



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:** Vázquez Reyes Sebastián

**N° de Cuenta:** 318150923

**GRUPO DE LABORATORIO:** 11

**GRUPO DE TEORÍA:** 6

**SEMESTRE 2024-2**

**FECHA DE ENTREGA LÍMITE:** miércoles 10 de abril de 2024

**CALIFICACIÓN:** \_\_\_\_\_

## REPORTE DE PRÁCTICA:

### 1.1.- Agregar un spotlight (que no sea luz de color blanco) que ilumine a su puerta creada para el previo.

La luz que ilumina mi puerta es de color morado y fue creada de la siguiente manera:

```
//luz de la puerta
spotLights[1] = SpotLight(1.0f, 0.0f, 1.0f,
    1.0f, 2.0f,
    0.0f, 0.0f, 10.0,
    0.0f, 5.0f, 10.0f,
    1.0f, 0.0f, 0.0005f,
    40.0f); //que tan amplia es la zona de il
spotLightCount++;
```

Los modelos de la puerta y el cartel son instanciados en las siguientes líneas:

```
//Puerta
model = glm::mat4(1.0);
model = glm::translate(model, glm::vec3(0.0f, 0.0f, 28.0));
model = glm::scale(model, glm::vec3(0.2f, 0.2f, 0.2f));
glUniformMatrix4fv(uniformModel, 1, GL_FALSE, glm::value_ptr(model));
puerta.RenderModel();

//cartel
model = glm::mat4(1.0);
model = glm::translate(model, glm::vec3(0.0f, 13.0f, 28.0));
model = glm::scale(model, glm::vec3(0.1f, 0.1f, 0.1f));
model = glm::rotate(model, -180 * toRadians, glm::vec3(0.0f, 1.0f, 0.0f));
glUniformMatrix4fv(uniformModel, 1, GL_FALSE, glm::value_ptr(model));
cartel.RenderModel();
```

El cartel tiene una textura de pared blanca, pero debido a la luz, se ve morado. Sin la luz, el prisma que funciona como cartel se ve de la siguiente forma:



## 1.2.- Agregar luz de tipo spotlight para el coche de tal forma que al avanzar ilumine con un spotlight hacia adelante y al retroceder ilumine con un spotlight hacia atrás.

El coche se mueve usando las teclas U y Y. Para este ejercicio, use el coche que ya viene en los modelos proporcionados, y no el coche que texturicé en prácticas anteriores.

Cuando se presiona la tecla U o Y, una de las articulaciones reinicia sus valores, la otra aumenta de valor de 1 en 1 (esta articulación es la que activa la condición para saber que luz se apaga y enciende), y otras 2 se encargan de agregar y restar valores a la traslación de las luces delanteras y traseras del carro, de esta manera cuando avanza el carro, la luz delantera y trasera lo hacen también, pero solo la luz delantera se enciende. “muevex” sirve para mover únicamente al carro.

```
}  
  
if (key == GLFW_KEY_Y)  
{  
    theWindow-> muevex += 0.2;  
    theWindow->articulacion2 += 0.2;  
    theWindow->articulacion3 -= 0.2;  
    theWindow->articulacion1 += 1.0;  
    theWindow->articulacion7 = 0.0;  
}  
  
if (key == GLFW_KEY_U)  
{  
    theWindow-> muevex -= 0.2;  
    theWindow->articulacion2 -= 0.2;  
    theWindow->articulacion3 += 0.2;  
    theWindow->articulacion7 += 1.0;  
    theWindow->articulacion1 = 0.0;  
}  
}
```

En las siguientes condiciones se lleva a cabo el encendido o apagado de luces, similar al de la lampara de la práctica pasada.

```
if (mainWindow.getarticulacion1() > 0.0) {  
    shaderList[0].SetSpotLights(spotLights, spotLightCount-1);  
    spotLights[2].SetPos(luzcarro);  
    spotLights2[2].SetPos(luzcarro2);  
}  
  
if (mainWindow.getarticulacion7() > 0.0) {  
    shaderList[0].SetSpotLights(spotLights2, spotLightCount-1);  
    spotLights2[2].SetPos(-luzcarro2);  
    spotLights[2].SetPos(-luzcarro);  
}
```

Los vectores que ayudan a actualizar la posición de las luces del carro son los siguientes:

```
//Información de Shader de Fuentes de Iluminación  
glm::vec3 luzcarro = glm::vec3(33.0f+mainWindow.getarticulacion2(), 2.0f, 1.5f);  
glm::vec3 luzcarro2 = glm::vec3(-40.0f+mainWindow.getarticulacion3(), 6.0f, -3.0f);  
spotLights[2].SetPos(luzcarro);  
spotLights2[2].SetPos(luzcarro2);
```

Por último, las luces creadas para el carro son las siguientes:

```
//Luz trasera del carro
spotLights[2] = SpotLight(0.0f, 0.0f, 1.0f, //rojo, verde y azul
    1.0f, 2.0f, //ambiental y difuso
    -40.0f, 6.0f, -3.0f, //pocision
    -10.0f, 2.0f, 0.0f, //direccion
    1.0f, 0.0f, 0.005f, //valores de la ecuacion cuadratica
    40.0f); //Angulo de apertura del cono del spotlight
spotLightCount++;

//Luz delantera del carro
spotLights[3] = SpotLight(0.0f, 0.0f, 1.0f,
    1.0f, 2.0f,
    33.0f, 2.0f, 1.5f,
    10.0f, 2.0f, 0.0f,
    1.0f, 0.0f, 0.005f,
    40.0f); //que tan amplia es la zona de iluminacion, sirve para
spotLightCount++;
```

### 1.3.- Agregar otra luz de tipo puntual ligada a un modelo elegido por ustedes y que puedan prender y apagar con teclado tanto la luz de la lámpara como la luz de este modelo

El modelo que escogí es una hamburguesa, que se enciende con una luz roja simulando un sobrecalentamiento en un microondas. Tanto la lampara como la hamburguesa se encienden y apagan con las mismas teclas: L y K. La luz de la hamburguesa es la siguiente:

```
//Luz de la anvorguesa
pointLights[0] = PointLight(1.0f, 0.0f, 0.0f,
    2.0f, 2.0f,
    -10.0f, 3.0f, 0.0f,
    0.0f, 0.7f, 0.5);
pointLightCount++;
```

### 2.- Liste los problemas que tuvo a la hora de hacer estos ejercicios y si los resolvió explicar cómo fue, en caso de error adjuntar captura de pantalla

Cuando utilice los arreglos de luces spotlights para el carro, la luz de la linterna desaparecía cuando apagaba una de las luces del carro. Esto se debía a que solo la luz de la linterna en el arreglo 1 se actualizaba con los valores de la cámara en tiempo de ejecución, para corregirlo, copie la línea de código que le da los valores de la cámara a la linterna para la luz de la linterna del arreglo 2:

```
spotLights[0].setFlash(lowerLight, camera.getCameraDirection());
spotLights2[0].setFlash(lowerLight, camera.getCameraDirection());
```

### 3.- Conclusión:

#### a. Los ejercicios del reporte: Complejidad, Explicación.

Fue un ejercicio relativamente sencillo, no me tomo mucho tiempo resolver lo que se pedía, lo más tedioso fue únicamente acomodar las luces del carro para que avanzaran y retrocedieran como debían hacerlo.

**b. Comentarios generales: Faltó explicar a detalle, ir más lento en alguna explicación, otros comentarios y sugerencias para mejorar desarrollo de la práctica**

Gracias a la explicación de las ultimas 2 clases, no hubo problemas para nada. Aunque estaría bien ahondar en un par de detalles, por ejemplo, cual auto usar: si el proporcionado o el que texturizamos nosotros, o como debían encenderse las luces de la lampara y el modelo elegido: si al mismo tiempo usando las mismas teclas para ambos elementos, o por separado, usando teclas distintas para cada uno.

**c. Conclusión**

Al finalizar esta práctica aprendí a trasladar luces en tiempo de ejecución, a colocarlas en lugares específicos para una mejor apreciación e iluminación de objetos, y aprendí a crear luces de tipo Point light y Spot light.

**Bibliografía**

- animatedheaven. (4 de septiembre de 2016). *Driveway Gate Old modelo 3d*. Free3D. <https://www.turbosquid.com/es/3d-models/3d-model-a-minimal-looking-reading-lamp-2153391>
- Earthless188. (25 de diciembre de 2022). *Low Poly Hamburger Free low-poly 3D model*. Cgtrader. <https://www.cgtrader.com/free-3d-models/food/miscellaneous/low-poly-hamburger-783c10e5-497f-4468-bdbd-ee48c1f913a6>