



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° 07**

**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 3 de abril de 2024

**CALIFICACIÓN:** \_\_\_\_\_

## REPORTE DE PRÁCTICA:

### 1.1.- Agregar movimiento con teclado al helicóptero hacia adelante y atrás.

Primero que nada, el helicóptero y su luz comparten las mismas teclas para moverse, de esta forma se mueven al mismo tiempo y en el mismo sentido. Para esto, en el archivo window.cpp fue necesario agregar la siguiente modificación:

```
if (key == GLFW_KEY_F)
{
    theWindow->articulacion2 -= 0.2;
}

if (key == GLFW_KEY_G)
{
    theWindow->articulacion2 += 0.2;
}
```

Con la tecla F y la tecla G se activa la misma articulación, pero una agrega valores y la otra resta valores.

El helicóptero se mueve gracias a la siguiente sentencia:

```
model = glm::translate(model, glm::vec3(0.0, mainWindow.getarticulacion2()*3.35, 0.0f));
```

El valor que se resta o suma a articulación 2 al presionar una tecla es multiplicado por 3.35. Este número es necesario para que el helicóptero avance o retroceda a la misma velocidad que la luz que posee.

### 1.2.- Crear una luz spotlight para el helicóptero de color amarilla que apunte hacia el piso y se mueva con el helicóptero

La luz del helicóptero fue creada de la siguiente manera:

```
//Luz del helicoptero
spotLights[2] = SpotLight(1.0f, 1.0f, 0.0f, //rojo, verde y azul
    1.0f, 2.0f, //ambiental y difuso
    0.0f, 5.5f, 6.0f, //pocision
    0.0f, -5.0f, 0.0f, //direccion
    1.0f, 0.0f, 0.0f, //valores de la ecuacion cuadratica
    15.0f); //Angulo de apertura del cono del spotlight
spotLightCount++;
```

Se modifica su color para mostrarse de color amarillo, y se agrega una posición en la que quede justo en la parte baja del helicóptero, como si la luz fuera disparada desde este vehículo.

Para que se mueva junto a él, fue necesario crear un vector de 3 dimensiones que represente su posición:

```
glm::vec3 luzheli = glm::vec3(-mainWindow.getarticulacion2(), 5.5f, 6.0f);
```

El vector luzheli guarda la posición que definimos en la declaración de la luz, y también permite actualizar la posición en el eje X de la luz en función de la tecla que se presione y que este ligada a la articulación 2. No es necesario multiplicar el valor obtenido en la articulación. Para que la luz actualice su posición en tiempo de ejecución, es necesario escribir esta última línea de código:

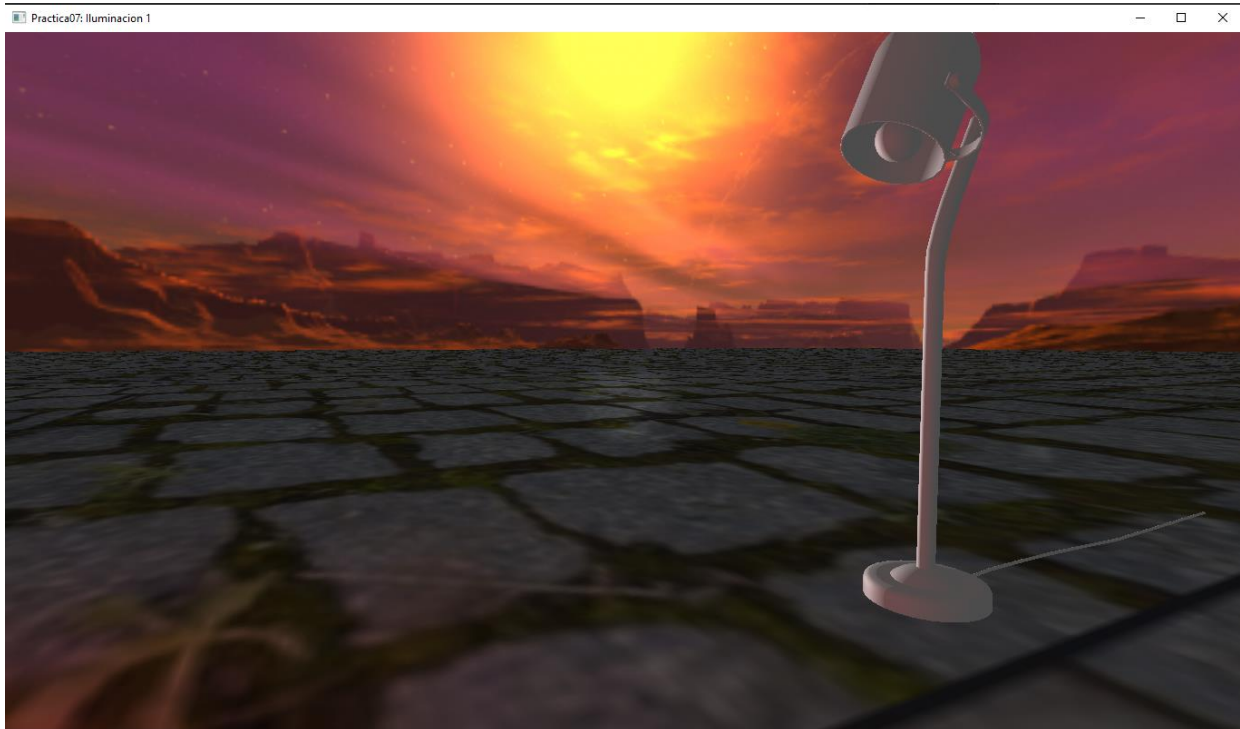
```
spotLights[2].SetPos(luzheli);
```

La función SetPos actualiza la posición de la luz de tipo spotlight, y toma los valores del vector definido anteriormente para definir su nueva posición.

El helicóptero con su luz moviéndose se pueden ver en el video adjuntado.

### 1.3.- Importar 1 modelo de lámpara y crearle luz puntual blanca.

El modelo importado está en la bibliografía de este documento. En la siguiente captura se observa el modelo importado sin una luz



La luz creada para este modelo es la siguiente:

```
//Luz de la lampara
pointLights[1] = PointLight(1.0f, 1.0f, 1.0f, //rojo, verde y azul
    0.0f, 1.0f, //ambiental y difuso
    -14.0f, 4.0f, 0.0f, //pocision
    1.0f, 1.0f, 1.0f); //valores de la ecuacion cuadratica
pointLightCount++;
```

Con la luz creada y colocada en la posición del foco de la lampara, obtenemos la siguiente vista del modelo:



**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**

Esta vez no hubo problemas, conseguí averiguar como mover la luz tanto adelante como hacia atrás y solo utilizando un vector, en vez de 2 como en el ejercicio pasado. El modelo ya estaba en formato .obj así que no hacia falta convertirlo en 3dsMax, y las luces para ambos modelos no fueron un problema de crear.

**3.- Conclusión:**

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

Fue un ejercicio relativamente sencillo, entender como funciona la luz puede tomar un poco de tiempo, pero cuando se logra entender las actividades son realmente muy rápidas.

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

En realidad entender como funciona las luces creadas no es tan complicado, salvo claro, la sección de la ecuación cuadrática. Me parece que el material

proporcionado y la explicación durante la última clase si que fueron suficientes para poder terminar las actividades solicitadas.

### **c. Conclusión**

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

### **Bibliografía**

- sougatadhar16. **UNA LÁMPARA DE LECTURA DE ASPECTO MÍNIMO**. Recuperado el 1 de abril de 2024 de: <https://www.turbosquid.com/es/3d-models/3d-model-a-minimal-looking-reading-lamp-2153391>