

COMP 370 assignment #2: Rendering of Shapes with Primitives

Thomas Williamson

id: 588206

2021/10/04

Rendering of Shapes with Primitives: make a graphics program that displays a cross in a default color(I chose red), when "1" is pressed the color changes to green, when "2" is pressed the color changes to yellow, when "3" is pressed the color changes to a random color, "c" is pressed the shape changes to a cross, "r" is pressed the shape changes to circle, "l" is pressed the shape changes to the letter "A"

```

1  /*
2  COMP 370 assignment #2: Rendering of Shapes with Primitives
3  Thomas Williamson
4  id: 588206
5  2021/10/04
6
7  */
8
9  "use strict";
10 //load variables
11 var gl;
12 var cross;
13 var letter;
14 var circle;
15 var program;
16 var program2;
17 var program3;
18 var cr_vPosition;
19 var l_vPosition;
20 var ci_vPosition;
21 var crBuffer;
22 var lBuffer;
23 var ciBuffer;
24 var CR_cBuffer;
25 var l_cBuffer;
26 var ci_cBuffer;
27 var color = 1;
28 var shape = 1; //1:(input c) cross, 2:(input l) letter "l", 3: (input r) circle,
29 var colorCR = [1,0,0, 1,0,0, 1,0,0, 1,0,0, 1,0,0, 1,0,0, 1,0,0, 1,0,0,
30               1,0,0, 1,0,0, 1,0,0, 1,0,0,
31               1,0,0, 1,0,0, 1,0,0];
32 var colorL = [1,0,0, 1,0,0, 1,0,0, 1,0,0, 1,0,0,
33               1,0,0, 1,0,0, 1,0,0, 1,0,0,
34               1,0,0, 1,0,0, 1,0,0, 1,0,0];
35 var colorC = [1,0,0, 1,0,0, 1,0,0, 1,0,0, 1,0,0,
36               1,0,0, 1,0,0, 1,0,0, 1,0,0,
37               1,0,0, 1,0,0, 1,0,0, 1,0,0,
38               1,0,0, 1,0,0, 1,0,0, 1,0,0,
39               1,0,0, 1,0,0, 1,0,0, 1,0,0,
40               1,0,0, 1,0,0, 1,0,0, 1,0,0,
41               1,0,0, 1,0,0, 1,0,0, 1,0,0,
42               1,0,0, 1,0,0, 1,0,0, 1,0,0,
43               1,0,0, 1,0,0, 1,0,0, 1,0,0,
44               1,0,0, 1,0,0, 1,0,0, 1,0,0,
45               1,0,0, 1,0,0, 1,0,0, 1,0,0,
46               1,0,0, 1,0,0, 1,0,0, 1,0,0,
47               1,0,0, 1,0,0, 1,0,0, 1,0,0,
48               1,0,0, 1,0,0, 1,0,0, 1,0,0,
49               1,0,0, 1,0,0, 1,0,0, 1,0,0,
50               1,0,0, 1,0,0, 1,0,0, 1,0,0,
51               1,0,0, 1,0,0, 1,0,0, 1,0,0,
52               1,0,0, 1,0,0, 1,0,0, 1,0,0,
53               1,0,0, 1,0,0, 1,0,0, 1,0,0,
54               1,0,0, 1,0,0, 1,0,0, 1,0,0,
55               1,0,0, 1,0,0, 1,0,0, 1,0,0,
56               1,0,0, 1,0,0, 1,0,0, 1,0,0,
57               1,0,0, 1,0,0, 1,0,0, 1,0,0,

```

```

58         1,0,0, 1,0,0, 1,0,0, 1,0,0,
59         1,0,0, 1,0,0, 1,0,0, 1,0,0,
60         1,0,0, 1,0,0, 1,0,0, 1,0,0,
61         1,0,0, 1,0,0, 1,0,0, 1,0,0,
62         1,0,0, 1,0,0, 1,0,0, 1,0,0,
63         1,0,0, 1,0,0, 1,0,0, 1,0,0,
64         1,0,0, 1,0,0, 1,0,0, 1,0,0, 1,0,0]
65     var CR_ColorLoc;
66     var s_ColorLoc;
67     var ci_ColorLoc;
68
69
70     window.onload = function init()
71     {
72         //load canvas
73         var canvas = document.getElementById( "gl-canvas" );
74
75         gl = canvas.getContext('webgl2');
76         if (!gl) { alert( "WebGL 2.0 isn't available" ); }
77
78         //set shape geometries
79         cross = new Float32Array([
80             0,0,
81             -.5, -1 ,
82             .5, -1 ,
83             .5, -.5,
84             1, -.5,
85             1, .5,
86             .5, .5,
87             .5, 1,
88             -.5, 1,
89             -.5, .5,
90             -1, .5,
91             -1, -.5,
92             -.5, -.5,
93             -.5, -1
94         ]);
95
96         letter = new Float32Array([
97             -.25,1,
98             .25,1,
99             0,.5,
100            1,-1,
101            .5,-1,
102            .25,-.5,
103            .2,0,
104            -.25,-.5,
105            -.2,0,
106            -1,-1,
107            -.5,-1,
108            -.25,1,
109            0, .5
110        ]);
111
112         //python code used to obtain vertices
113         // import math
114         // print("{")

```

```

115 // for i in range(90):
116 //     if (i%3) == 0:
117 //         #print(i)
118 //         x = round((math.cos(i*math.pi/180)*1),2)
119 //         y = round((math.sin(i*math.pi/180)*1),2)
120 //         print(str(x) + ", " + str(y) + ", ")
121 // for i in range(90):
122 //     if (i%3) == 0:
123 //         x = round(-(math.cos((90-i)*math.pi/180)*1),2)
124 //         y = round((math.sin((90-i)*math.pi/180)*1),2)
125 //         print(str(x) + ", " + str(y) + ", ")
126 // for i in range(90):
127 //     if (i%3) == 0:
128 //         x = round(-(math.cos(i*math.pi/180)*1),2)
129 //         y = round(-(math.sin(i*math.pi/180)*1),2)
130 //         print(str(x) + ", " + str(y) + ", ")
131 // for i in range(90):
132
133 //     if (i%3) == 0:
134 //         x = round((math.cos((90-i)*math.pi/180)*1),2)
135 //         y = round(-(math.sin((90-i)*math.pi/180)*1),2)
136 //         print(str(x) + ", " + str(y) + ", ")
137 circle = new Float32Array([
138     0, 0, 1.0, 0.0, 1.0, 0.05,
139     0.99, 0.1, 0.99, 0.16,
140     0.98, 0.21, 0.97, 0.26,
141     0.95, 0.31, 0.93, 0.36,
142     0.91, 0.41, 0.89, 0.45,
143     0.87, 0.5, 0.84, 0.54,
144     0.81, 0.59, 0.78, 0.63,
145     0.74, 0.67, 0.71, 0.71,
146     0.67, 0.74, 0.63, 0.78,
147     0.59, 0.81, 0.54, 0.84,
148     0.5, 0.87, 0.45, 0.89,
149     0.41, 0.91, 0.36, 0.93,
150     0.31, 0.95, 0.26, 0.97,
151     0.21, 0.98, 0.16, 0.99,
152     0.1, 0.99, 0.05, 1.0,
153     -0.0, 1.0, -0.05, 1.0,
154     -0.1, 0.99, -0.16, 0.99,
155     -0.21, 0.98, -0.26, 0.97,
156     -0.31, 0.95, -0.36, 0.93,
157     -0.41, 0.91, -0.45, 0.89,
158     -0.5, 0.87, -0.54, 0.84,
159     -0.59, 0.81, -0.63, 0.78,
160     -0.67, 0.74, -0.71, 0.71,
161     -0.74, 0.67, -0.78, 0.63,
162     -0.81, 0.59, -0.84, 0.54,
163     -0.87, 0.5, -0.89, 0.45,
164     -0.91, 0.41, -0.93, 0.36,
165     -0.95, 0.31, -0.97, 0.26,
166     -0.98, 0.21, -0.99, 0.16,
167     -0.99, 0.1, -1.0, 0.05,
168     -1.0, -0.0, -1.0, -0.05,
169     -0.99, -0.1, -0.99, -0.16,
170     -0.98, -0.21, -0.97, -0.26,
171     -0.95, -0.31, -0.93, -0.36,

```

```

172         -0.91, -0.41, -0.89, -0.45,
173         -0.87, -0.5, -0.84, -0.54,
174         -0.81, -0.59, -0.78, -0.63,
175         -0.74, -0.67, -0.71, -0.71,
176         -0.67, -0.74, -0.63, -0.78,
177         -0.59, -0.81, -0.54, -0.84,
178         -0.5, -0.87, -0.45, -0.89,
179         -0.41, -0.91, -0.36, -0.93,
180         -0.31, -0.95, -0.26, -0.97,
181         -0.21, -0.98, -0.16, -0.99,
182         -0.1, -0.99, -0.05, -1.0,
183         0.0, -1.0, 0.05, -1.0,
184         0.1, -0.99, 0.16, -0.99,
185         0.21, -0.98, 0.26, -0.97,
186         0.31, -0.95, 0.36, -0.93,
187         0.41, -0.91, 0.45, -0.89,
188         0.5, -0.87, 0.54, -0.84,
189         0.59, -0.81, 0.63, -0.78,
190         0.67, -0.74, 0.71, -0.71,
191         0.74, -0.67, 0.78, -0.63,
192         0.81, -0.59, 0.84, -0.54,
193         0.87, -0.5, 0.89, -0.45,
194         0.91, -0.41, 0.93, -0.36,
195         0.95, -0.31, 0.97, -0.26,
196         0.98, -0.21, 0.99, -0.16,
197         0.99, -0.1, 1.0, -0.05, 1, 0])
198
199     gl.viewport( 0, 0, canvas.width, canvas.height );
200     gl.clearColor( 1, 1, 1, 1.0 );
201     gl.clear(gl.COLOR_BUFFER_BIT);
202
203     //initiate vertex and fragment - shader buffer datta
204     program = initShaders( gl, "vertex-shader", "fragment-shader" );
205     program2 = initShaders( gl, "vertex-shader", "fragment-shader" );
206     program3 = initShaders( gl, "vertex-shader", "fragment-shader");
207
208     //cross buffer data and render
209     crBuffer = gl.createBuffer();
210     gl.bindBuffer( gl.ARRAY_BUFFER, crBuffer );
211     gl.bufferData( gl.ARRAY_BUFFER, cross, gl.STATIC_DRAW );
212
213     cr_vPosition = gl.getAttribLocation( program, "aPosition" );
214     gl.vertexAttribPointer( cr_vPosition, 2, gl.FLOAT, false, 0, 0 );
215
216     CR_cBuffer = gl.createBuffer();
217     gl.bindBuffer(gl.ARRAY_BUFFER, CR_cBuffer);
218     gl.bufferData(gl.ARRAY_BUFFER, new Float32Array(colorCR), gl.STATIC_DRAW );
219
220     CR_ColorLoc = gl.getAttribLocation( program, "aColor");
221     gl.vertexAttribPointer(CR_ColorLoc, 3, gl.FLOAT, false, 0, 0);
222
223     gl.useProgram( program );
224     gl.enableVertexAttribArray( cr_vPosition );
225
226     gl.enableVertexAttribArray(CR_ColorLoc);
227
228     render();

```



```

229
230 //letter ...
231 lBuffer = gl.createBuffer();
232 gl.bindBuffer( gl.ARRAY_BUFFER, lBuffer );
233 gl.bufferData( gl.ARRAY_BUFFER, letter, gl.STATIC_DRAW );
234 l_vPosition = gl.getAttribLocation( program2, "aPosition" );
235 gl.vertexAttribPointer( l_vPosition, 2, gl.FLOAT, false, 0, 0 );
236
237 l_cBuffer = gl.createBuffer();
238 gl.bindBuffer(gl.ARRAY_BUFFER, l_cBuffer);
239 gl.bufferData(gl.ARRAY_BUFFER, new Float32Array(colorL), gl.STATIC_DRAW );
240
241 s_ColorLoc = gl.getAttribLocation( program2, "aColor");
242 gl.vertexAttribPointer(s_ColorLoc, 3, gl.FLOAT, false, 0, 0);
243
244 //circle ...
245 ciBuffer = gl.createBuffer();
246 gl.bindBuffer( gl.ARRAY_BUFFER, ciBuffer );
247 gl.bufferData( gl.ARRAY_BUFFER, circle, gl.STATIC_DRAW );
248 ci_vPosition = gl.getAttribLocation( program3, "aPosition" );
249 gl.vertexAttribPointer( ci_vPosition, 2, gl.FLOAT, false, 0, 0 );
250
251 ci_cBuffer = gl.createBuffer();
252 gl.bindBuffer(gl.ARRAY_BUFFER, ci_cBuffer);
253 gl.bufferData(gl.ARRAY_BUFFER, new Float32Array(colorC), gl.STATIC_DRAW );
254
255 ci_ColorLoc = gl.getAttribLocation( program3, "aColor");
256 gl.vertexAttribPointer(ci_ColorLoc, 3, gl.FLOAT, false, 0, 0);
257
258 window.addEventListener('keydown', this.checkKey);
259
260 };
261
262
263 function render() {
264
265     if(shape==1){
266         gl.clear( gl.COLOR_BUFFER_BIT );
267         gl.drawArrays( gl.TRIANGLE_FAN, 0, 14 );
268     }else if(shape==2){
269         gl.clear( gl.COLOR_BUFFER_BIT );
270         gl.drawArrays( gl.TRIANGLE_STRIP, 0, 13 );
271     }else if(shape==3){
272         gl.clear( gl.COLOR_BUFFER_BIT );
273         gl.drawArrays( gl.TRIANGLE_FAN, 0, 122 );
274     }
275 }
276
277 // keyboard input
278
279 function checkKey(e){
280     console.log(e.keyCode)
281     switch(e.keyCode){
282         // input "1" color green
283         case 49:
284             colorCR = [0,1,0, 0,1,0, 0,1,0, 0,1,0, 0,1,0, 0,1,0,
285                       0,1,0, 0,1,0, 0,1,0, 0,1,0,

```

```

286         0,1,0, 0,1,0, 0,1,0, 0,1,0];
287     colorL = [0,1,0, 0,1,0, 0,1,0, 0,1,0, 0,1,0,
288             0,1,0, 0,1,0, 0,1,0, 0,1,0,
289             0,1,0, 0,1,0, 0,1,0, 0,1,0];
290     colorC = [0,1,0, 0,1,0, 0,1,0, 0,1,0, 0,1,0,
291             0,1,0, 0,1,0, 0,1,0, 0,1,0,
292             0,1,0, 0,1,0, 0,1,0, 0,1,0,
293             0,1,0, 0,1,0, 0,1,0, 0,1,0,
294             0,1,0, 0,1,0, 0,1,0, 0,1,0,
295             0,1,0, 0,1,0, 0,1,0, 0,1,0,
296             0,1,0, 0,1,0, 0,1,0, 0,1,0,
297             0,1,0, 0,1,0, 0,1,0, 0,1,0,
298             0,1,0, 0,1,0, 0,1,0, 0,1,0,
299             0,1,0, 0,1,0, 0,1,0, 0,1,0,
300             0,1,0, 0,1,0, 0,1,0, 0,1,0,
301             0,1,0, 0,1,0, 0,1,0, 0,1,0,
302             0,1,0, 0,1,0, 0,1,0, 0,1,0,
303             0,1,0, 0,1,0, 0,1,0, 0,1,0,
304             0,1,0, 0,1,0, 0,1,0, 0,1,0,
305             0,1,0, 0,1,0, 0,1,0, 0,1,0,
306             0,1,0, 0,1,0, 0,1,0, 0,1,0,
307             0,1,0, 0,1,0, 0,1,0, 0,1,0,
308             0,1,0, 0,1,0, 0,1,0, 0,1,0,
309             0,1,0, 0,1,0, 0,1,0, 0,1,0,
310             0,1,0, 0,1,0, 0,1,0, 0,1,0,
311             0,1,0, 0,1,0, 0,1,0, 0,1,0,
312             0,1,0, 0,1,0, 0,1,0, 0,1,0,
313             0,1,0, 0,1,0, 0,1,0, 0,1,0,
314             0,1,0, 0,1,0, 0,1,0, 0,1,0,
315             0,1,0, 0,1,0, 0,1,0, 0,1,0,
316             0,1,0, 0,1,0, 0,1,0, 0,1,0,
317             0,1,0, 0,1,0, 0,1,0, 0,1,0,
318             0,1,0, 0,1,0, 0,1,0, 0,1,0,
319             0,1,0, 0,1,0, 0,1,0, 0,1,0, 0,1,0];
320     if(shape==1){
321         cross_Binding();
322     }else if(shape==2){
323         letter_Binding();
324     }else if(shape==3){
325         circle_Binding();
326     }
327     render();
328     break
329
330 // input "2" color yellow
331 case 50:
332     colorCR = [1,1,0, 1,1,0, 1,1,0, 1,1,0, 1,1,0,
333             1,1,0, 1,1,0, 1,1,0, 1,1,0,
334             1,1,0, 1,1,0, 1,1,0, 1,1,0, 1,1,0, 1,1,0];
335     colorL = [1,1,0, 1,1,0, 1,1,0, 1,1,0, 1,1,0,
336             1,1,0, 1,1,0, 1,1,0, 1,1,0,
337             1,1,0, 1,1,0, 1,1,0, 1,1,0, 1,1,0];
338     colorC = [1,1,0, 1,1,0, 1,1,0, 1,1,0, 1,1,0,
339             1,1,0, 1,1,0, 1,1,0, 1,1,0,
340             1,1,0, 1,1,0, 1,1,0, 1,1,0,
341             1,1,0, 1,1,0, 1,1,0, 1,1,0,
342             1,1,0, 1,1,0, 1,1,0, 1,1,0,

```



```

343         1,1,0, 1,1,0, 1,1,0, 1,1,0,
344         1,1,0, 1,1,0, 1,1,0, 1,1,0,
345         1,1,0, 1,1,0, 1,1,0, 1,1,0,
346         1,1,0, 1,1,0, 1,1,0, 1,1,0,
347         1,1,0, 1,1,0, 1,1,0, 1,1,0,
348         1,1,0, 1,1,0, 1,1,0, 1,1,0,
349         1,1,0, 1,1,0, 1,1,0, 1,1,0,
350         1,1,0, 1,1,0, 1,1,0, 1,1,0,
351         1,1,0, 1,1,0, 1,1,0, 1,1,0,
352         1,1,0, 1,1,0, 1,1,0, 1,1,0,
353         1,1,0, 1,1,0, 1,1,0, 1,1,0,
354         1,1,0, 1,1,0, 1,1,0, 1,1,0,
355         1,1,0, 1,1,0, 1,1,0, 1,1,0,
356         1,1,0, 1,1,0, 1,1,0, 1,1,0,
357         1,1,0, 1,1,0, 1,1,0, 1,1,0,
358         1,1,0, 1,1,0, 1,1,0, 1,1,0,
359         1,1,0, 1,1,0, 1,1,0, 1,1,0,
360         1,1,0, 1,1,0, 1,1,0, 1,1,0,
361         1,1,0, 1,1,0, 1,1,0, 1,1,0,
362         1,1,0, 1,1,0, 1,1,0, 1,1,0,
363         1,1,0, 1,1,0, 1,1,0, 1,1,0,
364         1,1,0, 1,1,0, 1,1,0, 1,1,0,
365         1,1,0, 1,1,0, 1,1,0, 1,1,0,
366         1,1,0, 1,1,0, 1,1,0, 1,1,0,
367         1,1,0, 1,1,0, 1,1,0, 1,1,0, 1,1,0];
368     if(shape==1){
369         cross_Binding();
370     }else if(shape==2){
371         letter_Binding();
372     }else if(shape==3){
373         circle_Binding();
374     }
375     render();
376     break
377
378
379     // input "3" color random
380     case 51:
381
382         colorCR = [Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.
383             Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.random(),
384             Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.random(),
385             Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.random(),
386             colorL = [Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.
387             Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.random(),
388             Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.random(),
389             Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.random(),
390             colorC = [Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.
391             Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.random(),
392             Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.random(),
393             Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.random(),
394             Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.random(),
395             Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.random(),
396             Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.random(),
397             Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.random(),
398             Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.random(),
399             Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.random(),

```

```

400 Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.random(), Math.random
401 Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.random(), Math.random
402 Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.random(), Math.random
403 Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.random(), Math.random
404 Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.random(), Math.random
405 Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.random(), Math.random
406 Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.random(), Math.random
407 Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.random(), Math.random
408 Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.random(), Math.random
409 Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.random(), Math.random
410 Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.random(), Math.random
411 Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.random(), Math.random
412 Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.random(), Math.random
413 Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.random(), Math.random
414 Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.random(), Math.random
415 Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.random(), Math.random
416 Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.random(), Math.random
417 Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.random(), Math.random
418 Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.random(), Math.random
419 Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.random(), Math.random
420 Math.random(),Math.random(),Math.random(), Math.random(),Math.random(),Math.random(), Math.random
421     if(shape==1){
422         cross_Binding();
423     }else if(shape==2){
424         letter_Binding();
425     }else if(shape==3){
426         circle_Binding();
427     }
428     render();
429     break
430
431
432     // input "l" letter
433     case 76:
434         shape = 2;
435         letter_Binding();
436         render();
437         break;
438
439     // input "c" cross
440     case 67:
441         shape = 1;
442         cross_Binding();
443         render();
444         break;
445
446     // input "r" circle
447     case 82:
448         shape = 3;
449         circle_Binding();
450         render();
451         break;
452 }
453 }
454
455 //shape data
456 function cross_Binding(){

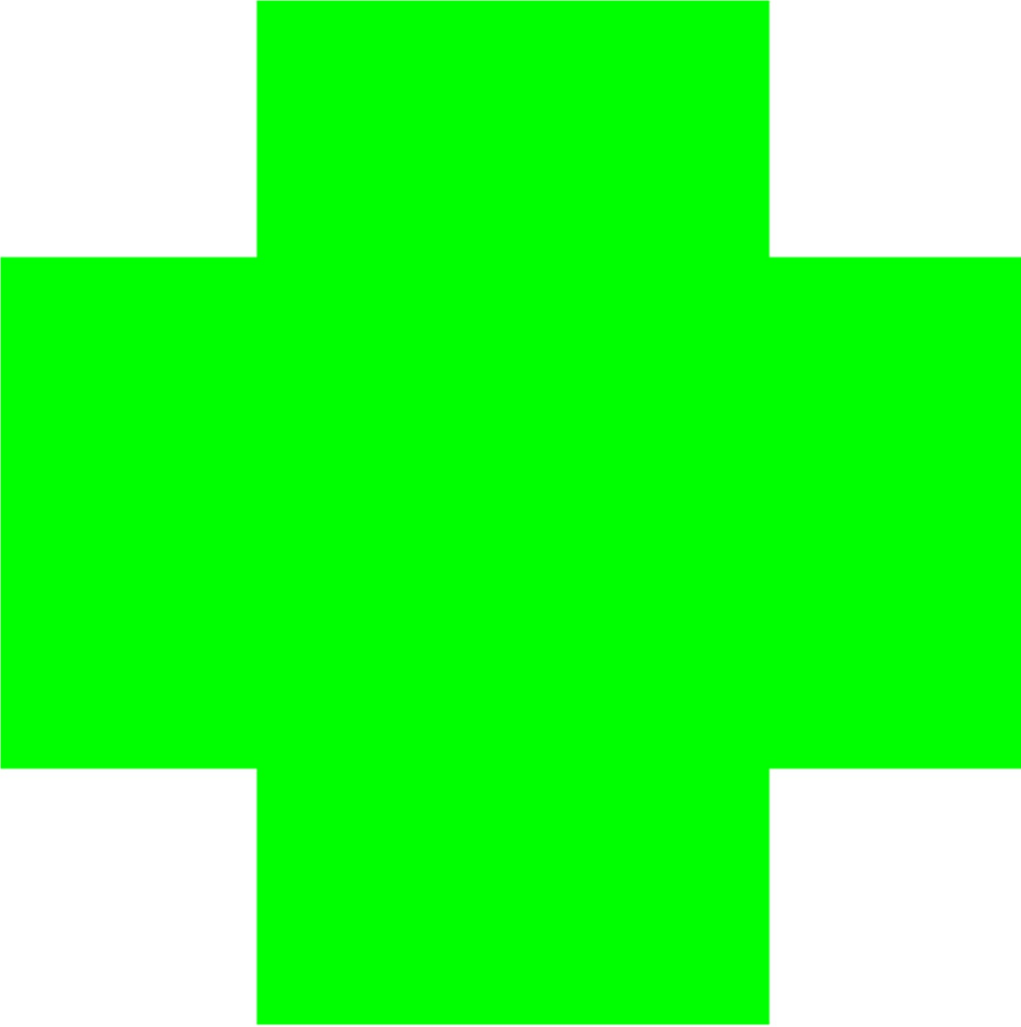
```

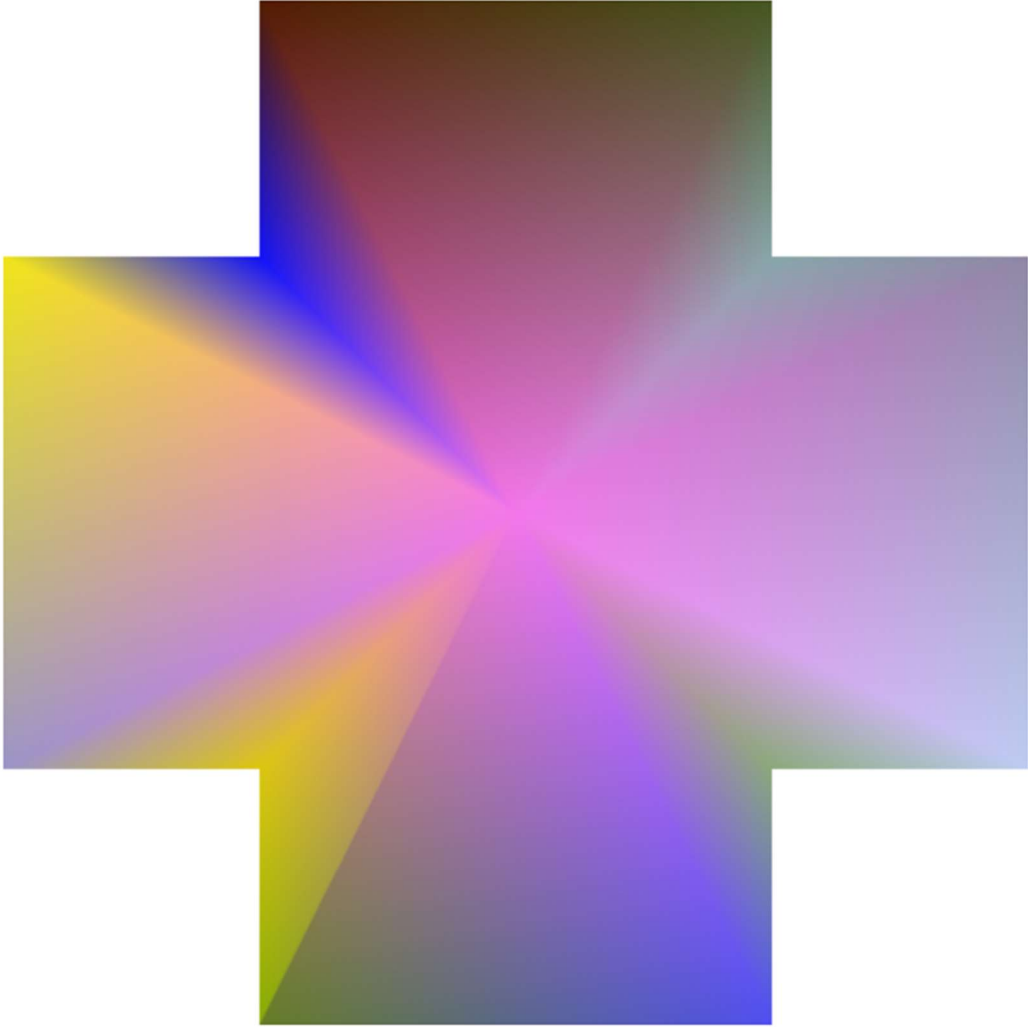
```

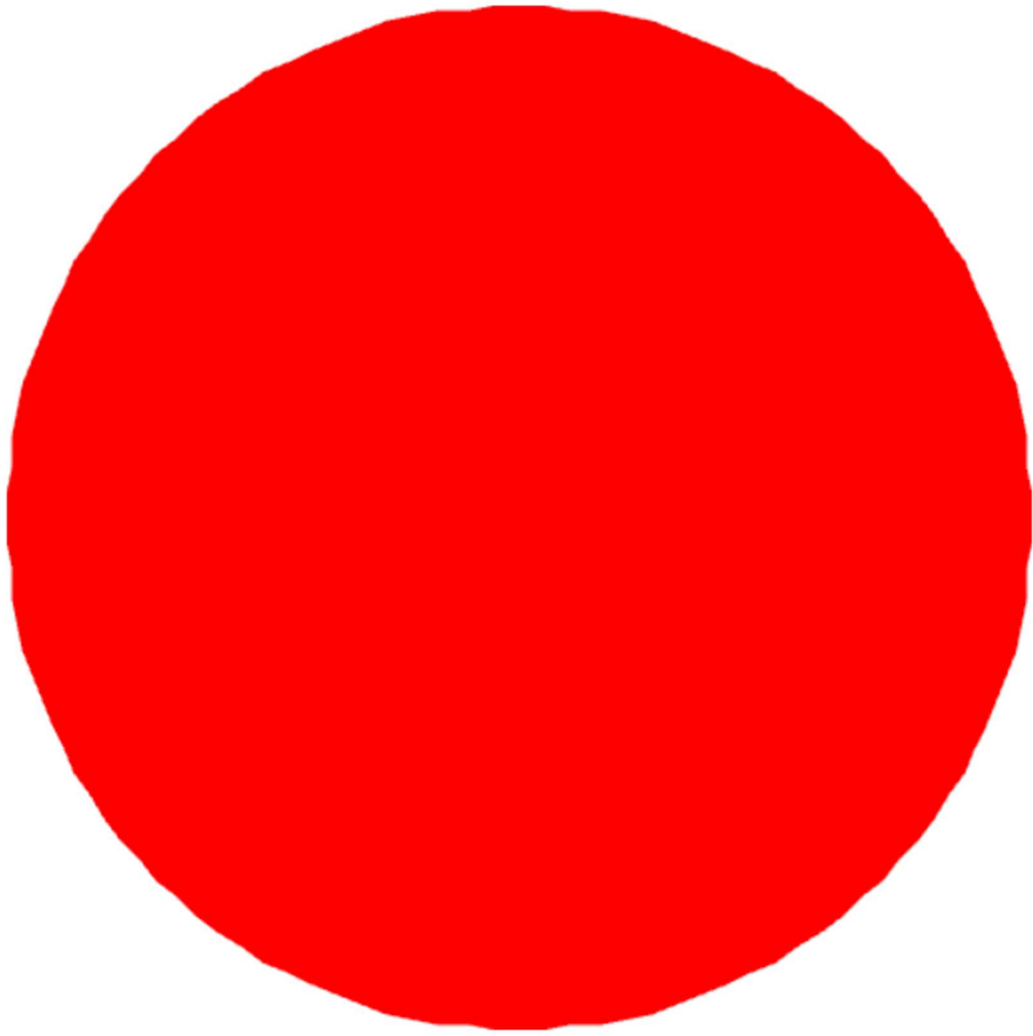
457     gl.useProgram( program );
458     gl.enableVertexAttribArray( cr_vPosition );
459     gl.bindBuffer( gl.ARRAY_BUFFER, crBuffer );
460     gl.vertexAttribPointer( cr_vPosition, 2, gl.FLOAT, false, 0, 0 );
461     gl.bindBuffer(gl.ARRAY_BUFFER, CR_cBuffer);
462     gl.bufferData(gl.ARRAY_BUFFER, new Float32Array(colorCR), gl.STATIC_DRAW );
463     CR_ColorLoc = gl.getAttribLocation( program, "aColor");
464     gl.vertexAttribPointer(CR_ColorLoc, 3, gl.FLOAT, false, 0, 0);
465 }
466
467 function letter_Binding(){
468     gl.useProgram( program2 );
469     gl.enableVertexAttribArray( l_vPosition );
470     gl.bindBuffer( gl.ARRAY_BUFFER, lBuffer );
471     gl.vertexAttribPointer( l_vPosition, 2, gl.FLOAT, false, 0, 0 );
472     gl.bindBuffer(gl.ARRAY_BUFFER, l_cBuffer);
473     gl.bufferData(gl.ARRAY_BUFFER, new Float32Array(colorL), gl.STATIC_DRAW );
474     s_ColorLoc = gl.getAttribLocation( program2, "aColor");
475     gl.vertexAttribPointer(s_ColorLoc, 3, gl.FLOAT, false, 0, 0);
476 }
477
478 function circle_Binding(){
479     gl.useProgram( program3 );
480     gl.enableVertexAttribArray( ci_vPosition );
481     gl.bindBuffer( gl.ARRAY_BUFFER, ciBuffer );
482     gl.vertexAttribPointer( ci_vPosition, 2, gl.FLOAT, false, 0, 0 );
483     gl.bindBuffer(gl.ARRAY_BUFFER, ci_cBuffer);
484     gl.bufferData(gl.ARRAY_BUFFER, new Float32Array(colorC), gl.STATIC_DRAW );
485     ci_ColorLoc = gl.getAttribLocation( program3, "aColor");
486     gl.vertexAttribPointer(ci_ColorLoc, 3, gl.FLOAT, false, 0, 0);
487 }

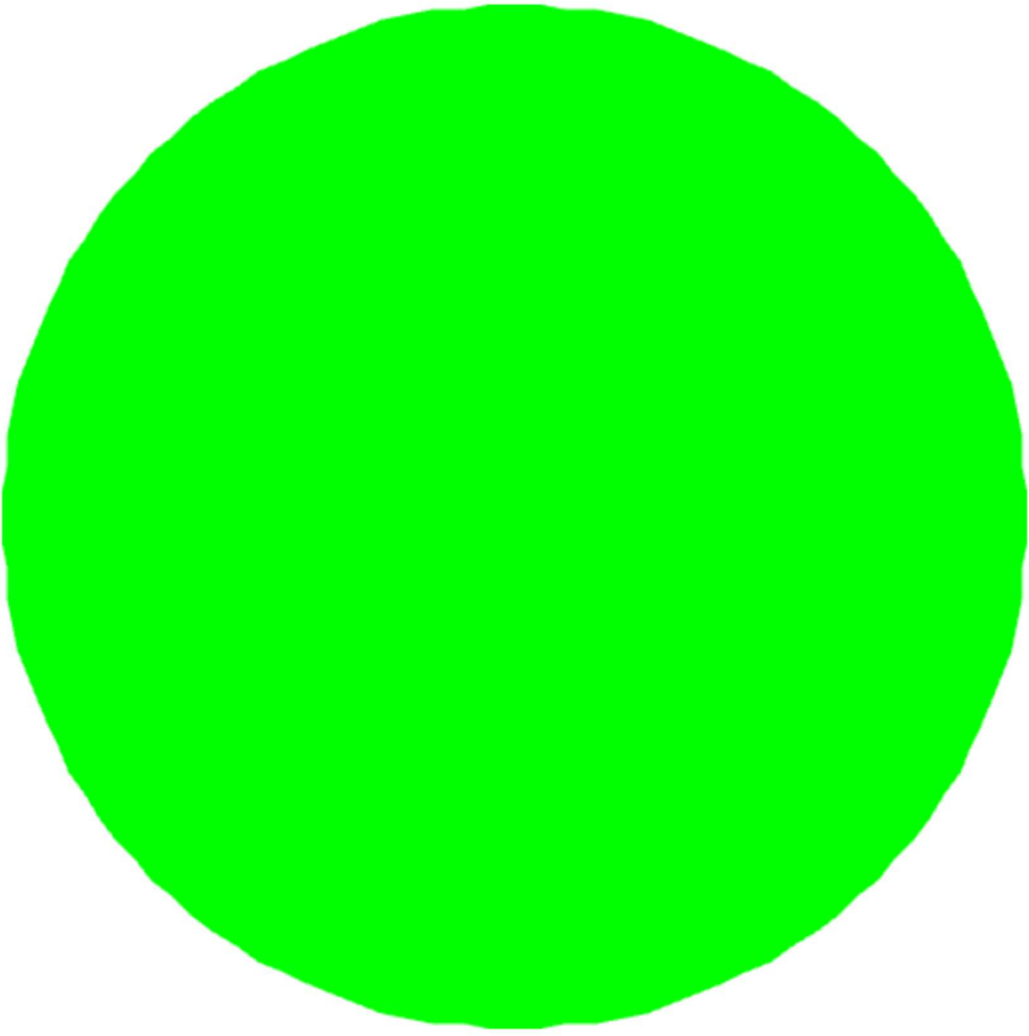
```













A

A

