



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° 08

NOMBRE COMPLETO: Vázquez Reyes Sebastián

N° de Cuenta: 318150923

GRUPO DE LABORATORIO: 11

GRUPO DE TEORÍA: 6

SEMESTRE 2024-2

FECHA DE ENTREGA LÍMITE: sábado 6 de abril de 2024

CALIFICACIÓN: _____

EJERCICIOS DE SESIÓN:

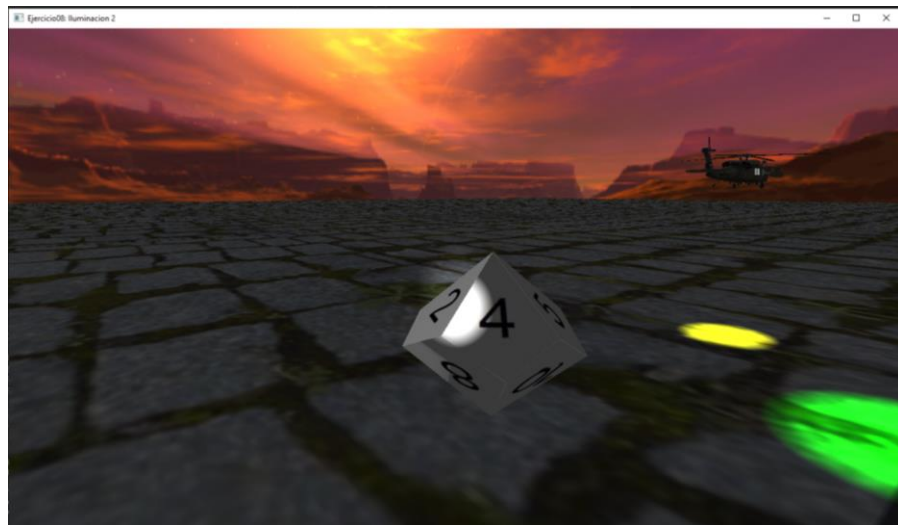
- **Actividades realizadas. Una descripción de los ejercicios y capturas de pantalla de bloques de código generados y de ejecución del programa**

1. Agregar su dado de 10 caras y editar sus normales para que las caras del dado sean iluminadas correctamente.

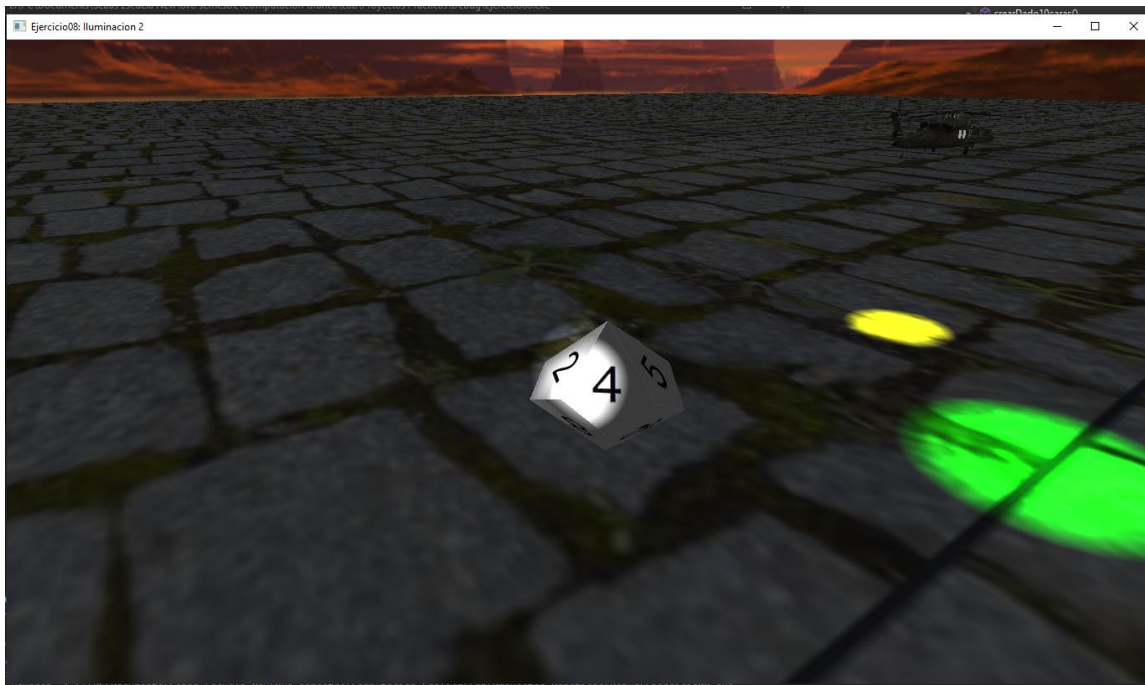
Esta sección del código fue sencilla de realizar, como el dado ya estaba construido solo tuve que modificar sus valores de normales para que la luz se refleje a base de pura prueba y error. Sin embargo, hay algunas caras dentro de mi código a las que les asigne valores distintos de normales en sus vértices, por ejemplo, para la cara 2, sus vértices tienen los siguientes valores:

```
// CARA 2
//x      y      z      S      T
0.5f, +0.0f, 0.5f, 0.3145f, 0.8906f, -1.0f, -1.0f, -1.0f,
1.f, -0.25f, 0.1f, 0.1875f, 0.8047f, -1.0f, -1.0f, -1.0f,
0.9f, +0.0f, -0.5f, 0.2266f, 0.6895f, -1.0f, -1.0f, 1.0f,
0.0f, 0.75f, -0.25f, 0.5957f, 0.7324f, -1.0f, -1.0f, -1.0f,
```

Vemos que los vértices de la cara 2 poseen los mismos valores de normales, salvo en X, donde el tercer vértice tiene un valor positivo. Cuando estamos rodeando el dado con la linterna de la cámara y apuntamos a una cara, esta se ilumina, pero hay casos en los que las esquinas de sus caras vecinas no se iluminan como deberían debido a la dirección en la que se apunta la linterna. Modificando un vértice de algunas caras, podemos hacer que se iluminen parcialmente cuando estamos iluminando a su cara vecina. Por ejemplo, así se ve la cara 2 iluminada cuando todos sus valores de normales en X son negativos y estamos concentrados en la cara 4



Y así se ve con el tercer vértice con normal positiva en X



Este mismo fenómeno se aplica a varias caras de mi dado.

2. Apagar con teclado la luz (pointlight) de su lámpara creada para el reporte de la práctica 7.

Para apagar y encender la luz pointlight de la lampara usé el siguiente código:

```

}
if (key == GLFW_KEY_K && action == GLFW_PRESS)
{
    theWindow->articulacion5 += 1.0;
    theWindow->articulacion6 = 0.0;
}
if (key == GLFW_KEY_L && action == GLFW_PRESS)
{
    theWindow->articulacion6 += 1.0;
    theWindow->articulacion5 = 0.0;
}

```

Al presionar la letra L, la luz de la lampara se enciende, y al presionar la letra K, se apaga. La forma en que se apaga la luz es la siguiente:

```

if(mainWindow.getarticulacion6() > 0.0) {
    shaderList[0].SetPointLights(pointLights, pointLightCount); //para apa
}

if (mainWindow.getarticulacion5() > 0.0) {
    shaderList[0].SetPointLights(pointLights, pointLightCount - 1);
}

```

Como en mi código solo tengo 2 luces de tipo pointlight, no tuve que utilizar el segundo arreglo de este tipo de luces. Sin embargo, debido a que el shader recibe la lista de luces hasta que presionamos una tecla, la luz roja puntual dentro del arreglo aparecerá tan pronto prendamos o apaguemos la luz de la lampara, pero una vez aparezca, no será modificada, solo la luz de la lampara.

La ejecución de este código esta en el video anexado a este documento.

- **Problemas Presentados**

No hubo ningún problema para la realización de este ejercicio

- **Conclusión:**

En este ejercicio, aprendí como funcionan las normales de un objeto creado en OpenGL para poder ser iluminado. Además, también aprendí como se encienden y apagan luces en este entorno.