



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

DIVISIÓN DE INGENIERÍA ELÉCTRICA

INGENIERÍA EN COMPUTACIÓN

LABORATORIO DE COMPUTACIÓN GRÁFICA e  
INTERACCIÓN HUMANO COMPUTADORA



## **REPORTE DE PRÁCTICA N° 08**

**NOMBRE COMPLETO:** Mino Guzmán Yara Amairani

**N° de Cuenta:** 422017028

**GRUPO DE LABORATORIO:** 03

**GRUPO DE TEORÍA:** 04

**SEMESTRE 2025-2**

**FECHA DE ENTREGA LÍMITE:** 12 abril 2025

**CALIFICACIÓN:** \_\_\_\_\_

## REPORTE DE PRÁCTICA:

### 1.- Actividades:

- Agregar un spotlight (que no sea luz de color blanco ni azul) que parta del cofre de su coche y abrir y cerrar el cofre ilumine en esa dirección.

```
//Luz del cofre del coche
spotLights[1] = SpotLight(1.0f, 0.5f, 0.0f, //color naranja
    1.5f, 1.5f, //intensidad difusa y especular
    7.0f, 2.0f, 2.5f, //posicion aproximada del cofre
    0.0f, -0.2f, 1.0f, //direccion hacia el frente
    1.0f, 0.1f, 0.01f, //atenuacion
    15.0f); //angulo del cono
spotLightCount++;
```

Cree un spotlight de color naranja para que fuera la luz de mi cofre.

```
//Activar spotlight del cofre si está abierto
if (mainWindow.getrotaCofre() < 0.0f) {
    glm::vec3 posCofre = glm::vec3(7.0f, 2.0f, -1.0f + mainWindow.getmueveCarro());
    glm::vec3 dirCofre = glm::vec3(0.0f, -0.3f, 1.0f);
    spotLights[1].SetFlash(posCofre, dirCofre);
}
else {
    spotLights[1].SetFlash(glm::vec3(0.0f), glm::vec3(0.0f)); // Apagar luz si cofre está cerrado
}
```

Cree un if para que la luz se active únicamente cuando el cofre se abra y se apague cuando el cofre se cierre.

```
GLfloat getrotaCofre() { return rotaCofre; }
```

Se ocupó el getrotaCofre que ya se ha usado en prácticas anteriores.

### Ejecución:



- Agregar luz de tipo spotlight para el coche de tal forma que al avanzar (mover con teclado hacia dirección de X negativa) ilumine con un spotlight hacia adelante y al retroceder (mover con teclado hacia dirección de X positiva) ilumine con un spotlight hacia atrás. Son dos spotlights diferentes que se prenderán y apagarán de acuerdo con alguna bandera asignada por ustedes.

```

//Faro del coche delantero
spotLights[2] = SpotLight(0.0f, 0.0f, 1.0f, //color azul
    1.0f, 2.0f, //intensidad difusa y especular
    7.5f, 0.0f, 0.0f, //Posicion relativa al auto
    0.0f, 0.0f, 1.2f, //Direccion hacia adelante
    1.0f, 0.0f, 0.0f, //Atenuacion
    20.0f //Angulo de apertura
);
spotLightCount++;

//Faro del coche trasero
spotLights[3] = SpotLight(1.0f, 0.0f, 0.0f, // color rojo
    1.0f, 2.0f, // intensidad difusa y especular
    7.5f, 0.0f, -3.0f, // posición aproximada trasera
    0.0f, 0.0f, -1.2f, // dirección hacia atrás
    1.0f, 0.0f, 0.0f, // atenuación
    20.0f); // ángulo
spotLightCount++;

```

El faro del coche delantero ya estaba creado, así que únicamente cree el faro del coche trasero.

```

GLfloat getBanderaDelantera() { return BanderaDelantera; }
GLfloat getBanderaTrasera() { return BanderaTrasera; }

```

Declaré BanderaDelantera y BanderaTrasera en Window.h.

```

//Trasladar el carro
// Mover coche hacia adelante (G)
if (key == GLFW_KEY_G) {
    theWindow->mueveCarro += 0.5;
    theWindow->rotaLlantas -= 10.0;

    // Activar faro delantero, desactivar trasero
    theWindow->BanderaDelantera = 1.0f;
    theWindow->BanderaTrasera = 0.0f;
}

// Mover coche hacia atrás (H)
if (key == GLFW_KEY_H) {
    theWindow->mueveCarro -= 0.5;
    theWindow->rotaLlantas += 10.0;

    // Activar faro trasero, desactivar delantero
    theWindow->BanderaDelantera = 0.0f;
    theWindow->BanderaTrasera = 1.0f;
}

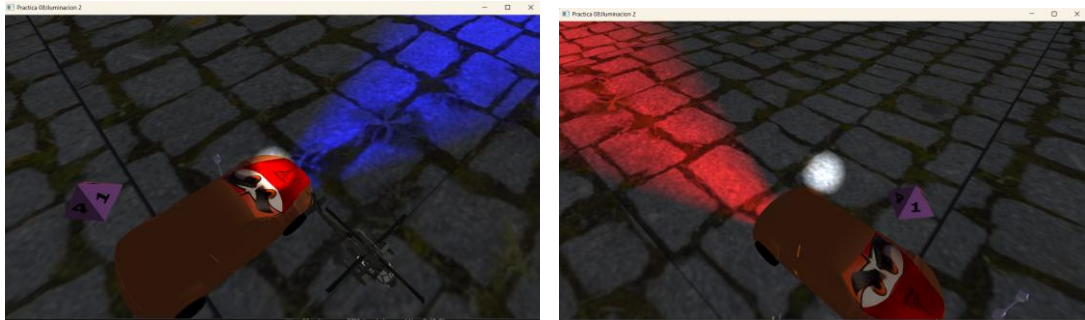
```

En el movimiento hacia adelante y hacia atrás del auto, agregué las banderas de mis luces, para que se movieran de acuerdo con mi auto.

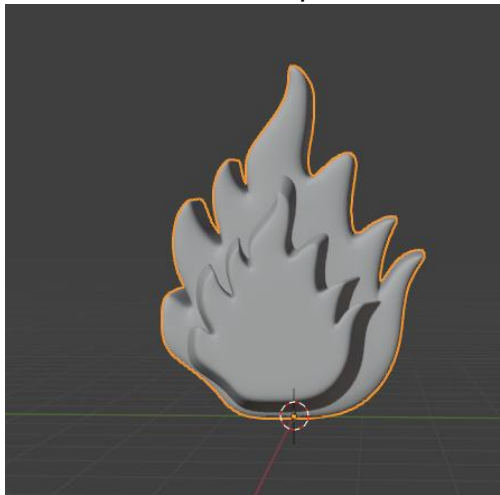
```
//Encender/apagar faros del auto
if (mainWindow.getBanderaDelantera() > 0.0f) {
    glm::vec3 posFrente = glm::vec3(7.5f, 0.0f, 0.0f + mainWindow.getmueveCarro());
    glm::vec3 dirFrente = glm::vec3(0.0f, 0.0f, 1.0f);
    spotLights[2].SetFlash(posFrente, dirFrente);
}
else {
    spotLights[2].SetFlash(glm::vec3(0.0f), glm::vec3(0.0f));
}
if (mainWindow.getBanderaTrasera() > 0.0f) {
    glm::vec3 posTrasera = glm::vec3(7.0f, 0.0f, -3.0f + mainWindow.getmueveCarro());
    glm::vec3 dirTrasera = glm::vec3(0.0f, 0.0f, -1.0f);
    spotLights[3].SetFlash(posTrasera, dirTrasera);
}
else {
    spotLights[3].SetFlash(glm::vec3(0.0f), glm::vec3(0.0f));
}
```

Cree este if para que los faros del coche se encendieran o apagaran de acuerdo con el movimiento del auto; si el auto va hacia adelante, se enciende el faro azul, si el auto va hacia atrás, se enciende el faro rojo.

### Ejecución:



- Agregar otra luz de tipo puntual ligada a un modelo elegido por ustedes (no lámpara) y que puedan prender y apagar de forma independiente con teclado tanto la luz de la lámpara como la luz de este modelo.



Escogí al fuego, ya que mi personaje (Furia), cuando se enoja saca fuego.

```
//Luz puntual blanca para la lámpara
pointLights[1] = PointLight(1.0f, 1.0f, 1.0f, // Color blanco
    1.0f, 1.0f, // Intensidad difusa y especular
    10.5f, 5.0f, 3.0f, // Posición en la parte superior de la lámpara
    0.3f, 0.2f, 0.1f); // Atenuación (constante, lineal, cuadrática)
pointLightCount++;

//Luz puntual naranja para el fuego
pointLights[2] = PointLight(1.0, 0.5f, 0.0f,
    1.5f, 1.5f,
    -4.5f, 1.5f, 5.0f,
    0.3f, 0.2f, 0.1f);
pointLightCount++;
```

Desde la práctica anterior, ya tenía creada mi luz puntual para la lámpara, así que solo cree mi luz puntual para el fuego.

```
GLfloat getFuego() { return Fuego; }
```

Declaré getFuego para la luz de mi fuego en Window.h.

```
//Encender y apagar la luz fuego
if (key == GLFW_KEY_3)
{
    theWindow->Fuego = 0.0f;
}

if (key == GLFW_KEY_4)
{
    theWindow->Fuego = 1.0f;
}
```

Le asigné las teclas a mi luz en Window.cpp.

```
// Encender y apagar lámpara y fuego
if (mainWindow.getValor() == 1.0f && mainWindow.getFuego() == 1.0f) {
    // Ambas luces encendidas
    pointLights[0] = pointLights[1]; // copia lámpara
    pointLights[1] = pointLights[2]; // copia fuego
    pointLightCount = 2;
}
else if (mainWindow.getValor() == 1.0f && mainWindow.getFuego() == 0.0f) {
    // Solo lámpara encendida
    pointLights[0] = pointLights[1]; // copia lámpara
    pointLightCount = 1;
}
else if (mainWindow.getValor() == 0.0f && mainWindow.getFuego() == 1.0f) {
    // Solo fuego encendido
    pointLights[0] = pointLights[2]; // copia fuego
    pointLightCount = 1;
}
else {
    // Ambas apagadas
    pointLightCount = 0;
}
```

Cree este if para que de acuerdo con las teclas que les asigné en Window.cpp, se prendieran y apagaran de manera independiente.

Ejecución: Se mostrará en un video que subiré a classroom.

## 2.- Problemas a la hora de hacer las actividades solicitadas:

- Realmente no presenté problemas al desarrollar la práctica, ya que todo se ha hecho con cosas que ya se han visto anteriormente.

## 3.- Conclusión:

- a. Los ejercicios del reporte fueron los adecuados para poder comprender el uso de la iluminación y como asignarles movimiento junto con nuestros modelos; así como crear luces puntuales para ciertos objetos y que prendan y apaguen utilizando el teclado.
- b. Las explicaciones fueron claras y, gracias a eso, se realizó con éxito la práctica.
- c. En conclusión, la práctica me ayudó a explorar nuevas cosas de la computación gráfica, también me ayudó a entender el uso de pointlight y spotlight y a seguir mejorando en la texturización.