



UNIVERSIDAD NACIONAL AUTÓNOMA DE
MÉXICO

FACULTAD DE INGENIERÍA

DIVISIÓN DE INGENIERÍA ELÉCTRICA

INGENIERÍA EN COMPUTACIÓN

LABORATORIO DE COMPUTACIÓN GRÁFICA e
INTERACCIÓN HUMANO COMPUTADORA



EJERCICIOS DE CLASE N° 01

NOMBRE COMPLETO: Caballero Garcia Yessica Lizeth

N° de Cuenta: 319104125

GRUPO DE LABORATORIO: 02

GRUPO DE TEORÍA: 04

SEMESTRE: 2026-1

FECHA DE ENTREGA LÍMITE: 19 agosto 2025

CALIFICACIÓN: _____

EJERCICIOS DE SESIÓN:

1. Actividades realizadas.

En la actividad en clase se buscó dibujar un cuadrado y un rombo sin que haya traslape entre ellos.

El fondo debe ser variable y cíclico entre los colores rojo -> verde -> azul. Para lograrlo, se obtiene el tiempo actual desde que se inició el programa y se divide en la cantidad de segundos que se desea cada color. Con la operación modulo se obtienen cantidades entre 0, 1 y 2 (ya que son 3 colores) y así el cambio se vuelve cíclico.

```
//CAMBIAR COLOR DE FONDO
float tiempo = glfwGetTime();
int estado = static_cast<int>(tiempo / 2.0f) % 3;
```

Usando una estructura de control switch, el programa sabrá que color es el correspondiente para cada estado:

```
switch (estado) {
case 0:
    RED = 1.0f; GREEN = 0.0f; BLUE = 0.0f;
    break;
case 1:
    RED = 0.0f; GREEN = 1.0f; BLUE = 0.0f;
    break;
case 2:
    RED = 0.0f; GREEN = 0.0f; BLUE = 1.0f;
    break;
}
```

Debido a que las formas se podrían confundir con el color de fondo, modifiqué el color de las formas:

```

//Fragment Shader
//recibir Vcolor y dar de salida color
static const char* fShader = "
#version 330
out vec4 color;
void main()
{
    color = vec4(0.8f,1.0f,0.4f,1.0f);
}";

```

Dibujé 4 triángulos en total: 2 triángulos formaron el cuadrado en la parte superior izquierda y otros 2 triángulos formaron el rombo en la parte inferior derecha.

Declarando los vértices:

```

GLfloat vertices[] = {
    -0.8f,0.8f,0.0f,
    -0.6f,0.8f,0.0f,
    -0.8f,0.6f,0.0f,
    -0.6f,0.8f,0.0f,
    -0.8f,0.6f,0.0f,
    -0.6f,0.6f,0.0f,
    0.6f,-0.6f,0.0f,
    0.4f,-0.8f,0.0f,
    0.8f,-0.8f,0.0f,
    0.4f,-0.8f,0.0f,
    0.8f,-0.8f,0.0f,
    0.6f,-1.0f,0.0f,
};

```

Dibujando los triángulos a partir de los 12 vértices:

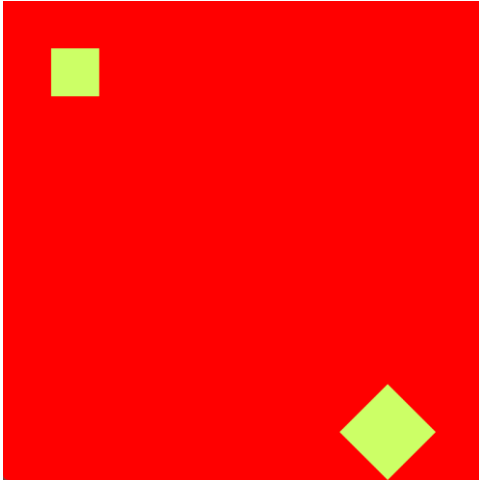
```

glBindVertexArray(VAO);
glDrawArrays(GL_TRIANGLES,0,12);
glBindVertexArray(0);

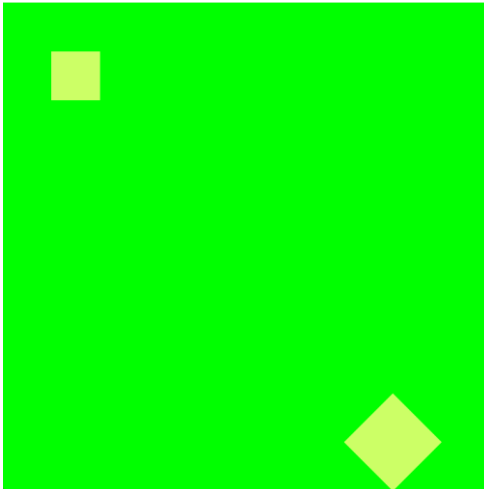
```

EJECUCIÓN DEL PROGRAMA

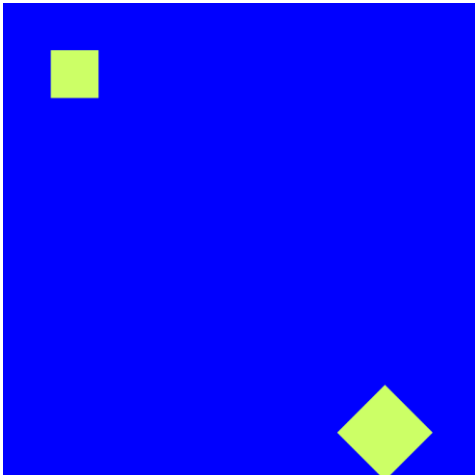
Primer estado



Segundo estado



Tercer estado



2. Problemas presentados.

Un problema que tuve al momento de ejecutar es que únicamente me mostraba un triángulo, eso fue debido a que olvidé modificar el número de vértices y únicamente tenía los necesarios para un triángulo en lugar de cuatro.

3. Conclusión

Este primer ejercicio no fue especialmente complicado. Lo que más se me facilitó fue definir los puntos para trazar los triángulos, ya que esa parte me pareció bastante intuitiva. Sin embargo, al ser el primer ejercicio del semestre, todavía me estoy adaptando al ritmo de trabajo.

Durante la clase, la resolución de dudas fue buena, aunque creo que la explicación podría profundizar un poco más en los tipos de archivos que conforman el proyecto. Tener más claridad en ese aspecto ayudaría a organizar mejor los entregables desde el inicio.