



**Escuela Superior  
de Ingeniería y Tecnología**  
Universidad de La Laguna

# Programación Optimizada para Videojuegos:

## Cubo de colores en OpenTK

Adrián Mora Rodríguez ([alu0101465883@ull.edu.es](mailto:alu0101465883@ull.edu.es))

C/ Padre Herrera s/n  
38207 La Laguna  
Santa Cruz de Tenerife. España

T: 900 43 25 26

**ull.es**



<b>1. Cambios realizados</b>	<b>2</b>
<b>2. Resultado</b>	<b>3</b>
<b>3. Dificultades encontradas</b>	<b>3</b>
<b>4. Summary</b>	<b>3</b>



# 1. Cambios realizados

El cambio principal se hizo en la definición de los arrays de vértices e índices. Anteriormente, el cubo solo tenía 8 vértices compartidos entre las caras, lo que hacía que no se pudieran aplicar colores diferentes por cara. Para solucionarlo, se duplicaron los vértices de modo que cada cara tenga sus propios vértices con su propio color asignado. El nuevo bloque de código con los datos de los vértices quedó así:

```
public float[] vertexData = {  
    // Cara frontal  
    -0.5f, -0.5f, 0.5f, 1.0f, 0.0f, 0.0f, 1.0f, // 0  
    0.5f, -0.5f, 0.5f, 1.0f, 0.0f, 0.0f, 1.0f, // 1  
    0.5f, 0.5f, 0.5f, 1.0f, 0.0f, 0.0f, 1.0f, // 2  
    -0.5f, 0.5f, 0.5f, 1.0f, 0.0f, 0.0f, 1.0f, // 3  
    // Cara trasera  
    -0.5f, -0.5f, -0.5f, 0.0f, 1.0f, 0.0f, 1.0f, // 4  
    0.5f, -0.5f, -0.5f, 0.0f, 1.0f, 0.0f, 1.0f, // 5  
    0.5f, 0.5f, -0.5f, 0.0f, 1.0f, 0.0f, 1.0f, // 6  
    -0.5f, 0.5f, -0.5f, 0.0f, 1.0f, 0.0f, 1.0f, // 7  
    // Cara izquierda  
    -0.5f, -0.5f, -0.5f, 0.0f, 0.0f, 1.0f, 1.0f, // 8  
    -0.5f, -0.5f, 0.5f, 0.0f, 0.0f, 1.0f, 1.0f, // 9  
    -0.5f, 0.5f, 0.5f, 0.0f, 0.0f, 1.0f, 1.0f, // 10  
    -0.5f, 0.5f, -0.5f, 0.0f, 0.0f, 1.0f, 1.0f, // 11  
    // Cara derecha  
    0.5f, -0.5f, -0.5f, 1.0f, 1.0f, 0.0f, 1.0f, // 12  
    0.5f, -0.5f, 0.5f, 1.0f, 1.0f, 0.0f, 1.0f, // 13  
    0.5f, 0.5f, 0.5f, 1.0f, 1.0f, 0.0f, 1.0f, // 14  
    0.5f, 0.5f, -0.5f, 1.0f, 1.0f, 0.0f, 1.0f, // 15  
    // Cara superior  
    -0.5f, 0.5f, -0.5f, 1.0f, 0.0f, 1.0f, 1.0f, // 16  
    0.5f, 0.5f, -0.5f, 1.0f, 0.0f, 1.0f, 1.0f, // 17  
    0.5f, 0.5f, 0.5f, 1.0f, 0.0f, 1.0f, 1.0f, // 18  
    -0.5f, 0.5f, 0.5f, 1.0f, 0.0f, 1.0f, 1.0f, // 19  
    // Cara inferior
```



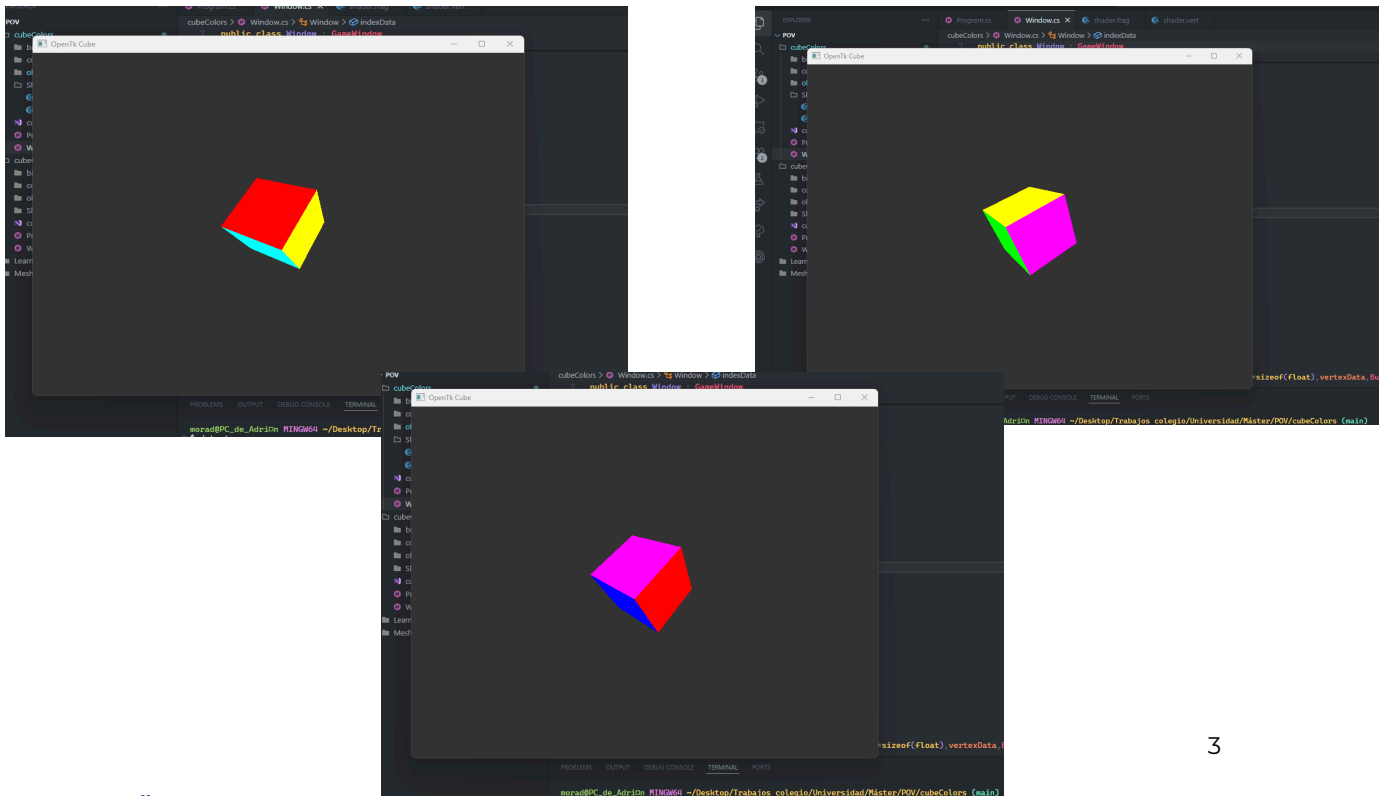
```
-0.5f, -0.5f, -0.5f,  0.0f, 1.0f, 1.0f, 1.0f, // 20
 0.5f, -0.5f, -0.5f,  0.0f, 1.0f, 1.0f, 1.0f, // 21
 0.5f, -0.5f,  0.5f,  0.0f, 1.0f, 1.0f, 1.0f, // 22
-0.5f, -0.5f,  0.5f,  0.0f, 1.0f, 1.0f, 1.0f // 23
};
```

Y los índices se actualizaron para dibujar cada cara correctamente:

```
public int[] indexData = {
    0, 1, 2, 2, 3, 0, //
    4, 7, 6, 4, 6, 5, //
    8, 9, 10, 10, 11, 8, //
    12, 14, 13, 14, 12, 15, //
    16, 18, 17, 18, 16, 19, //
    20, 21, 22, 22, 23, 20 //
};
```

## 2. Resultado

Al ejecutar el programa, el cubo rota como antes, pero ahora cada cara muestra un color diferente.





### 3. Dificultades encontradas

Tuve problemas en entender el orden en que hay que definir los vértices. Sabía que debían estar en sentido antihorario para que OpenTK los considerara como la cara visible. Tras una tutoría entendí que no estaba teniendo en cuenta la posición de la cámara y las rotaciones.

### 4. Summary

In this exercise, I modified the rotating cube project to display a different color on each face. To achieve this, I created a new vertex array where each face has its own set of vertices with unique color values. The index array was also updated to draw all six faces correctly. The main difficulty I found was understanding the correct vertex order. I knew they had to be defined counterclockwise, but even doing that, some faces didn't render properly. As a result, the cube now rotates while showing six different colors, one per face.