**NOMBRE:** QUINO JIMENEZ VICTOR DANIEL **CI:** 6944974 LP

Q

**MATERIA:** INF-323 “PROGRAMACION GRAFICA” **LIC.** FELIPEZ

**UMSA – INFORMATICA**

3. Problema asignado para entrega. Escriba un programa en OpenGL ES, que genere un interesante escenario en dos dimensiones utilizando cuadrados, triangulos y cırculos en multiples colores. Denomine a este proyecto OpenGLES-Escena. Puede dibujar un cırculo con una secuencia de segmentos de lınea.

La solución debe incluir en un documento pdf, con la siguiente información:

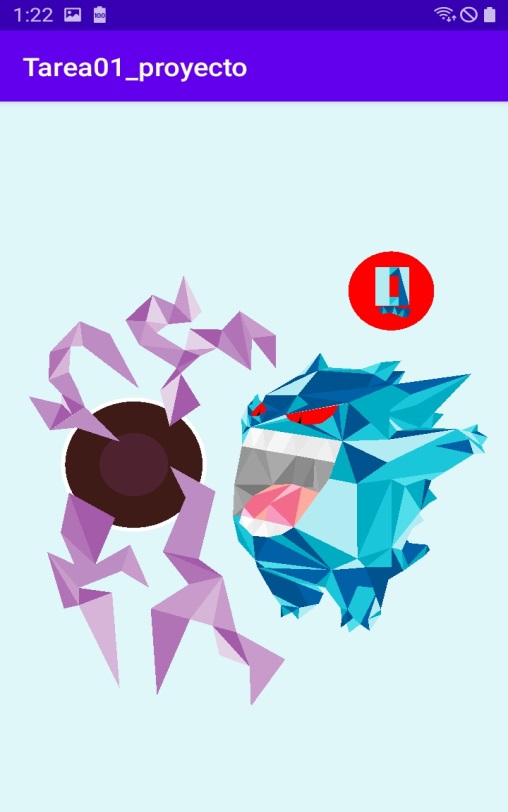
1. Código fuente.

**public class** Renderiza **implements** Renderer {  
 */\* Objeto \*/* **private** List<Triangulo> **triangulo\_array** = **new** ArrayList();  
 **private** List<Triangulo> **triangulo\_array\_circulo\_blanco** = **new** ArrayList();  
 **private** List<Triangulo> **triangulo\_array\_circulo\_cafe1** = **new** ArrayList();  
 **private** List<Triangulo> **triangulo\_array\_circulo\_cafe2** = **new** ArrayList();  
 **private** List<Triangulo> **triangulo\_array\_dientes** = **new** ArrayList(); *// para los dientes* **private** List<Triangulo> **triangulo\_array\_lengua** = **new** ArrayList(); *// para la lengua* **private** List<Triangulo> **triangulo\_array\_boca** = **new** ArrayList(); *// para la boca* **private** List<Triangulo> **triangulo\_array\_ojos** = **new** ArrayList(); *// para la ojos* **private** List<Triangulo> **triangulo\_array\_iris** = **new** ArrayList(); *// para la iris* **private** List<Triangulo> **triangulo\_array\_mano\_d** = **new** ArrayList(); *// para la mano d* **private** List<Triangulo> **triangulo\_array\_mano\_i** = **new** ArrayList(); *// para la mano i* **private** List<Triangulo> **triangulo\_array\_pies** = **new** ArrayList(); *// para los pies* **private** List<Triangulo> **triangulo\_array\_cuerpo** = **new** ArrayList(); *// para el cuerpo* **private** List<Triangulo> **triangulo\_array\_quino** = **new** ArrayList(); *// para el cuerpo* @Override  
 **public void** onSurfaceCreated(GL10 gl, EGLConfig arg1) {  
 *// COLORES:* Colores colores = **new** Colores();  
  
 *// BOMBA POKEMON* **int** nro\_triangulos=360;  
 CirculoVertices circuloVertices = **new** CirculoVertices(-140,-10,85);  
 **for**(**int** i = 0; i < nro\_triangulos; i++){  
 **triangulo\_array\_circulo\_blanco**.add(**new** Triangulo(  
 circuloVertices.getVerticesCirculo(i),  
 colores.getRgba(2)));  
 }  
 circuloVertices = **new** CirculoVertices(-140,-10,80);  
 **for**(**int** i = 0; i < nro\_triangulos; i++){  
 **triangulo\_array\_circulo\_cafe1**.add(**new** Triangulo(  
 circuloVertices.getVerticesCirculo(i),  
 colores.getRgba(3)));  
 }  
 circuloVertices = **new** CirculoVertices(-140,-10,40);  
 **for**(**int** i = 0; i < nro\_triangulos; i++){  
 **triangulo\_array\_circulo\_cafe2**.add(**new** Triangulo(  
 circuloVertices.getVerticesCirculo(i),  
 colores.getRgba(4)));  
 }  
  
 *// RAYOS BOMBA POKEMON* nro\_triangulos = 38+12+15;  
 TrianguloVertices trianguloVertices = **new** TrianguloVertices();  
 **for**(**int** i = 0; i <= nro\_triangulos; i++){  
 **triangulo\_array**.add(**new** Triangulo(  
 trianguloVertices.getVerticesTriangulo(i),  
 colores.getRgba(1)));  
 }  
  
 *// DIENTES* nro\_triangulos = 20;  
 trianguloVertices = **new** TrianguloVertices();  
 **for**(**int** i = 0; i < nro\_triangulos; i++){  
 **triangulo\_array\_dientes**.add(**new** Triangulo(  
 trianguloVertices.getVerticesTriangulo\_dientes(i),  
 colores.getRgba(5)));  
 }  
  
 *// LENGUA* nro\_triangulos = 9;  
 trianguloVertices = **new** TrianguloVertices();  
 **for**(**int** i = 0; i < nro\_triangulos; i++){  
 **triangulo\_array\_lengua**.add(**new** Triangulo(  
 trianguloVertices.getVerticesTriangulo\_lengua(i),  
 colores.getRgba(6)));  
 }  
  
 *// BOCA* nro\_triangulos = 16;  
 trianguloVertices = **new** TrianguloVertices();  
 **for**(**int** i = 0; i < nro\_triangulos; i++){  
 **triangulo\_array\_boca**.add(**new** Triangulo(  
 trianguloVertices.getVerticesTriangulo\_boca(i),  
 colores.getRgba(7)));  
 }  
  
 *// OJOS* nro\_triangulos = 14;  
 trianguloVertices = **new** TrianguloVertices();  
 **for**(**int** i = 0; i < nro\_triangulos; i++){  
 **triangulo\_array\_ojos**.add(**new** Triangulo(  
 trianguloVertices.getVerticesTriangulo\_ojos(i),  
 colores.getRgba(8)));  
 }  
  
 *// IRIS* nro\_triangulos = 2;  
 trianguloVertices = **new** TrianguloVertices();  
 **for**(**int** i = 0; i < nro\_triangulos; i++){  
 **triangulo\_array\_iris**.add(**new** Triangulo(  
 trianguloVertices.getVerticesTriangulo\_iris(i),  
 colores.getRgba(9)));  
 }  
  
 *// CUERPO MANO D* nro\_triangulos = 17;  
 trianguloVertices = **new** TrianguloVertices();  
 **for**(**int** i = 0; i < nro\_triangulos; i++){  
 **triangulo\_array\_mano\_d**.add(**new** Triangulo(  
 trianguloVertices.getVerticesTriangulo\_mano\_d(i),  
 colores.getRgba(10)));  
 }  
  
 *// CUERPO MANO I* nro\_triangulos = 14;  
 trianguloVertices = **new** TrianguloVertices();  
 **for**(**int** i = 0; i < nro\_triangulos; i++){  
 **triangulo\_array\_mano\_i**.add(**new** Triangulo(  
 trianguloVertices.getVerticesTriangulo\_mano\_i(i),  
 colores.getRgba(10)));  
 }  
 *// CUERPO PIES* nro\_triangulos = 19+16;  
 trianguloVertices = **new** TrianguloVertices();  
 **for**(**int** i = 0; i < nro\_triangulos; i++){  
 **triangulo\_array\_pies**.add(**new** Triangulo(  
 trianguloVertices.getVerticesTriangulo\_pies(i),  
 colores.getRgba(10)));  
 }  
  
 *// CUERPO* nro\_triangulos = 23+16+17+12+25;  
 trianguloVertices = **new** TrianguloVertices();  
 **for**(**int** i = 0; i < nro\_triangulos; i++){  
 **triangulo\_array\_cuerpo**.add(**new** Triangulo(  
 trianguloVertices.getVerticesTriangulo\_cuerpo(i),  
 colores.getRgba(10)));  
 }  
  
 *// QUINO* circuloVertices = **new** CirculoVertices(160,210,50);  
 **for**(**int** i = 0; i < 360; i++){  
 **triangulo\_array\_circulo\_cafe2**.add(**new** Triangulo(  
 circuloVertices.getVerticesCirculo(i),  
 colores.getRgba(8)));  
 }  
 nro\_triangulos = 17;*//17* trianguloVertices = **new** TrianguloVertices();  
 **for**(**int** i = 0; i < nro\_triangulos; i++){  
 **triangulo\_array\_quino**.add(**new** Triangulo(  
 trianguloVertices.getVerticesTriangulo\_quino(i),  
 colores.getRgba(10)));  
 }  
  
 */\* Color de fondo \*/* gl.glClearColor(0, 1, 1, 0);  
 *//gl.glClearColor(224f/255f, 247f/255f, 250f/255f, 0); // claro  
 //gl.glClearColor(50f/255f, 33f/255f, 35f/255f, 0); // oscuro* }  
 @Override  
 **public void** onDrawFrame(GL10 gl) {  
 */\* Inicializa el buffer de color \*/* gl.glClear(GL10.***GL\_COLOR\_BUFFER\_BIT***);  
  
 **for** (**int** i = 0; i < **triangulo\_array\_circulo\_blanco**.size(); i++){  
 **triangulo\_array\_circulo\_blanco**.get(i).dibuja(gl);  
 }  
 **for** (**int** i = 0; i < **triangulo\_array\_circulo\_cafe1**.size(); i++){  
 **triangulo\_array\_circulo\_cafe1**.get(i).dibuja(gl);  
 }  
 **for** (**int** i = 0; i < **triangulo\_array\_circulo\_cafe2**.size(); i++){  
 **triangulo\_array\_circulo\_cafe2**.get(i).dibuja(gl);  
 }  
  
 **for** (**int** i = 0; i < **triangulo\_array**.size(); i++){  
 **triangulo\_array**.get(i).dibuja(gl);  
 }  
 *// DIENTES* **for** (**int** i = 0; i < **triangulo\_array\_dientes**.size(); i++)  
 **triangulo\_array\_dientes**.get(i).dibuja(gl);  
 *// LENGUA* **for** (**int** i = 0; i < **triangulo\_array\_lengua**.size(); i++)  
 **triangulo\_array\_lengua**.get(i).dibuja(gl);  
 *// BOCA* **for** (**int** i = 0; i < **triangulo\_array\_boca**.size(); i++)  
 **triangulo\_array\_boca**.get(i).dibuja(gl);  
 *// OJOS* **for** (**int** i = 0; i < **triangulo\_array\_ojos**.size(); i++)  
 **triangulo\_array\_ojos**.get(i).dibuja(gl);  
 *// IRIS* **for** (**int** i = 0; i < **triangulo\_array\_iris**.size(); i++)  
 **triangulo\_array\_iris**.get(i).dibuja(gl);  
 *// CUERPO MANO DERECHA* **for** (**int** i = 0; i < **triangulo\_array\_mano\_d**.size(); i++)  
 **triangulo\_array\_mano\_d**.get(i).dibuja(gl);  
 *// CUERPO MANO IZQUIERDA* **for** (**int** i = 0; i < **triangulo\_array\_mano\_i**.size(); i++)  
 **triangulo\_array\_mano\_i**.get(i).dibuja(gl);  
 *// CUERPO PIES* **for** (**int** i = 0; i < **triangulo\_array\_pies**.size(); i++)  
 **triangulo\_array\_pies**.get(i).dibuja(gl);  
 *// CUERPO* **for** (**int** i = 0; i < **triangulo\_array\_cuerpo**.size(); i++)  
 **triangulo\_array\_cuerpo**.get(i).dibuja(gl);  
 *// QUINO* **for** (**int** i = 0; i < **triangulo\_array\_quino**.size(); i++)  
 **triangulo\_array\_quino**.get(i).dibuja(gl);  
  
 }  
}

**public class** Colores {  
 **private float rgba**[];  
  
 */\* Los colores x vértice (r,g,b,a) \*/* **private float lilas\_tonos**[] = **new float** [] {  
 228f/255f, 210f/255f, 231f/255f, 1, *// 0 lila* 215f/255f, 181f/255f, 216f/255f, 1, *// 1 lila* 201f/255f, 155f/255f, 203f/255f, 1, *// 2 lila* 189f/255f, 140f/255f, 195f/255f, 1, *// 3 lila* 177f/255f, 114f/255f, 182f/255f, 1, *// 4 lila* 164f/255f, 91f/255f, 170f/255f, 1, *// 5 lila* };  
 **private float azul\_tonos**[] = **new float** [] {  
 178f/255f, 235f/255f, 242f/255f, 1, *// 0 azul* 80f/255f, 222f/255f, 234f/255f, 1, *// 1 azul* 26f/255f, 198f/255f, 218f/255f, 1, *// 2 azul* 00f/255f, 172f/255f, 193f/255f, 1, *// 3 azul* 00f/255f, 97f/255f, 167f/255f, 1, *// 4 azul* 00f/255f, 83f/255f, 143f/255f, 1, *// 5 azul* };  
 **private float entero**[] = **new float** [] {  
 1, 0, 0, 1, *// 0 rojo* 0, 1, 0, 1, *// 1 verde* 0, 0, 1, 1, *// 2 azul* 0, 0, 0, 1, *// 3 negro* };  
 **private float rosado\_tonos**[] = **new float** [] {  
 253f/255f, 164f/255f, 186f/255f, 1, *// 0 blanco* 252f/255f, 186f/255f, 203f/255f, 1, *// 0 blanco* 254f/255f, 197f/255f, 229f/255f, 1, *// 0 blanco* 247f/255f, 154f/255f, 192f/255f, 1, *// 0 blanco* 253f/255f, 171f/255f, 159f/255f, 1, *// 0 blanco* 242f/255f, 107f/255f, 138f/255f, 1, *// 0 blanco* };  
 **private float gris\_tonos**[] = **new float** [] {  
 120f/255f, 120f/255f, 120f/255f, 1, *// 0 blanco* 130f/255f, 130f/255f, 130f/255f, 1, *// 0 blanco* 140f/255f, 140f/255f, 140f/255f, 1, *// 0 blanco* 150f/255f, 150f/255f, 150f/255f, 1, *// 0 blanco* 160f/255f, 160f/255f, 160f/255f, 1, *// 0 blanco* 170f/255f, 170f/255f, 170f/255f, 1, *// 0 blanco* };  
 **private float blanco\_tonos**[] = **new float** [] {  
 255f/255f, 255f/255f, 255f/255f, 1, *// 0 blanco* 250f/255f, 250f/255f, 250f/255f, 1, *// 0 blanco* 245f/255f, 245f/255f, 245f/255f, 1, *// 0 blanco* 240f/255f, 240f/255f, 240f/255f, 1, *// 0 blanco* 235f/255f, 235f/255f, 235f/255f, 1, *// 0 blanco* 230f/255f, 230f/255f, 230f/255f, 1, *// 0 blanco* };  
 **private float cafe\_tonos**[] = **new float** [] {  
 62f/255f, 27f/255f, 23f/255f, 1, *// 0 lila* 78f/255f, 34f/255f, 46f/255f, 1, *// 0 lila* 201f/255f, 155f/255f, 203f/255f, 1, *// 2 lila* 189f/255f, 140f/255f, 195f/255f, 1, *// 3 lila* 177f/255f, 114f/255f, 182f/255f, 1, *// 4 lila* 164f/255f, 91f/255f, 170f/255f, 1, *// 5 lila* };  
 **float**[] getRgba(**int** option){  
 **rgba** = **new float**[4];  
 **switch** (option){  
 **case** 1:  
 *// colores RGBA tonos de lila* **int** random = getRandomInt();  
 **rgba** = **new float**[4];  
 **rgba**[0] = **lilas\_tonos**[(random\*4) + 0];  
 **rgba**[1] = **lilas\_tonos**[(random\*4) + 1];  
 **rgba**[2] = **lilas\_tonos**[(random\*4) + 2];  
 **rgba**[3] = **lilas\_tonos**[(random\*4) + 3];  
 **break**;  
 **case** 2:  
 **rgba**[0] = **blanco\_tonos**[0];  
 **rgba**[1] = **blanco\_tonos**[1];  
 **rgba**[2] = **blanco\_tonos**[2];  
 **rgba**[3] = **blanco\_tonos**[3];  
 **break**;  
 **case** 3:  
 **rgba**[0] = **cafe\_tonos**[0];  
 **rgba**[1] = **cafe\_tonos**[1];  
 **rgba**[2] = **cafe\_tonos**[2];  
 **rgba**[3] = **cafe\_tonos**[3];  
 **break**;  
 **case** 4:  
 **rgba**[0] = **cafe\_tonos**[4];  
 **rgba**[1] = **cafe\_tonos**[5];  
 **rgba**[2] = **cafe\_tonos**[6];  
 **rgba**[3] = **cafe\_tonos**[7];  
 **break**;  
 **case** 5:  
 *// colores RGBA para dientes* random = getRandomInt();  
 **rgba** = **new float**[4];  
 **rgba**[0] = **blanco\_tonos**[(random\*4) + 0];  
 **rgba**[1] = **blanco\_tonos**[(random\*4) + 1];  
 **rgba**[2] = **blanco\_tonos**[(random\*4) + 2];  
 **rgba**[3] = **blanco\_tonos**[(random\*4) + 3];  
 **break**;  
 **case** 6:  
 *// colores RGBA para lengua* random = getRandomInt();  
 **rgba** = **new float**[4];  
 **rgba**[0] = **rosado\_tonos**[(random\*4) + 0];  
 **rgba**[1] = **rosado\_tonos**[(random\*4) + 1];  
 **rgba**[2] = **rosado\_tonos**[(random\*4) + 2];  
 **rgba**[3] = **rosado\_tonos**[(random\*4) + 3];  
 **break**;  
 **case** 7:  
 *// colores RGBA para boca* random = getRandomInt();  
 **rgba** = **new float**[4];  
 **rgba**[0] = **gris\_tonos**[(random\*4) + 0];  
 **rgba**[1] = **gris\_tonos**[(random\*4) + 1];  
 **rgba**[2] = **gris\_tonos**[(random\*4) + 2];  
 **rgba**[3] = **gris\_tonos**[(random\*4) + 3];  
 **break**;  
 **case** 8:  
 *// colores RGBA para ojos ROJO* **rgba** = **new float**[4];  
 **rgba**[0] = **entero**[0];  
 **rgba**[1] = **entero**[1];  
 **rgba**[2] = **entero**[2];  
 **rgba**[3] = **entero**[3];  
 **break**;  
 **case** 9:  
 *// colores RGBA para ojos ROJO* **rgba** = **new float**[4];  
 **rgba**[0] = **entero**[12];  
 **rgba**[1] = **entero**[13];  
 **rgba**[2] = **entero**[14];  
 **rgba**[3] = **entero**[15];  
 **break**;  
 **case** 10:  
 *// colores RGBA tonos de lila* random = getRandomInt();  
 **rgba** = **new float**[4];  
 **rgba**[0] = **azul\_tonos**[(random\*4) + 0];  
 **rgba**[1] = **azul\_tonos**[(random\*4) + 1];  
 **rgba**[2] = **azul\_tonos**[(random\*4) + 2];  
 **rgba**[3] = **azul\_tonos**[(random\*4) + 3];  
 **break**;  
 **default**:  
 **break**;  
 }  
 **return rgba**;  
 }  
 **public int** getRandomInt(){  
 **int** numero = (**int**)(Math.*random*()\*10);  
 **if** (numero > 5)  
 numero = numero - 4;  
 **return** numero;  
 }  
}

1. Salida de su código.



Repositorio: https://github.com/victordanielgithub/INF-323/tree/master/tarea02