



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° 05

NOMBRE COMPLETO: Sanchez Villalpando Johan

N° de Cuenta: 422028657

GRUPO DE LABORATORIO: 04

GRUPO DE TEORÍA: 02

SEMESTRE 2025-2

FECHA DE ENTREGA LÍMITE: 22/03/25

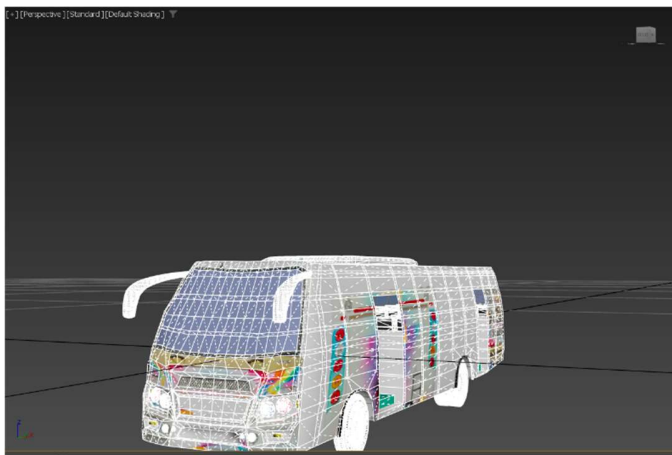
CALIFICACIÓN: _____

REPORTE DE PRÁCTICA:

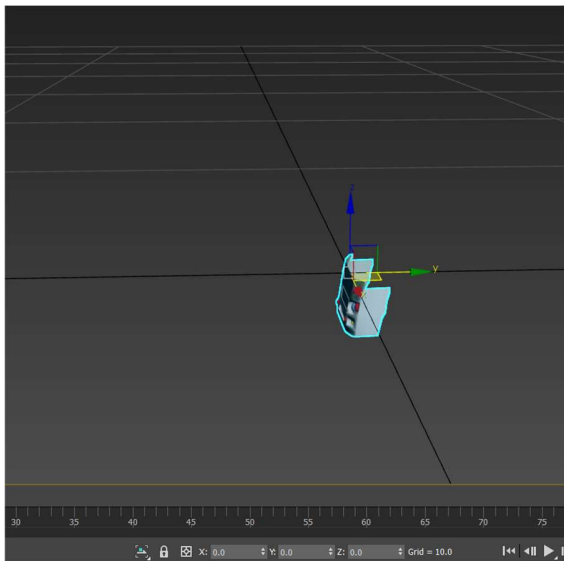
1.- Ejecución de los ejercicios que se dejaron, comentar cada uno y capturas de pantalla de bloques de código generados y de ejecución del programa.

Para esta práctica el ejercicio consiste en usar nuestro modelo de carro del previo, separar las llantas y el capó para exportarlos como objetos aparte y luego por código unirlos, darle movimiento a las llantas, al capó y al carro completo, de modo que las llantas avancen,etrocedan y el auto avance yetroceda.

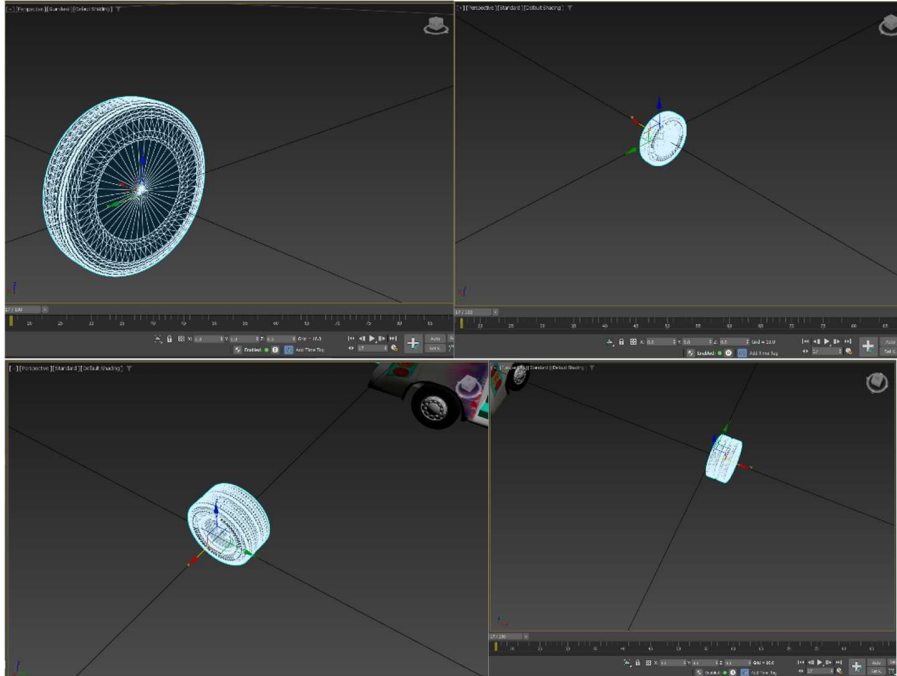
Primero se toma el modelo del vehiculo y se trianguliza en 3ds max, una vez hecho esto noté que en mi modelo de carro el capó no existia como objeto aparte, por lo que tuve que tomarlo del cuerpo del bus y separarlo seleccionando esa sección.



Al capó se le ajusta el pivote en la esquina para rotar desde ahí y se manda al centro.



Para el caso de las llantas ya existe un objeto separado para cada una, por lo que fue más sencillo, en el caso de la llanta el pivote se puso en el centro y del lado interno de la llanta.



Luego estos modelos se declaran en el código:

```
//modelos práctica
Model cuerpo_bus;
Model capo_bus;
Model llanta_del_izq;
Model llanta_del_der;
Model llanta_tras_izq;
Model llanta_tras_der;
```

```
//IMPORTADOS PARA EJERCICIO DE PRÁCTICA
cuerpo_bus = Model(); //Creamos modelo
cuerpo_bus.LoadModel("Models/cuerpo_bus.obj");
capo_bus = Model(); //Creamos modelo
capo_bus.LoadModel("Models/capo_bus.obj");
llanta_del_izq = Model(); //Creamos modelo
llanta_del_izq.LoadModel("Models/llanta_del_izq.obj");
llanta_del_der = Model(); //Creamos modelo
llanta_del_der.LoadModel("Models/llanta_del_der.obj");
llanta_tras_der = Model(); //Creamos modelo
llanta_tras_der.LoadModel("Models/llanta_tras_der.obj");
llanta_tras_izq = Model(); //Creamos modelo
llanta_tras_izq.LoadModel("Models/llanta_tras_izq.obj");
```

Despues del piso inicia el dibujado del carro.

Se dibuja el cuerpo del bus, luego se dibuja el capó, debido a que no se hacen cambios en escala en el bus, no es necesario usar denuevo la variable auxiliar para el capó, sin embargo para las llantas si lo es, ya que es necesario partir de la posición del cuerpo del bus.

```
//INICIA EJERCICIO DE PRÁCTICA
//Bus
color = glm::vec3(0.0f, 0.0f, 1.0f);
model = glm::mat4(1.0);
model = glm::translate(model, glm::vec3(0.0f, -0.8f, -1.5f));
//Movimiento usando K para adelante y L para atras, además llantas se mueven al avanzar o retroceder.
model = glm::translate(model, glm::vec3(0.0f, 0.0f, mainWindow.getarticulacion4()));
modelaux = model; //se guarda traslación
glUniform3fv(uniformColor, 1, glm::value_ptr(color));
glUniformMatrix4fv(uniformModel, 1, GL_FALSE, glm::value_ptr(model));
cuerpo_bus.RenderModel();//crear cuerpo de bus
//capo
color = glm::vec3(0.0f, 0.0f, 0.5f);
model = glm::translate(model, glm::vec3(0.0f, 0.0f, 3.4f));
//articulación para el capo
model = glm::rotate(model, glm::radians(mainWindow.getarticulacion1()), glm::vec3(-1.0f, 0.0f, 0.0f));
glUniform3fv(uniformColor, 1, glm::value_ptr(color));
glUniformMatrix4fv(uniformModel, 1, GL_FALSE, glm::value_ptr(model));
capo_bus.RenderModel();//capo
model = modelaux; //volvemos a cuerpo
```

Para las llantas se asigna la misma variable articulación debido a que se requiere que todas avancen o retrocedan a la vez.

```

//llanta izq del
color = glm::vec3(0.5f, 0.0f, 0.0f);
model = glm::translate(model, glm::vec3(0.7f, -0.8f, 2.1f));
//Se usa misma variable para rotar todas las llantas
//H hacia adelante y J hacia atrás
model = glm::rotate(model, glm::radians(mainWindow.getarticulacion3()), glm::vec3(1.0f, 0.0f, 0.0f));
glUniform3fv(uniformColor, 1, glm::value_ptr(color));
glUniformMatrix4fv(uniformModel, 1, GL_FALSE, glm::value_ptr(model));
llanta_del_izq.RenderModel(); //llanta adelatne izq
model = modelaux; //volvemos a cuerpo
//llanta izq del
color = glm::vec3(0.5f, 0.0f, 0.0f);
model = glm::translate(model, glm::vec3(-0.7f, -0.8f, 2.1f));
//H hacia adelante y J hacia atrás
model = glm::rotate(model, glm::radians(mainWindow.getarticulacion3()), glm::vec3(1.0f, 0.0f, 0.0f));
glUniform3fv(uniformColor, 1, glm::value_ptr(color));
glUniformMatrix4fv(uniformModel, 1, GL_FALSE, glm::value_ptr(model));
llanta_del_der.RenderModel(); //llanta adelatne der
model = modelaux; //volvemos a cuerpo

```

```

//llanta izq tras
color = glm::vec3(0.5f, 0.0f, 0.0f);
model = glm::translate(model, glm::vec3(0.5f, -0.8f, -1.55f));
//H hacia adelante y J hacia atrás
model = glm::rotate(model, glm::radians(mainWindow.getarticulacion3()), glm::vec3(1.0f, 0.0f, 0.0f));
glUniform3fv(uniformColor, 1, glm::value_ptr(color));
glUniformMatrix4fv(uniformModel, 1, GL_FALSE, glm::value_ptr(model));
llanta_tras_izq.RenderModel(); //llanta adelatne der
model = modelaux; //volvemos a cuerpo
//llanta der tras
color = glm::vec3(0.5f, 0.0f, 0.0f);
//H hacia adelante y J hacia atrás
model = glm::translate(model, glm::vec3(-0.5f, -0.8f, -1.55f));
model = glm::rotate(model, glm::radians(mainWindow.getarticulacion3()), glm::vec3(1.0f, 0.0f, 0.0f));
glUniform3fv(uniformColor, 1, glm::value_ptr(color));
glUniformMatrix4fv(uniformModel, 1, GL_FALSE, glm::value_ptr(model));
llanta_tras_der.RenderModel(); //llanta adelatne der

```

Para que el bus se mueva hacia adelante y atrás, se utiliza la variable de articulación4, la cual aumenta o disminuye con k y l.

```

//Movimiento usando K para adelante y L para atras, ademas llantas se mueven al avanzar o retroceder.
model = glm::translate(model, glm::vec3(0.0f, 0.0f, mainWindow.getarticulacion4()));

```

```

}
if (key == GLFW_KEY_K)
{
    theWindow->articulacion4 += 0.2;
    theWindow->articulacion3 += 10.0; //llantas giran cuando avanza
    /* ejercicio
    if (theWindow->articulacion4 <= 45) {
        theWindow->articulacion4 += 10.0;
    }
    */
}
if (key == GLFW_KEY_L)
{
    theWindow->articulacion4 -= 0.2;
    theWindow->articulacion3 -= 10.0; //llantas giran cuando avanza
    /* EJERCICIO
    if (theWindow->articulacion4 >= -45) {
        theWindow->articulacion4 -= 10.0;
    }
    */
}
}

```

Adicionalmente se añade también cambios en la articulación3, ya que esta se encarga del movimiento de las llantas, haciendo que al avanzar o retroceder giren las llantas en su respectivo sentido.

Para el capó se usa la articulación1, dando límites de abrir máximo a 70° y mínimo cerrar a -10° usando las teclas f y g.

```
if (key == GLFW_KEY_F)
{
    /*Ejercicio
    theWindow->articulacion1 += 10.0;
    */
    if (theWindow->articulacion1 <= 70) {
        theWindow->articulacion1 += 10.0;
    }
}

if (key == GLFW_KEY_G)
{
    /*Ejercicio
    theWindow->articulacion2 += 10.0;
    */
    if (theWindow->articulacion1 >= 10) {
        theWindow->articulacion1 -= 10.0;
    }
}
```

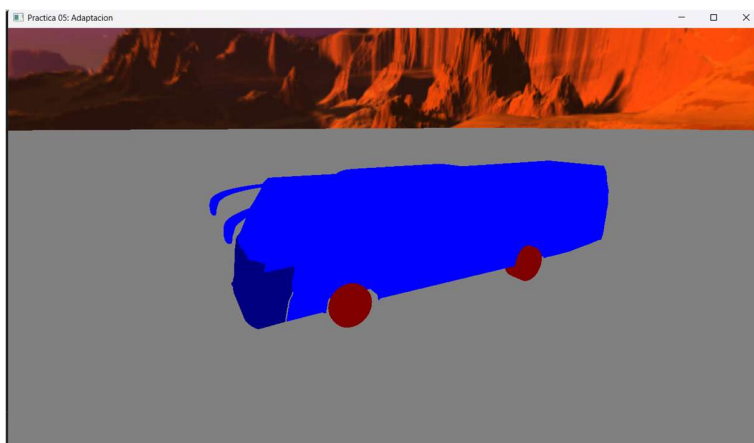
Para las llantas se usa articulación3, la cual avanza con h y retrocede con j

```
if (key == GLFW_KEY_H)
{
    theWindow->articulacion3 += 10.0;
    /* Ejercicio
    if (theWindow->articulacion3 <= 45) {
        theWindow->articulacion3 += 10.0;
    }*/
}

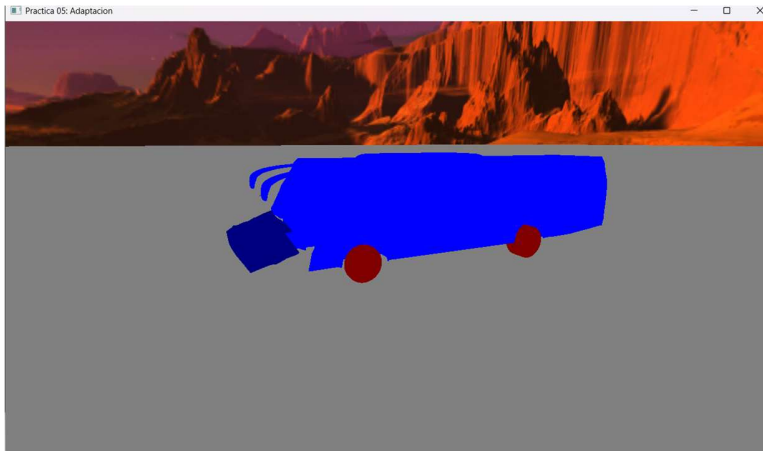
if (key == GLFW_KEY_J)
{
    theWindow->articulacion3 -= 10.0;
    /* ejercicio
    if (theWindow->articulacion3 >= -45) {
        theWindow->articulacion3 -= 10.0;
    }*/
}
```

Finalmente tenemos el resultado:

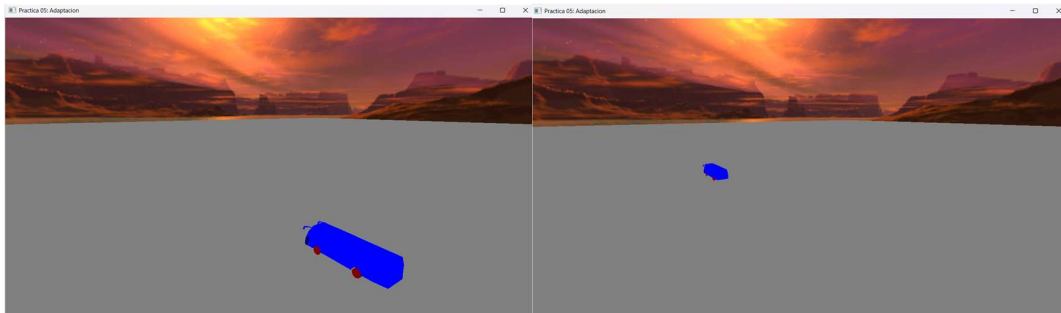
Modelo



Capó



Movimiento del autobus alejandose.



Para el movimiento de las llantas no es posible apreciarlo con capturas.

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.

Hubo algunos problemas debido a que no entendí bien las especificaciones, como si las llantas tenían que girar todas a la vez o cada llanta tener sus teclas independientes, fuera de eso todo fue bastante rápido, también hubo detalles con el modelo de carro que elegí ya que el capó no lo tenía separado, por lo que tuve que seleccionar esa sección y generar el capó por mi cuenta, en cuanto a lo demás el modelo resultó buena elección ya que tenía las llantas separadas y el tamaño era adecuado.

3.- Conclusión:

- Los ejercicios del reporte: Complejidad, Explicación.
- Comentarios generales: Faltó explicar a detalle, ir más lento en alguna explicación, otros comentarios y sugerencias para mejorar desarrollo de la práctica

Esta práctica fue sencilla debido a que se retomó lo usado en las prácticas anteriores, para generar un modelo y pieza por pieza acomodarlos y darles movimiento.

Fue entretenido ya que empezamos a trabajar con modelos ya hechos tanto en el ejercicio como en la práctica, considero que esto abre las puertas para cuando nos toque trabajar con nuestro personaje del proyecto final, además de los juegos o demás modelos de los que se requiera tener movimiento, el usar teclas para manejar los modelos en el código es interesante ya que también añadimos traslación, teniendo un auto que además de abrir y cerrar capó también avanza y retrocede incluso con movimiento de las ruedas.

Bibliografía en formato APA

- Rbj, P. P. (s/f). *TN BUS Modelo 3D gratis - .blend .obj .3ds .fbx - Free3D*. Free3d.com. Recuperado el 01 de marzo de 2025, de <https://free3d.com/es/modelo-3d/tn-bus-842000.html>