Martinez Aguilar Diego Ulises
Tarea: 2 Grupo: 3
DIMyR

1. Crear un programa que dibuje dos triángulos consecutivos, esto se logra agregando más vértices a nuestro buffer (VBO), sin utilizar índices.

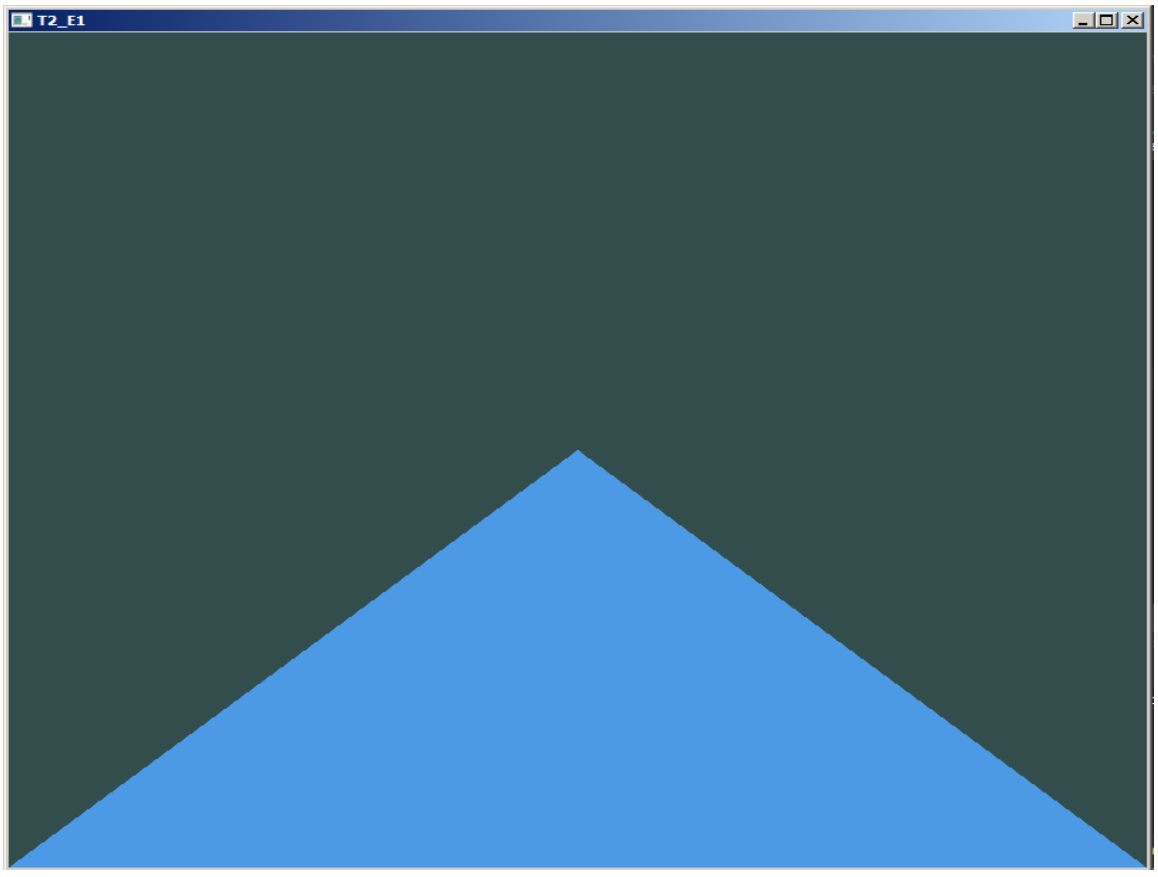
Para este ejercicio se colocaron los siguientes vértices:

```
GLfloat vertices[] = {

-1.0f, -1.0f, 0.0f,
 0.0f, 0.0f, 0.0f,
 0.0f, -1.0f, 0.0f,

 0.0f, -1.0f, 0.0f,
 1.0f, -1.0f, 0.0f,
 0.0f, 0.0f, 0.0f,
 };
```

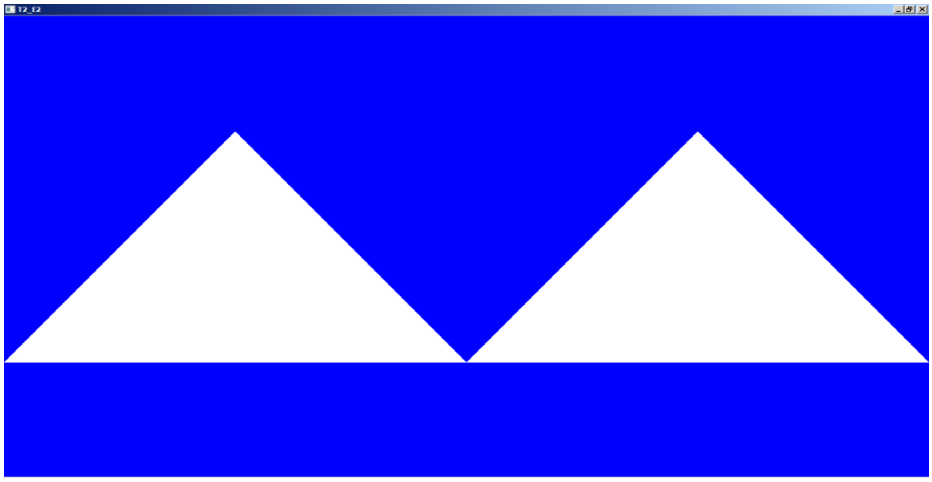
Y agregar el número de vértices: glDrawArrays(GL_TRIANGLES, 0, 6);



Martinez Aguilar Diego Ulises
Tarea: 2 Grupo: 3
DIMyR

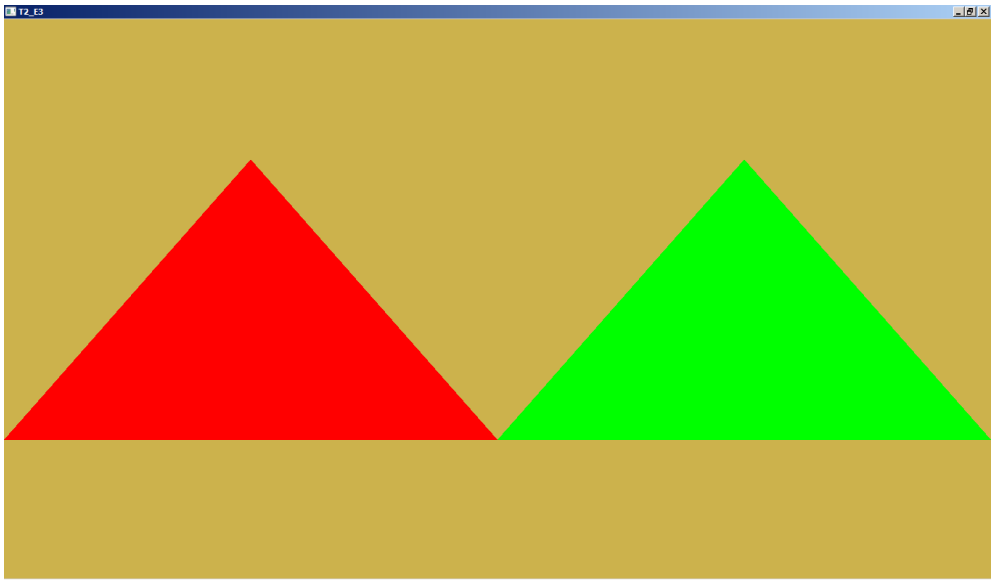
2. Crear un programa que dibuje dos triángulos usando dos diferentes VBOs y VAOs para los datos.

   Se declaró un VBO y un VBA extras y se añadieron al código dos grupos de vértices, uno para cada triangulo. Cada uno se cargó en su propio buffer y arreglo.



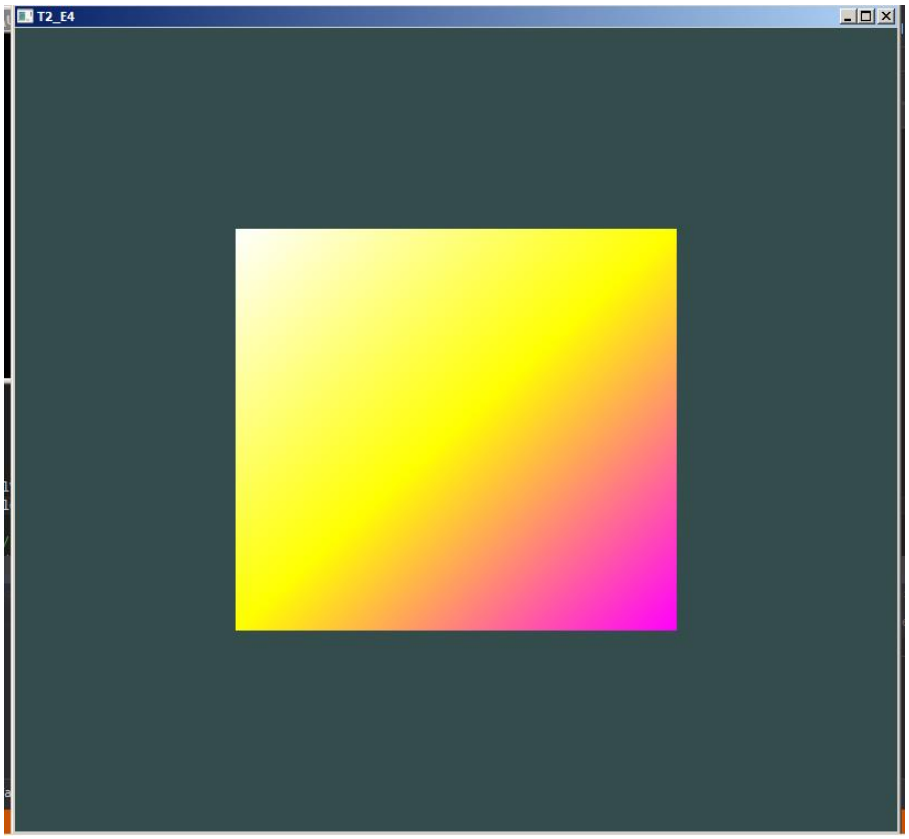
3. Crear un programa que dibuje dos triángulos con dos shaders diferentes, un shader que dibuje un triángulo de color rojo y otro de color verde.

   Se crearon dos fragment shader donde en el primero se colocó el color verde y en otro el color rojo. También se crearon dos shaderprogram para que vincular cada uno de los fragment. Finalmente se mandan a llamar en el GLApplication::applicationLoop().



4. Crear un programa que dibuje un cubo, el color del cubo debe pasarse al shader como atributo de vértice, como se mostró en la clase, se debe de ver como se hace interpolación de los colores

Se crearon los arreglos para cada uno de los vértices y para los colores de cada uno, posteriormente se cargaron en el shader.



5. Desde el shader ajustar el triángulo para que se dibuje invertido.

   Se modificaron los ejes de referencia para cambiar la posición de nuestro triangulo.

6. Con el ejercicio 4 hacer que el cubo rote, para ello se necesita manejar los eventos del teclado, posición y click. Los movimientos debe de realizarse en los dos ejes.

7. Actualice el ejercicio 7 para que el cubo tenga mapeada una textura.