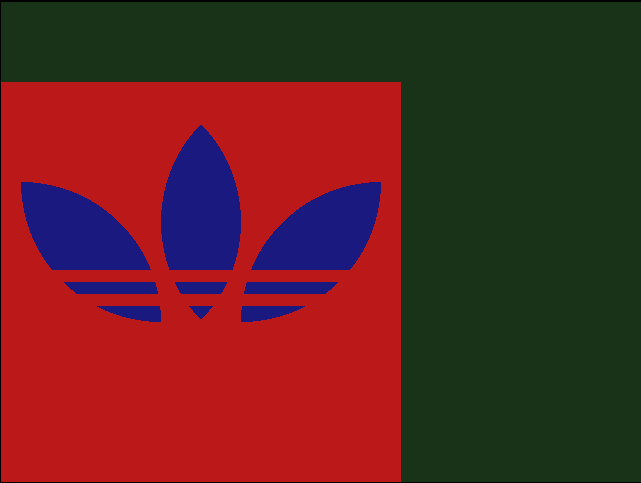
**CSE 5542 - Realtime Rendering**

**Homework #1 Implicit functions to create backgrounds**

**Date: Tuesday, August 29, 2017**

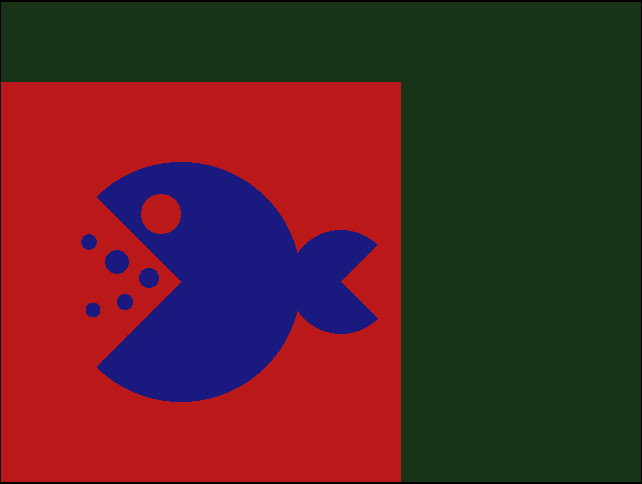
**Name: Ming Yi, Su**

**1. Adidas logo**



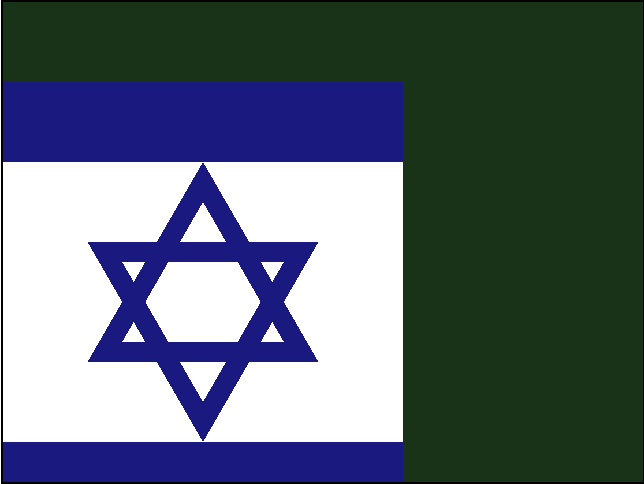
**I am trying to draw the irregular plot by circle and limitations.**

**2. Fish**



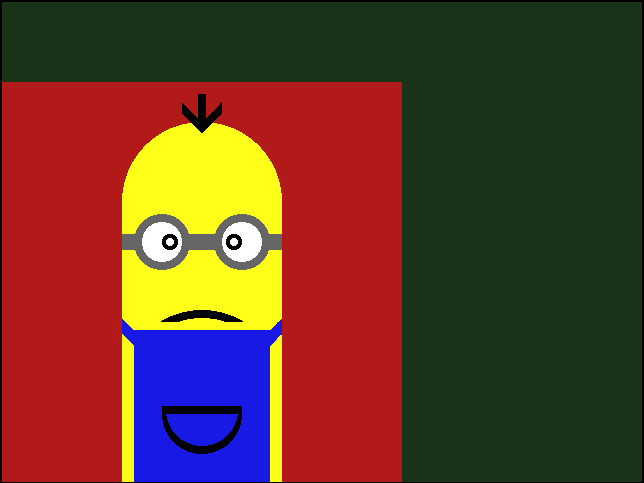
**With circle and graphic tools, I try to draw an simple animal.**

**3.** **Hexagram**



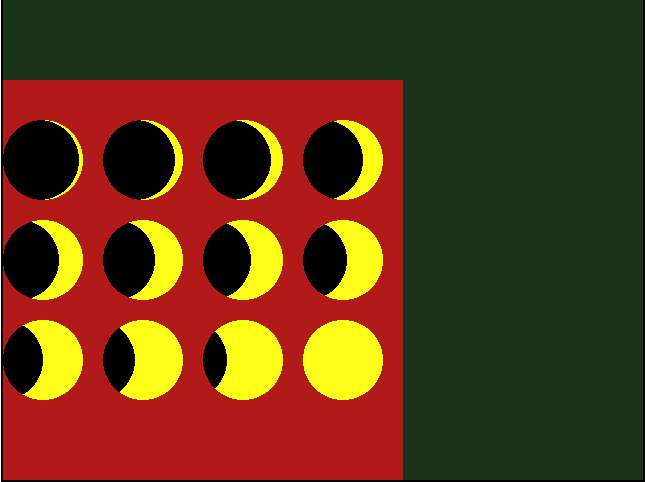
**I use limitations of lines to draw the hexagram.**

**4. Minions**



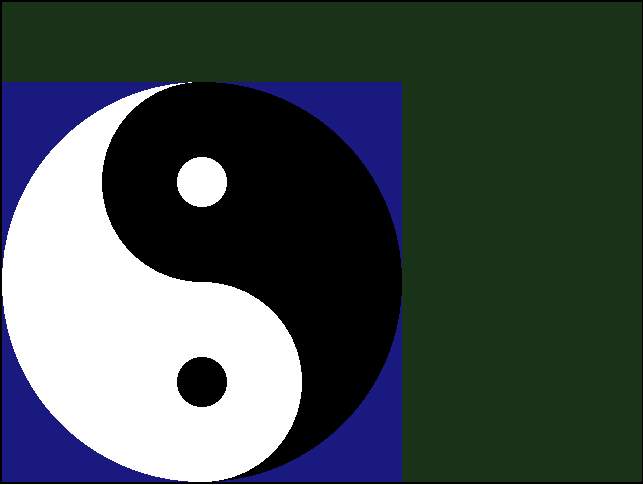
**It is my favorite one. This plot combines everything including lines, circles, and limitations.**

**5. Moon circle**



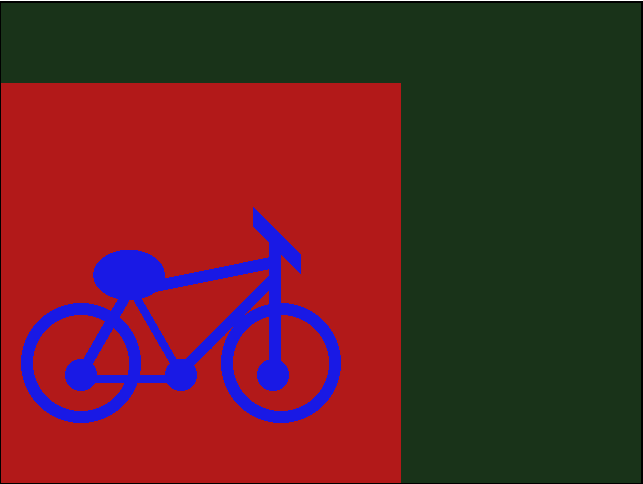
**I try to repeat something and change it a little bit when in next circle.**

**6.** **Yin-Yang**



**This is the example provided by professor. It lets me understand how java works with if and else.**

**7. bicycle**



**With lots of lines and circles, I try to present a bicycle in this plot.**

**Conclusion:**

**I think that GPU, browser, and webGL were worked in this homework. I have countered an issue that weird error. We must provide “1.0” rather than “1” to webGL. Otherwise, it results in error all the time. And, I understand the statement that a Fragment Shader is the Shader stage that will process a Fragment generated by the Rasterization into a set of colors and a single depth value.**

|  |  |
| --- | --- |
| **Picture number** | **Fragment Shader** |
| **1** | **<script id="SMY\_hw1\_fs" type="x-shader/x-fragment">**  **precision highp float;**  **void main(void) {**  **float yThreshold;**  **float xThreshold;**  **const float xScale = 1.0 / 400.0;**  **const float yScale = 1.0 / 400.0;**  **const vec4 scarlet = vec4(0.733, 0.1, 0.1, 1.0);**  **const vec4 grey = vec4(0.4, 0.4, 0.4, 1.0);**  **const vec4 blue = vec4(0.1, 0.1, 0.5, 1.0);**  **float x = xScale \* gl\_FragCoord.x;**  **float y = yScale \* gl\_FragCoord.y;**  **vec4 color;**  **yThreshold = 30.0\*(sin(40.0 \* 3.1415\*x) + 1.0);**  **xThreshold = 30.0\*(cos(40.0 \* 3.1415\*y) + 1.0);**        **if(((x-0.25)\*(x-0.25)+(y-0.65)\*(y-0.65)<0.1225&&(x-0.75)\*(x-0.75)+(y-0.65)\*(y-0.65)<0.1225**  **|| (x-0.6)\*(x-0.6)+(y-0.75)\*(y-0.75)<0.1225&&(x-0.95)\*(x-0.95)+(y-0.4)\*(y-0.4)<0.1225**  **|| (x-0.4)\*(x-0.4)+(y-0.75)\*(y-0.75)<0.1225&&(x-0.05)\*(x-0.05)+(y-0.4)\*(y-0.4)<0.1225)**  **&&((y>0.53)||(y<0.5&&y>0.47)||(y<0.44)))**  **color = blue;**      **else**  **color = scarlet;**  **gl\_FragColor = color;**  **}**  **</script>** |
| **2** | **<script id="SMY\_hw1\_fs" type="x-shader/x-fragment">**  **precision highp float;**  **void main(void) {**  **float yThreshold;**  **float xThreshold;**  **const float xScale = 1.0 / 400.0;**  **const float yScale = 1.0 / 400.0;**  **const vec4 scarlet = vec4(0.733, 0.1, 0.1, 1.0);**  **const vec4 grey = vec4(0.4, 0.4, 0.4, 1.0);**  **const vec4 blue = vec4(0.1, 0.1, 0.5, 1.0);**  **float x = xScale \* gl\_FragCoord.x;**  **float y = yScale \* gl\_FragCoord.y;**  **vec4 color;**  **yThreshold = 30.0\*(sin(40.0 \* 3.1415\*x) + 1.0);**  **xThreshold = 30.0\*(cos(40.0 \* 3.1415\*y) + 1.0);**        **if((((x-0.45)\*(x-0.45)+(y-0.5)\*(y-0.5)<0.09)**  **&&((0.95-x<y)||(x+0.05>y))**  **&&((x-0.4)\*(x-0.4)+(y-0.67)\*(y-0.67)>0.0025)**  **)**  **||(((x-0.85)\*(x-0.85)+(y-0.5)\*(y-0.5)<0.0169)**  **&&((x-0.35<y)||(y<1.35-x)))**  **||((x-0.29)\*(x-0.29)+(y-0.55)\*(y-0.55)<0.0009)**  **||((x-0.31)\*(x-0.31)+(y-0.45)\*(y-0.45)<0.00042)**  **||((x-0.22)\*(x-0.22)+(y-0.60)\*(y-0.60)<0.00038)**  **||((x-0.37)\*(x-0.37)+(y-0.51)\*(y-0.51)<0.000625)**  **||((x-0.23)\*(x-0.23)+(y-0.43)\*(y-0.43)<0.00035)**  **)**  **color = blue;**      **else**  **color = scarlet;**  **gl\_FragColor = color;**  **}**  **</script>** |
| **3** | **<script id="SMY\_hw1\_fs" type="x-shader/x-fragment">**  **precision highp float;**  **void main(void) {**  **float yThreshold;**  **float xThreshold;**  **const float xScale = 1.0 / 400.0;**  **const float yScale = 1.0 / 400.0;**  **const vec4 scarlet = vec4(1.0, 1.0, 1.0, 1.0);**  **const vec4 grey = vec4(0.4, 0.4, 0.4, 1.0);**  **const vec4 blue = vec4(0.1, 0.1, 0.5, 1.0);**  **float x = xScale \* gl\_FragCoord.x;**  **float y = yScale \* gl\_FragCoord.y;**  **vec4 color;**  **yThreshold = 30.0\*(sin(40.0 \* 3.1415\*x) + 1.0);**  **xThreshold = 30.0\*(cos(40.0 \* 3.1415\*y) + 1.0);**    **if((y<1.732\*x-0.066 && y>1.732\*x-0.166&&x<0.5&&y>0.3)||**  **(y<0.35&&y>0.3&&y<1.732\*x-0.066&&y<1.666-1.732\*x)||**  **(y<1.666-1.732\*x&&x>0.5&&y>0.3&&y>1.566-1.732\*x)||**  **(y<0.6&&y>0.55&&y>0.966-1.732\*x&&y>1.732\*x-0.766)||**  **(y>0.966-1.732\*x&&y<1.066-1.732\*x&&x<0.5&&y<0.6)||**  **(y>1.732\*x-0.766&&y<1.732\*x-0.666&&x>0.5&&y<0.6)||**  **(y>0.8)||(y<0.1))**  **color = blue;**      **else**  **color = scarlet;**  **gl\_FragColor = color;**  **}**  **</script>** |
| **4** | **<script id="SMY\_hw1\_fs" type="x-shader/x-fragment">**  **precision highp float;**  **void main(void) {**  **float yThreshold;**  **float xThreshold;**  **const float xScale = 1.0 / 400.0;**  **const float yScale = 1.0 / 400.0;**  **const vec4 scarlet = vec4(0.7, 0.1, 0.1, 1.0);**  **const vec4 grey = vec4(0.4, 0.4, 0.4, 1.0);**  **const vec4 yellow = vec4(1.0, 1.0, 0.1, 1.0);**  **const vec4 white= vec4(1.0, 1.0, 1.0, 1.0);**  **const vec4 black= vec4(0.0, 0.0, 0.0, 0.0);**  **const vec4 blue = vec4(0.1, 0.1, 0.9, 1.0);**  **float x = xScale \* gl\_FragCoord.x;**  **float y = yScale \* gl\_FragCoord.y;**  **vec4 color;**  **yThreshold = 30.0\*(sin(40.0 \* 3.1415\*x) + 1.0);**  **xThreshold = 30.0\*(cos(40.0 \* 3.1415\*y) + 1.0);**  **if ((x-0.5)\*(x-0.5)+(y-0.7)\*(y-0.7)<0.04||**  **(x>0.3&&x<0.7&&y<0.7)){**  **color = yellow;**  **if(((x-0.4)\*(x-0.4)+(y-0.6)\*(y-0.6)<0.0049&&**  **(x-0.4)\*(x-0.4)+(y-0.6)\*(y-0.6)>0.0025)||**  **(x>0.3&&x<0.34&&0.58<y&&y<0.62)||**  **(x>0.45&&x<0.55&&0.58<y&&y<0.62)||**  **(x>0.65&&x<0.70&&0.58<y&&y<0.62))**  **color = grey;**  **if((x-0.6)\*(x-0.6)+(y-0.6)\*(y-0.6)<0.0049&&**  **(x-0.6)\*(x-0.6)+(y-0.6)\*(y-0.6)>0.0025)**  **color = grey;**  **if((x-0.4)\*(x-0.4)+(y-0.6)\*(y-0.6)<0.0025)**  **color = white;**  **if((x-0.6)\*(x-0.6)+(y-0.6)\*(y-0.6)<0.0025)**  **color = white;**  **if(((x>0.33&&x<0.67&&y<0.38)||((y<0.71-x)&&(y>0.67-x)&&x<0.5))||**  **((y<x-0.29)&&x>0.5&&y>x-0.33))**  **color =blue;**  **if(((x-0.42)\*(x-0.42)+(y-0.6)\*(y-0.6)<0.0004&&**  **(x-0.42)\*(x-0.42)+(y-0.6)\*(y-0.6)>0.0001)||**  **((x-0.58)\*(x-0.58)+(y-0.6)\*(y-0.6)<0.0004&&**  **(x-0.58)\*(x-0.58)+(y-0.6)\*(y-0.6)>0.0001)||**  **(((x-0.5)\*(x-0.5)+(y-0.23)\*(y-0.23)<0.04)&&**  **((x-0.5)\*(x-0.5)+(y-0.23)\*(y-0.23)>0.0324)&&y>0.40)||**  **(((x-0.5)\*(x-0.5)+(y-0.17)\*(y-0.17)<0.01)&&**  **(x-0.5)\*(x-0.5)+(y-0.18)\*(y-0.18)>0.0081&&y<0.17)||**  **(x<0.6&&x>0.4&&y>0.17&&y<0.19)||**  **(x<0.51&&x>0.49&&y>0.88&&y<0.9)||**  **(x>0.45&&x<0.5&&y>1.37-x&&y<1.4-x)||**  **(x>0.5&&x<0.55&&y>x+0.37&&y<x+0.4))**  **color = black;**  **}**  **else**  **if((x<0.51&&x>0.49&&y>0.9&&y<0.97)||**  **(x>0.45&&x<0.5&&y>1.37-x&&y<1.4-x)||**  **(x>0.5&&x<0.55&&y>x+0.37&&y<x+0.4))**  **color= black;**  **else**  **color = scarlet;**    **gl\_FragColor = color;**  **}**  **</script>** |
| **5** | **<script id="SMY\_hw1\_fs" type="x-shader/x-fragment">**  **precision highp float;**  **void main(void) {**  **float yThreshold;**  **float xThreshold;**  **const float xScale = 1.0 / 400.0;**  **const float yScale = 1.0 / 400.0;**  **const vec4 scarlet = vec4(0.7, 0.1, 0.1, 1.0);**  **const vec4 grey = vec4(0.4, 0.4, 0.4, 1.0);**  **const vec4 yellow = vec4(1.0, 1.0, 0.1, 1.0);**  **const vec4 white= vec4(1.0, 1.0, 1.0, 1.0);**  **const vec4 black= vec4(0.0, 0.0, 0.0, 0.0);**  **const vec4 blue = vec4(0.1, 0.1, 0.9, 1.0);**  **float x = xScale \* gl\_FragCoord.x;**  **float y = yScale \* gl\_FragCoord.y;**  **vec4 color;**  **yThreshold = 30.0\*(sin(40.0 \* 3.1415\*x) + 1.0);**  **xThreshold = 30.0\*(cos(40.0 \* 3.1415\*y) + 1.0);**  **if((x-0.1)\*(x-0.1)+(y-0.8)\*(y-0.8)<0.01||**  **((x-0.35)\*(x-0.35)+(y-0.8)\*(y-0.8)<0.01)||**  **((x-0.6)\*(x-0.60)+(y-0.8)\*(y-0.8)<0.01)||**  **((x-0.85)\*(x-0.85)+(y-0.8)\*(y-0.8)<0.01)||**  **/\* second line moon\*/**  **((x-0.1)\*(x-0.1)+(y-0.55)\*(y-0.55)<0.01)||**  **((x-0.35)\*(x-0.35)+(y-0.55)\*(y-0.55)<0.01)||**  **((x-0.6)\*(x-0.6)+(y-0.55)\*(y-0.55)<0.01)||**  **((x-0.85)\*(x-0.85)+(y-0.55)\*(y-0.55)<0.01)||**  **/\* third line moon\*/**  **((x-0.1)\*(x-0.1)+(y-0.3)\*(y-0.3)<0.01)||**  **((x-0.35)\*(x-0.35)+(y-0.3)\*(y-0.3)<0.01)||**  **((x-0.6)\*(x-0.6)+(y-0.3)\*(y-0.3)<0.01)||**  **((x-0.85)\*(x-0.85)+(y-0.3)\*(y-0.3)<0.01)**  **){**  **color = yellow;**  **if((x-0.09)\*(x-0.09)+(y-0.8)\*(y-0.8)<0.01||**  **((x-0.33)\*(x-0.33)+(y-0.8)\*(y-0.8)<0.01)||**  **((x-0.57)\*(x-0.57)+(y-0.8)\*(y-0.8)<0.01)||**  **((x-0.80)\*(x-0.80)+(y-0.8)\*(y-0.8)<0.01)||**  **((x-0.04)\*(x-0.04)+(y-0.55)\*(y-0.55)<0.01)||**  **((x-0.28)\*(x-0.28)+(y-0.55)\*(y-0.55)<0.01&&x>0.25)||**  **((x-0.52)\*(x-0.52)+(y-0.55)\*(y-0.55)<0.01&&x>0.5)||**  **((x-0.76)\*(x-0.76)+(y-0.55)\*(y-0.55)<0.01&&x>0.75)||**  **(x)\*(x)+(y-0.3)\*(y-0.3)<0.01||**  **((x-0.23)\*(x-0.23)+(y-0.3)\*(y-0.3)<0.01&&x>0.25)||**  **((x-0.46)\*(x-0.46)+(y-0.3)\*(y-0.3)<0.01&&x>0.5)**  **)**  **color = black;**    **}**  **else**  **color = scarlet;**    **gl\_FragColor = color;**  **}**  **</script>** |
| **6** | **<script id="SMY\_hw1\_fs" type="x-shader/x-fragment">**  **precision highp float;**  **void main(void) {**  **float yThreshold;**  **float xThreshold;**  **const float xScale = 1.0 / 400.0;**  **const float yScale = 1.0 / 400.0;**  **const vec4 scarlet = vec4(1.0, 1.0, 1.0, 1.0);**  **const vec4 grey = vec4(0.0, 0.0, 0.0, 1.0);**  **const vec4 blue = vec4(0.1, 0.1, 0.5, 1.0);**  **float x = xScale \* gl\_FragCoord.x;**  **float y = yScale \* gl\_FragCoord.y;**  **vec4 color;**  **yThreshold = 30.0\*(sin(40.0 \* 3.1415\*x) + 1.0);**  **xThreshold = 30.0\*(cos(40.0 \* 3.1415\*y) + 1.0);**    **/\* first step\*/**  **if(x >0.5)**  **if((x-0.5)\*(x-0.5)+(y-0.5)\*(y-0.5)<0.25)**  **color = grey;**  **else**  **color = blue;**    **if(x <0.5)**  **if((x-0.5)\*(x-0.5)+(y-0.5)\*(y-0.5)<0.25)**  **color = scarlet;**  **else**  **color = blue;**  **/\* Second step\*/**  **if(x <0.5)**  **if((x-0.5)\*(x-0.5)+(y-0.75)\*(y-0.75)<0.0625)**  **color = grey;**  **if(x >0.5)**  **if((x-0.5)\*(x-0.5)+(y-0.25)\*(y-0.25)<0.0625)**  **color = scarlet;**  **/\* Third step\*/**  **if ((x-0.5)\*(x-0.5)+(y-0.75)\*(y-0.75)<0.00390625)**  **color = scarlet;**  **if ((x-0.5)\*(x-0.5)+(y-0.25)\*(y-0.25)<0.00390625)**  **color = grey;**            **gl\_FragColor = color;**  **}**  **</script>** |
| **7** | **<script id="SMY\_hw1\_fs" type="x-shader/x-fragment">**  **precision highp float;**  **void main(void) {**  **float yThreshold;**  **float xThreshold;**  **const float xScale = 1.0 / 400.0;**  **const float yScale = 1.0 / 400.0;**  **const vec4 scarlet = vec4(0.7, 0.1, 0.1, 1.0);**  **const vec4 grey = vec4(0.4, 0.4, 0.4, 1.0);**  **const vec4 yellow = vec4(1.0, 1.0, 0.1, 1.0);**  **const vec4 white= vec4(1.0, 1.0, 1.0, 1.0);**  **const vec4 black= vec4(0.0, 0.0, 0.0, 0.0);**  **const vec4 blue = vec4(0.1, 0.1, 0.9, 1.0);**  **float x = xScale \* gl\_FragCoord.x;**  **float y = yScale \* gl\_FragCoord.y;**  **vec4 color;**  **yThreshold = 30.0\*(sin(40.0 \* 3.1415\*x) + 1.0);**  **xThreshold = 30.0\*(cos(40.0 \* 3.1415\*y) + 1.0);**      **if (((x-0.2)\*(x-0.2)+(y-0.3)\*(y-0.3)<0.0225&&**  **(x-0.2)\*(x-0.2)+(y-0.3)\*(y-0.3)>0.0144)||**  **((x-0.7)\*(x-0.7)+(y-0.3)\*(y-0.3)<0.0225&&**  **(x-0.7)\*(x-0.7)+(y-0.3)\*(y-0.3)>0.0144)||**  **(y<1.732\*x-0.0464&&y>1.732\*x-0.1&&x<0.32&&x>0.2)||**  **(x<0.45&&x>0.2&&y>0.25&&y<0.27)||**  **(y>1.03-1.732\*x&&y<1.07-1.732\*x&&x<0.45&&x>0.30)||**  **(0.5\*(x-0.32)\*(x-0.32)+(y-0.52)\*(y-0.52)<0.004)||**  **(y<x-0.15&&y>x-0.19&&x>0.45&&x<0.7)||**  **(x>0.67&&x<0.7&&y<0.6&&y>0.25)||**  **(y>0.2\*x+0.4&&y<0.2\*x+0.43&&x>0.32&&x<0.7)||**  **(y>1.27-x&&y<1.32-x&&x>0.63&&x<0.75)||**  **((x-0.2)\*(x-0.2)+(y-0.27)\*(y-0.27)<0.0016)||**  **((x-0.45)\*(x-0.45)+(y-0.27)\*(y-0.27)<0.0016)||**  **((x-0.68)\*(x-0.68)+(y-0.27)\*(y-0.27)<0.0016)**  **)**  **color = blue;**    **else**  **color = scarlet;**    **gl\_FragColor = color;**  **}**  **</script>** |