



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° 05

NOMBRE COMPLETO: Carlos Alberto Arroyo Ramírez

N° de Cuenta: 320185865

GRUPO DE LABORATORIO: 03

GRUPO DE TEORÍA: 04

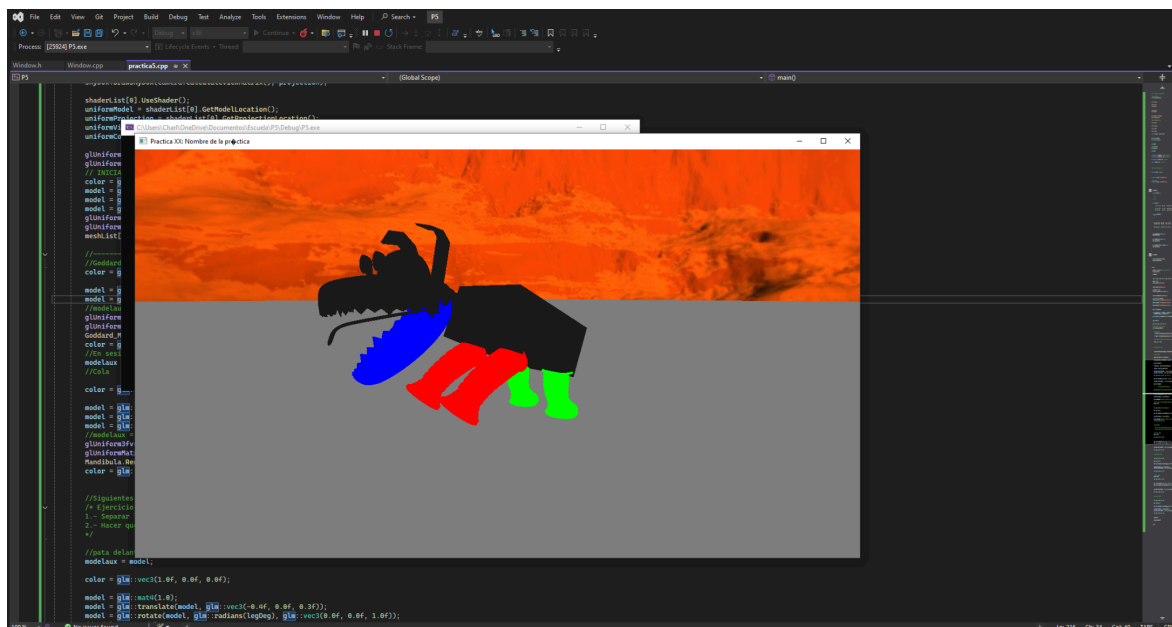
SEMESTRE 2026-1

FECHA DE ENTREGA LÍMITE: 23/09/2025

CALIFICACIÓN: _____

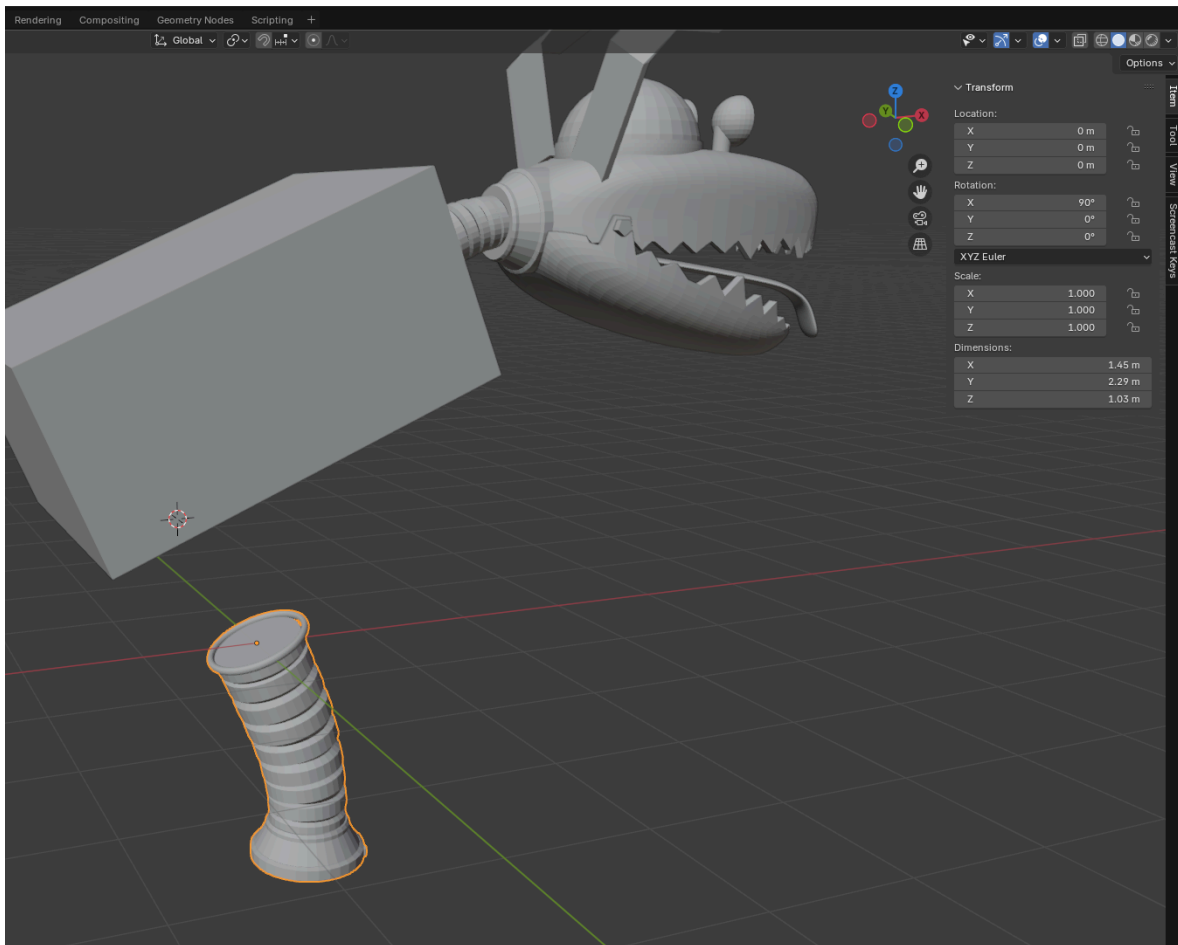
EJERCICIOS DE SESIÓN:

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



En esta actividad exporte en diferentes formatos el modelo 3D de Goddard, para comenzar separe por partes el modelo 3D, se conformó de la mandíbula, las patas por separado, la cola y el cuerpo.

En Blender antes de exportar los modelos 3D por separado, tuve que mandar el centro de origen de cada modelo al punto donde pasa el eje que sirve como referencia para el giro de cada una de las partes.



Una vez exportado el modelo 3d, en el código de mi programa realice los siguientes pasos:

```
Camera camera;
Model Goddard_M;
Model Mandibula;
Model PatasDelanteras;
Model PatasTraseras;
```

Declarar modelos para importar.

```
// Rangos (grados)
static constexpr float MANDIBULA_MIN = -15.0f; // Mandíbula -15°
static constexpr float MANDIBULA_MAX = 30.0f; // Mandíbula +30°
static constexpr float PATA_MIN = -45.0f; // Pata -45°
static constexpr float PATA_MAX = 45.0f; // Pata +45°
```

Establecer variables con los ángulos limite de giro para cada parte.

```
Goddard_M = Model();
Goddard_M.LoadModel("Models/Cuerpo.obj");

Mandibula = Model();
Mandibula.LoadModel("Models/Mandibula.obj");

PatasDelanteras = Model();
PatasDelanteras.LoadModel("Models/FrenteDerecha.obj");

PatasTraseras = Model();
PatasTraseras.LoadModel("Models/AtrasDerecha.obj");
```

Importar mis

modelos obj almacenados en mi carpeta Models.

```
//Limitando angulos:
float mandDeg = glm::clamp(mainWindow.getarticulacion1(), MANDIBULA_MIN, MANDIBULA_MAX);
float legDeg = glm::clamp(mainWindow.getarticulacion2(), PATA_MIN, PATA_MAX);
float legDeg2 = glm::clamp(mainWindow.getarticulacion3(), PATA_MIN, PATA_MAX);
```

Un paso adicional para asignar a una variable los límites de giro y la articulación correspondiente, es decir la tecla con la que se iban a activar.

```
//pata trasera derecha
modelaux = model;

color = glm::vec3(0.0f, 1.0f, 0.0f);

model = glm::mat4(1.0);
model = glm::translate(model, glm::vec3(1.4f, -0.65f, 0.3f));
model = glm::rotate(model, glm::radians(legDeg2), glm::vec3(0.0f, 0.0f, 1.0f));
//modelaux = model;
glUniform3fv(uniformColor, 1, glm::value_ptr(color));
glUniformMatrix4fv(uniformModel, 1, GL_FALSE, glm::value_ptr(model));
PatasTraseras.RenderModel();
color = glm::vec3(0.0f, 0.0f, 1.0f);
```

Definir color, coordenadas, rotación y todo lo necesario para dibujar cada parte del robot, en la rotación, solo fue necesario llamar a la variable que defini en el código de arriba.

También modifique un poco Window.cpp para el tema de las articulaciones, algunas tomaron valores negativos para hacer un giro contrario:

```

if (key == GLFW_KEY_T)
{
    theWindow->rotaz += 10.0;
}
if (key == GLFW_KEY_F)
{
    theWindow->articulacion1 += 10.0;
}

if (key == GLFW_KEY_G)
{
    theWindow->articulacion1 -= 10.0;
}
if (key == GLFW_KEY_H)
{
    theWindow->articulacion2 += 10.0;
}
if (key == GLFW_KEY_J)
{
    theWindow->articulacion2 -= 10.0;
}
if (key == GLFW_KEY_K)

```

2. Problemas presentados. Listar si surgieron problemas a la hora de ejecutar el código

Jugar un poco con las coordenadas para acomodar el modelo 3D, y aplicar la rotación, me pasaba que la rotación la hacía completa, es decir solo hacia los extremos, lo solucione cambiando un poco esta seccion del codigo:

```

//Lo modifique para que los giros los diera naturales y no completos
GLfloat now = static_cast<GLfloat>(glfwGetTime());
deltaTime = now - lastTime;
lastTime = now;

```

También mi código no es perfecto, al usar un acumulador, cuando mantengo presionado una tecla de giro por mucho más tiempo, ya después de llegar al límite, sigue contando, así que si presiono la tecla contraria a la del giro contrario puede que tarde un poco en responder ya que resta al acumulador.

3. Conclusión:

- a. Los ejercicios de la clase: Complejidad, explicación
- b. Comentarios generales: Faltó explicar a detalle, ir más lento en alguna explicación, otros comentarios y sugerencias.

Lo que se me complicó fue colocar bien el pivote para que la rotación fuera natural. También me costó exportar correctamente las piezas para que, al importar, no quedaran mal, y separar los controles para que la mandíbula y las patas no se estorbaran. Al final aprendí a importar y exportar bien y, sobre todo, a limitar ángulos con clamp para que el movimiento se vea fluido y con topes claros.

Como sugerencia, en clase no da el tiempo para completar todo el flujo, así que esta actividad funciona mejor como práctica en casa.