



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: Miranda González José Francisco

N° de Cuenta: 318222327

GRUPO DE LABORATORIO: 03

GRUPO DE TEORÍA: 04

SEMESTRE 2025-1

FECHA DE ENTREGA LÍMITE: 21/09/24

CALIFICACIÓN: _____

REPORTE DE PRÁCTICA:

Reporte de práctica 5: Optimización y Carga de Modelos

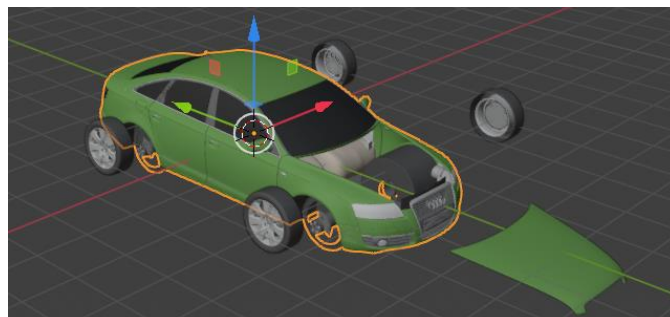
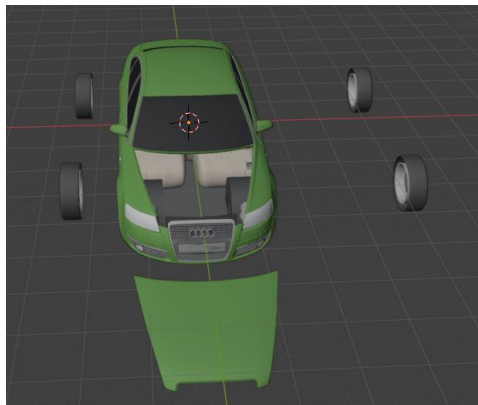
Instrucciones:

- 1.- Importar su modelo de coche propio dentro del escenario a una escala adecuada.
- 2.- Importar sus 4 llantas y acomodarlas jerárquicamente, agregar el mismo valor de rotación a las llantas para que al presionar puedan rotar hacia adelante y hacia atrás.
- 3.- Importar el cofre del coche, acomodarlo jerárquicamente y agregar la rotación para poder abrir y cerrar.
- 4.- Agregar traslación con teclado para que pueda avanzar y retroceder de forma independiente

Actividades realizadas

Para completar las actividades solicitadas realice lo siguiente:

En Blender 4.2 importe el modelo del coche, separe cada parte solicitada, modifique los pivotes y exporte por separado desde el origen.



En mian.cpp realice los siguientes cambios:

Crear los modelos:

```
38      // MODELOS PARA LA PRACTICA 05
39
40      //////////////////////////////////////
41      Model coche;
42      Model cofre_coche;
43      Model llanta_trasera_izquierda_coche;
44      Model llanta_trasera_derecha_coche;
45      Model llanta_delantera_izquierda_coche;
46      Model llanta_delantera_derecha_coche;
47      //////////////////////////////////////
```

Cargar los modelos:

```
127      // MODELOS PARA LA PRACTICA 05
128
129      //////////////////////////////////////
130      coche = Model();
131      coche.LoadModel("Models/coche.obj");
132      cofre_coche = Model();
133      cofre_coche.LoadModel("Models/cofre_coche.obj");
134      llanta_trasera_izquierda_coche = Model();
135      llanta_trasera_izquierda_coche.LoadModel("Models/llanta_trasera_izquierda_coche.obj");
136      llanta_trasera_derecha_coche = Model();
137      llanta_trasera_derecha_coche.LoadModel("Models/llanta_trasera_derecha_coche.obj");
138      llanta_delantera_izquierda_coche = Model();
139      llanta_delantera_izquierda_coche.LoadModel("Models/llanta_delantera_izquierda_coche.obj");
140      llanta_delantera_derecha_coche = Model();
141      llanta_delantera_derecha_coche.LoadModel("Models/llanta_delantera_derecha_coche.obj");
142      //////////////////////////////////////
```

Matriz auxiliar para el centro del coche:

```
162      // COCHE
163      // MATRIZ AUX
164      //////////////////////////////////////
165      glm::mat4 modelaux(1.0);
166      //////////////////////////////////////
```

Para el dibujo del coche:

```
206      //-----*INICIA DIBUJO DE NUESTROS DEMÁS OBJETOS-----*
207
208      model = glm::mat4(1.0);
209
210      // COCHE
211
212      model = glm::translate(model, glm::vec3(0.0f, 3.5f, -3.0f));
213      model = glm::translate(model, glm::vec3(0.0f, 0.0f, mainWindow.gettraslacion())); // Trasladar con C y V
214
215      modelaux = model;
216
217      model = glm::scale(model, glm::vec3(0.5f, 0.5f, 0.5f));
218      glUniformMatrix4fv(uniformModel, 1, GL_FALSE, glm::value_ptr(model));
219      color = glm::vec3(1.0f, 1.0f, 0.0f);
220      glUniform3fv(uniformColor, 1, glm::value_ptr(color));
221      coche.RenderModel();
```

En donde vamos a ocupar la traslación por teclado (C y V) sobre el eje Z con `mainWindow.gettraslacion()`.

Para el cofre:

```
223     model = modelaux; // A PARTIR DEL COCHE
224
225     // COFRE
226
227     model = glm::translate(model, glm::vec3(0.0f, 3.0f, 12.5f));
228     model = glm::scale(model, glm::vec3(0.5f, 0.5f, 0.5f));
229     model = glm::rotate(model, glm::radians(mainWindow.getarticulacionCofre()), glm::vec3(1.0f, 0.0f, 0.0f)); // Rotar con: Y y U
230     glUniformMatrix4fv(uniformModel, 1, GL_FALSE, glm::value_ptr(model));
231     color = glm::vec3(0.0f, 1.0f, 0.0f);
232     glUniform3fv(uniformColor, 1, glm::value_ptr(color));
233     cofre_cocher.RenderModel();
```

En donde vamos a ocupar rotación por teclado (Y y U) sobre el eje X para abrir y cerrar el cofre.

Para la llanta trasera izquierda:

```
235     model = modelaux; // A PARTIR DEL COCHE
236
237     // LLANTA TRASERA IZQUIERDA
238
239     model = glm::translate(model, glm::vec3(6.28f, -2.46f, -9.0f));
240     model = glm::scale(model, glm::vec3(0.5f, 0.5f, 0.5f));
241     model = glm::rotate(model, glm::radians(mainWindow.getarticulacionLlantaTraseraIzquierda()), glm::vec3(1.0f, 0.0f, 0.0f));
242     glUniformMatrix4fv(uniformModel, 1, GL_FALSE, glm::value_ptr(model));
243     color = glm::vec3(1.0f, 0.0f, 0.0f);
244     glUniform3fv(uniformColor, 1, glm::value_ptr(color));
245     llanta_trasera_izquierda_cocher.RenderModel();
```

En donde vamos a ocupar rotación sobre el eje X si el coche avanza o retrocede.

Para la llanta trasera derecha:

```
247     model = modelaux; // A PARTIR DEL COCHE
248
249     // LLANTA TRASERA DERECHA
250
251     model = glm::translate(model, glm::vec3(-6.5f, -2.46f, -9.0f));
252     model = glm::scale(model, glm::vec3(0.5f, 0.5f, 0.5f));
253     model = glm::rotate(model, glm::radians(mainWindow.getarticulacionLlantaTraseraDerecha()), glm::vec3(1.0f, 0.0f, 0.0f));
254     glUniformMatrix4fv(uniformModel, 1, GL_FALSE, glm::value_ptr(model));
255     color = glm::vec3(0.0f, 1.0f, 1.0f);
256     glUniform3fv(uniformColor, 1, glm::value_ptr(color));
257     llanta_trasera_derecha_cocher.RenderModel();
```

En donde vamos a ocupar rotación sobre el eje X si el coche avanza o retrocede.

Para la llanta delantera izquierda:

```
259     model = modelaux; // A PARTIR DEL COCHE
260
261     // LLANTA DELANTERA IZQUIERDA
262
263     model = glm::translate(model, glm::vec3(6.2f, -2.46f, 14.7f));
264     model = glm::scale(model, glm::vec3(0.5f, 0.5f, 0.5f));
265     model = glm::rotate(model, glm::radians(mainWindow.getarticulacionLlantaDelanteraIzquierda()), glm::vec3(1.0f, 0.0f, 0.0f));
266     glUniformMatrix4fv(uniformModel, 1, GL_FALSE, glm::value_ptr(model));
267     color = glm::vec3(1.0f, 0.0f, 1.0f);
268     glUniform3fv(uniformColor, 1, glm::value_ptr(color));
269     llanta_delantera_izquierda_cocher.RenderModel();
```

En donde vamos a ocupar rotación sobre el eje X si el coche avanza o retrocede.

Para la llanta delantera derecha:

```
271 |     model = modelaux; // A PARTIR DEL COCHE
272 |
273 |     // LLANTA DELANTERA DERECHA
274 |
275 |     model = glm::translate(model, glm::vec3(-6.5f, -2.46f, 14.7f));
276 |     model = glm::scale(model, glm::vec3(0.5f, 0.5f, 0.5f));
277 |     model = glm::rotate(model, glm::radians(mainWindow.getarticulacionLlantaDelanteraDerecha()), glm::vec3(1.0f, 0.0f, 0.0f));
278 |     glUniformMatrix4fv(uniformModel, 1, GL_FALSE, glm::value_ptr(model));
279 |     color = glm::vec3(0.42f, 0.0f, 0.18f);
280 |     glUniform3fv(uniformColor, 1, glm::value_ptr(color));
281 |     llanta_delantera_derecha_coche.RenderModel();
```

En donde vamos a ocupar rotación sobre el eje X si el coche avanza o retrocede.

En window.cpp realice los siguientes cambios:

Articulaciones para el coche:

```
24 | // ARTICULACIONES COCHE
25 | ///////////////////////////////////////////////////////////////////
26 | articulacionCofre = 0.0f;
27 | articulacionLlantaTraseraIzquierda = 0.0f;
28 | articulacionLlantaTraseraDerecha = 0.0f;
29 | articulacionLlantaDelanteraIzquierda = 0.0f;
30 | articulacionLlantaDelanteraDerecha = 0.0f;
31 | ///////////////////////////////////////////////////////////////////
```

Traslación para el coche:

```
33 | // TRASLACION COCHE
34 | ///////////////////////////////////////////////////////////////////
35 | traslacion = 0.0f;
36 | ///////////////////////////////////////////////////////////////////
```

Nombre de la practica en la ventana:

```
59 | //CREAR VENTANA
60 | mainWindow = glfwCreateWindow(width, height, "Practica 5: Optimizacion y Carga de Modelos", NULL, NULL);
```

Para la articulación del cofre:

```
157 | // PARA EL COCHE (ARTICULACION)
158 | ///////////////////////////////////////////////////////////////////
159 | // COFRE
160 | if (key == GLFW_KEY_Y)
161 | {
162 |     if (theWindow->articulacionCofre >= 0.0)
163 |     {
164 |     }
165 |     else
166 |     {
167 |         theWindow->articulacionCofre += 10.0;
168 |     }
169 | }
```

```

170     if (key == GLFW_KEY_U)
171     {
172         if (theWindow->articulacionCofre <= -60.0)
173         {
174         }
175         else
176         {
177             theWindow->articulacionCofre -= 10.0;
178         }
179     }
180     //////////////////////////////////////

```

Limitar la rotación del cofre al abrir y cerrar.

Para la traslación:

```

183     // PARA EL COCHE (TRASLACION)
184     //////////////////////////////////////
185     if (key == GLFW_KEY_C)
186     {
187         if (theWindow->traslacion >= 100.0)
188         {
189         }
190         else
191         {
192             // PARA HACER LA TRASLACION
193             theWindow->traslacion += 1.0;
194
195             // PARA HACER LA ROTACION
196             // LLANTA TRASERA IZQUIERDA
197             theWindow->articulacionLlantaTraseraIzquierda += 10.0;
198             // LLANTA TRASERA DERECHA
199             theWindow->articulacionLlantaTraseraDerecha += 10.0;
200             // LLANTA DELANTERA IZQUIERDA
201             theWindow->articulacionLlantaDelanteraIzquierda += 10.0;
202             // LLANTA DELANTERA DERECHA
203             theWindow->articulacionLlantaDelanteraDerecha += 10.0;
204         }
205     }
206     if (key == GLFW_KEY_V)
207     {
208         if (theWindow->traslacion <= -100.0)
209         {
210         }
211         else
212         {
213             // PARA HACER LA TRASLACION
214             theWindow->traslacion -= 1.0;
215
216             // PARA HACER LA ROTACION
217             // LLANTA TRASERA IZQUIERDA
218             theWindow->articulacionLlantaTraseraIzquierda -= 10.0;

```

```

219 // LLANTA TRASERA DERECHA
220 theWindow->articulacionLlantaTraseraDerecha -= 10.0;
221 // LLANTA DELANTERA IZQUIERDA
222 theWindow->articulacionLlantaDelanteraIzquierda -= 10.0;
223 // LLANTA DELANTERA DERECHA
224 theWindow->articulacionLlantaDelanteraDerecha -= 10.0;
225     }
226 }
227 //////////////////////////////////////

```

Limitar la traslación del coche al avanzar y retroceder.

Al avanzar las llantas rotaran hacia adelante en X y al retroceder las llantas rotaran hacia atrás en X.

En window.h realice los siguientes cambios:

Articulaciones del coche:

```

28 // PARA EL COCHE (ARTICULACIONES)
29 //////////////////////////////////////
30 GLfloat getarticulacionCofre() { return articulacionCofre; }
31 GLfloat getarticulacionLlantaTraseraIzquierda() { return articulacionLlantaTraseraIzquierda; }
32 GLfloat getarticulacionLlantaTraseraDerecha() { return articulacionLlantaTraseraDerecha; }
33 GLfloat getarticulacionLlantaDelanteraIzquierda() { return articulacionLlantaDelanteraIzquierda; }
34 GLfloat getarticulacionLlantaDelanteraDerecha() { return articulacionLlantaDelanteraDerecha; }
35 //////////////////////////////////////

```

Traslación del coche:

```

37 // PARA EL COCHE (TRASLACION)
38 //////////////////////////////////////
39 GLfloat gettraslacion() { return traslacion; }
40 //////////////////////////////////////

```

Articulaciones y traslación:

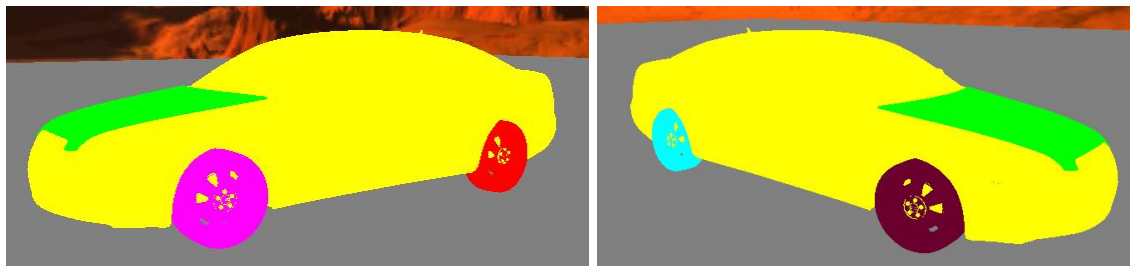
```

49 // PARA EL COCHE (ARTICULACIONES)
50 //////////////////////////////////////
51 GLfloat articulacionCofre, articulacionLlantaTraseraIzquierda, articulacionLlantaTraseraDerecha, articulacionLlantaDelanteraIzquierda,
52 articulacionLlantaDelanteraDerecha;
53 //////////////////////////////////////
54
55 // PARA EL COCHE (TRASLACION)
56 //////////////////////////////////////
57 GLfloat traslacion;
58 //////////////////////////////////////

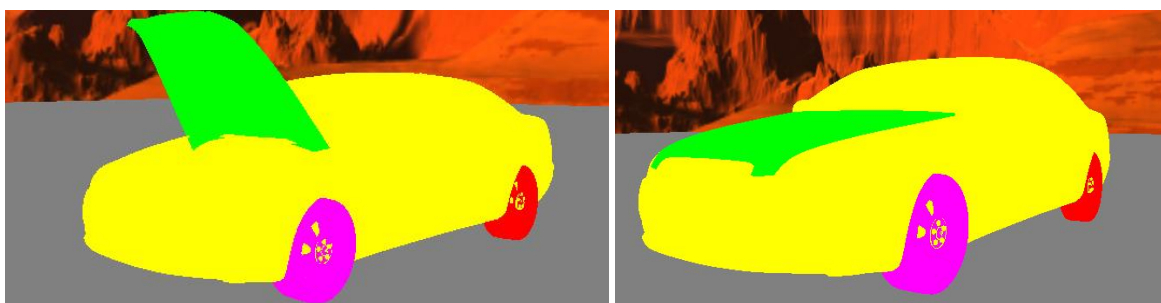
```


Ejecución del programa:

Vista inicial:



Cofre abierto y cerrado:



Avanzar:



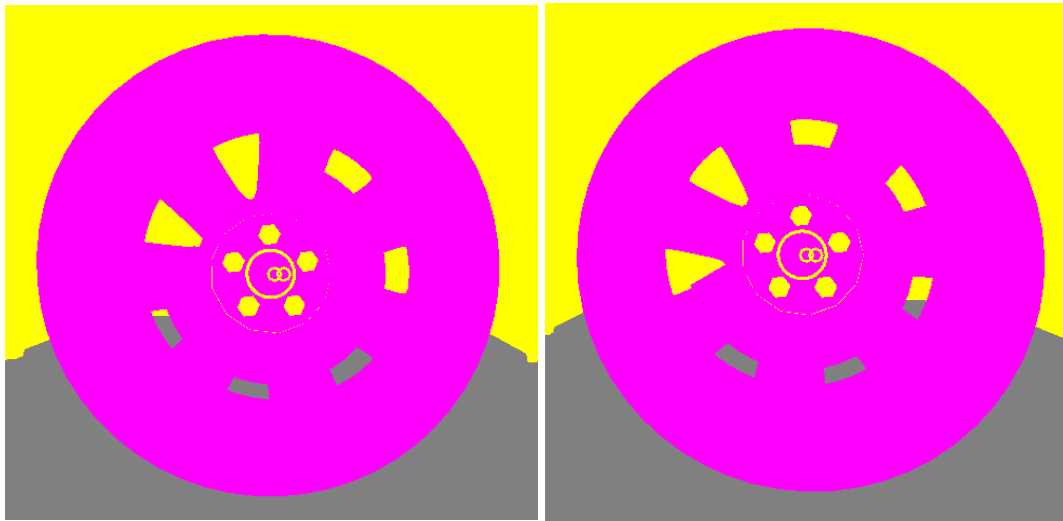
Se puede apreciar observando el escenario de atrás

Retroceder:



Se puede apreciar observando el escenario de atrás

Rotación al avanzar y retroceder:



Problemas presentados

1. Coche
El coche que elegí en un inicio no contaba con un cofre, así que tuve que buscar otro para realizar la práctica.
2. Modelos
Al principio me costo un poco de trabajo exportar los modelos de las llantas, ya que tienen varios detalles y a veces no se exportaba todo lo que quería.
3. Traslación:
No sabía cómo poder realizar la traslación con teclado, pero después de varios intentos logre encontrar la forma modificando el valor de Z en translate.
4. Escala
En Blender 4.2 no escale a un buen tamaño los modelos, pero lo solucione en el main.cpp.

Conclusión

Tuve algunas complicaciones como la escala del coche después de exportarlo, podía corregir eso desde Blender, pero preferí hacer uso de scale dentro del main.cpp

Otro problema fue la traslación por teclado, pero logre implementarlo al cambiar el valor de Z en translate por `mainWindow.gettraslacion()`, además, ahí mismo (traslacion dentro de window.cpp) coloque las rotaciones para cada llanta, así al avanzar cada llanta rotaba dependiendo si estaba avanzando o retrocediendo el coche.

Acomodar las llantas también fue algo tedioso, pues tenían que quedar en el lugar exacto y eso me llevo tiempo.

Al finalizar la practica pienso que logre completar las actividades solicitadas de manera correcta, ya que lo indicado en las instrucciones se puede observar al ejecutar el programa. .

Bibliografía

Open3DModel.com. (última actualización: 13 de octubre del 2019). Modelo 6D Audi A3 Sedán blanco. Consultado el 21 de septiembre del 2024 de https://open3dmodel.com/es/3d-models/white-audi-a6-sedan_32406.html