 154.

3) There will be no triangle because the VertexAttribute are disabled by default.

4) The triangle will also be missing because it never draws it.

5)  The range is -1 to 1. The point (0,0) is at the center of the canvas.

6) If I change it to 128,256. The triangle will move to the top right of the canvas. When I change it to 512, there will be no triangle on the canvas. The first two values will specify the coordinates of the viewport, the default is 0,0. When I change that (x,y) values, the vertex point (0,0) will also be changed to correspond to that new coordinate. (More explanation in question 8)

7) Line 29: <canvas id="gl-canvas" width="512" height="512"> </canvas>

8) The canvas is located at the top left of the browser window because it’s part of the HTML setting or design. For the viewport, it will be set at line 151. Its size can be bigger that the canvas, but the part that goes out will not be shown because the canvas is the area, we do projection.

The viewport has a (x,y) which specifies the coordinate of the lower-left corner of the viewport rectangle, or the canvas context; its default is (0,0). The triangle vertices will correspond to the viewport, and its value are all normalized to -1.0-1.0. Based on what I found online, the viewport will set the vertices’ coordinates by doing some calculations. Therefore, the triangle vertex with value (0,0) will be at the center of the viewport.

x w = (x nd + 1) \* width /2 + x

y w = (y nd + 1) \* height /2 + y