

# MODELO

Luis Alberto Mejía Troya  
DNI: 01889311E  
luisalbertomt@correo.ugr.es  
Profesor: Luis López Escudero  
Grupo: 3A  
Fecha: 3-01-2025



# UNIVERSIDAD DE GRANADA

# Índice general

<b>1. Introducción</b>	<b>1</b>
<b>2. Transformaciones</b>	<b>3</b>
2.1. Función . . . . .	3
<b>3. Grafos de escena</b>	<b>4</b>
3.1. Grafo general . . . . .	4
3.2. Especificación por partes . . . . .	4
3.2.1. Rueda . . . . .	4
3.2.2. Eje . . . . .	5
3.2.3. Cuerpo . . . . .	6
3.2.4. Cabina . . . . .	8
3.2.5. Brazo1 . . . . .	9
3.2.6. Brazo2 . . . . .	10
3.2.7. Grúa . . . . .	11

# Capítulo 1

## Introducción

Antes de pasar a presentar el modelo que se ha implementado, vamos a pasar a explicar en que consiste.

Se ha realizado el modelado de una Grúa en la que se moverán los brazos. Tenemos que tener en cuenta que la cabina siempre tiene que estar en posición horizontal. Por lo tanto, tendremos que el grado de libertad de la cabina  $\alpha$  será una variable que depende del primer brazo, donde el grado de libertad de este es  $\beta$ , es decir

$$\alpha = f(\beta)$$

Si este sube o baja, la cabina tendrá que permanecer en todo momento en posición horizontal. Por otro lado, si el brazo2 baja o sube también tendremos que mantener la cabina en posición horizontal del mismo modo, por lo que  $\alpha$  también dependerá del grado de libertad del segundo brazo  $\delta$ , es decir tendremos que crear una función de dos variables para mantener la cabina en posición vertical.

$$\alpha = f(\beta, \delta)$$

Vamos a mostrar una imagen de cual será el resultado final tras la modelización.

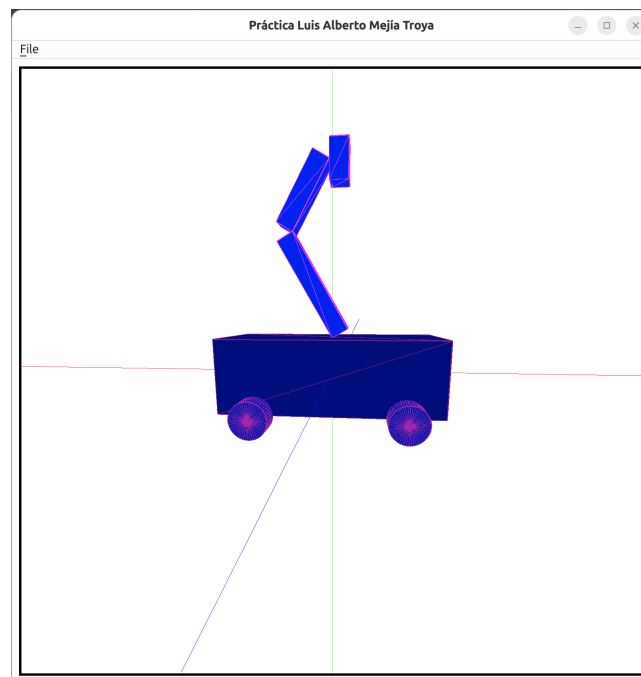
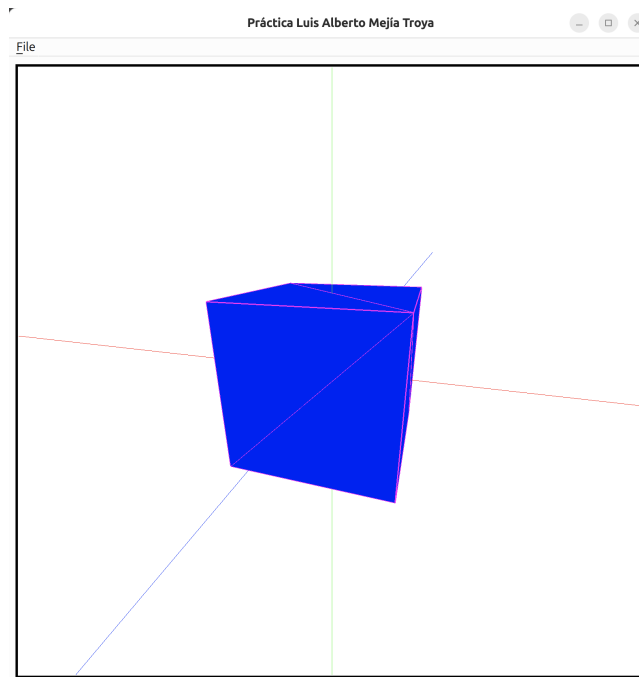


Figura 1.1: Grúa

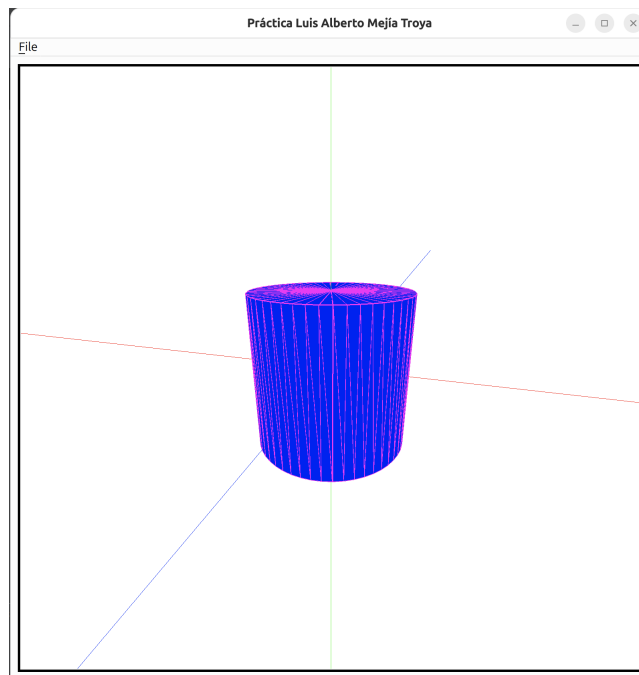
Los elementos que se han utilizado para la creación del modelo son dos:

- Cilindros
- Cubos

Ambos de tamaños unitarios.



(a) Cubo.



(b) Cilindro.

Figura 1.2: Elementos utilizados para el modelo.

Los movimientos que se han introducido son dos como ya hemos comentado en la parte superior. Por un lado, tenemos el **brazo1** que realizará una rotación en el eje  $Z$  y por otro lado, tenemos el **brazo2** que realizará una rotación en el eje  $Z$  y también realizará un escalado para cumplir con los objetivos de la práctica. También realizaremos rotaciones en la cabina para mantenerla en una posición horizontal en cada momento.

## Capítulo 2

# Transformaciones

Las transformaciones que se han realizado para conseguir el objeto final han sido las estudiadas en la materia, es decir, hemos utilizado unicamente transformaciones de tres tipos:

- Rotación
- Escalado
- Traslación

Por otro lado, tenemos tres variables para controlar el movimiento de los brazos

- $\alpha$  es la variable que va a mantener la cabina siempre en posición vertical y por lo tanto, tendremos que utilizar una función de dos variables para controlar el movimiento que va a depender de los dos brazos.
- $\beta$  es la variable que nos permite controlar el primer brazo, va desde los  $-60^\circ$  hasta los  $-120^\circ$ . Este brazo, tendrá un grado de libertad, ya que solamente haremos rotaciones.
- $\delta, \gamma$  son las variables que usaremos para el control del segundo brazo,  $\gamma$  va desde los valores  $120^\circ$  hasta los  $170^\circ$ . Por otro lado, tenemos  $\delta$  que realizará un escalado para aumentar el brazo o disminuirlo.

### 2.1. Función

Para controlar el movimiento y hacer que sea dependiente, tendremos hallar un modelo para  $f(x, y)$ . Como se ha realizado un modelado jerárquico, los parámetros no son visibles para la implementación de la última clase, ya que debemos mantener los principios básicos de la programación orientada a objetos. Por lo tanto, una aproximación de esta función seria:

$$f(x, y) = (h \circ g)(x, y) = h(g(x, y))$$

Donde la función  $g$  representa la variación en de angulo que hay con respecto a  $\gamma$  y  $h$  es la función que realiza la suma o la resta de dicha variación en cada caso.

De esta forma, podemos crear dicho método en cada una de las clases donde se realizar el movimiento siguiendo el estándar de la programación orientada a objetos.

## Capítulo 3

# Grafos de escena

Una vez introducido el modelo, se va a presentar el grafo completo de escena y en las siguientes secciones se va a descomponer dicho diagrama en partes y se explicará detalladamente.

### 3.1. Grafo general

El grafo que vamos a introducir, esta compuesto por dos nodos hojas que son los elementos base que hemos utilizado y comentado en la sección anterior. La raíz de árbol, esta formado por el elemento grúa que será la unión del Cuerpo con los brazos de la grúa.

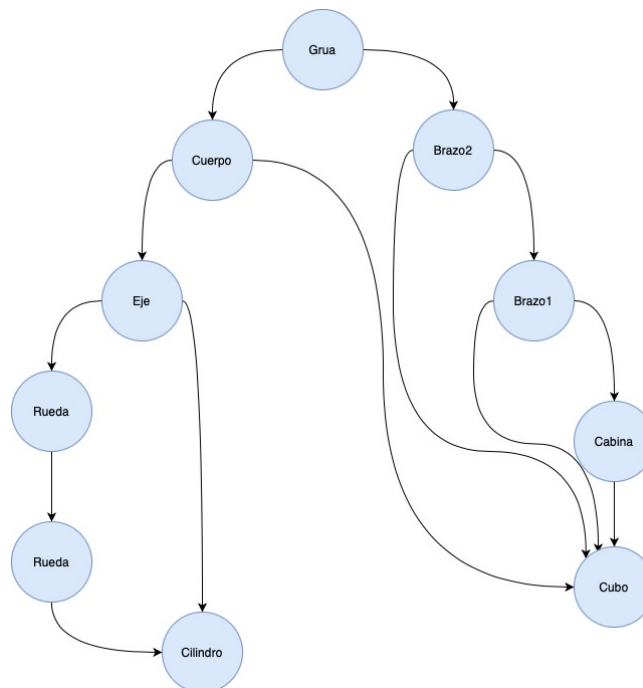


Figura 3.1: Diagrama completo

### 3.2. Especificación por partes

Una vez presentado el grafo general, se va a introducir cada una de las parte del grafo especificando el código necesario para su construcción.

#### 3.2.1. Rueda

Para la rueda solamente necesitaremos realizar un escalado y una rotación y estará centrada en el centro de coordenadas  $O$ .

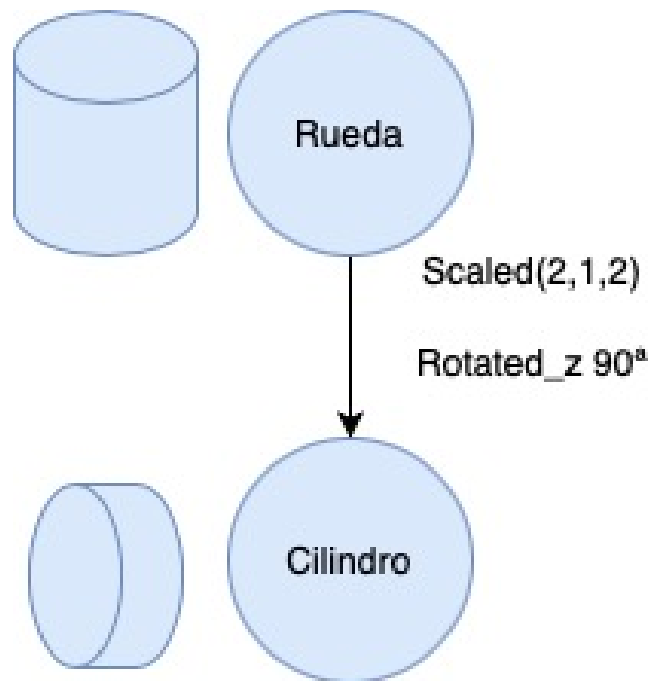


Figura 3.2: Rueda

El código implementado para la realización del modelo es el siguiente

#### Rueda

```
glRotated(90, 0, 0, 1);
glScaled(2, 1, 2);
```

Realizando esto, conseguimos el siguiente resultado

### 3.2.2. Eje

Para la realización del eje, hemos utilizado el modelo de la rueda y el cilindro para crear de esta manera un eje.

Hemos utilizado únicamente una rueda ya que, mediante transformaciones, podemos ponerla en distintas partes y finalmente dibujarla. Así, realizamos un uso eficiente de la memoria.

El código que hemos implementado para la realización de esta parte es la siguiente

#### Eje

```
glRotated(90, 0, 0, 1);
glScaled(1, 6, 1);
Cylinder.draw();
glTranslated(-3, 0, 0);
Rueda.draw();
glTranslated(3, 0, 0);
Rueda.draw();
```

Tenemos que comentar que el código que aparece aquí no es el correcto para realizar una implementación final ya que hemos eliminado la información para escoger la matriz MODELVIEW y el guardado de la matriz original para que sea más legible el código.

Finalmente, el resultado obtenido es el siguiente

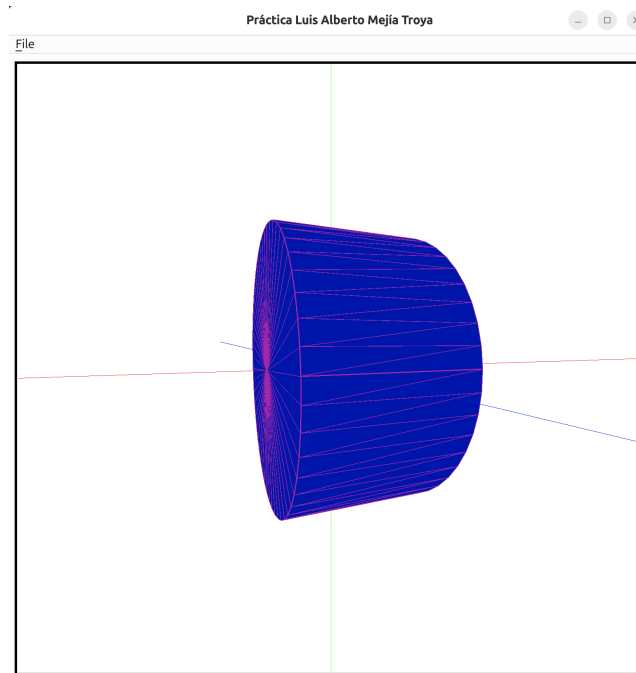


Figura 3.3: rueda

### 3.2.3. Cuerpo

El Cuerpo es la “carrocería del vehículo”, los elementos que hemos utilizado para la construcción es el cubo y el eje. El grafo de diseño es el siguiente

El código asociado a esta parte es el siguiente

#### Cuerpo

```
glScaled(12, 4, 5);  
Cube.draw();  
glTranslated(-4, -2, 0);  
glRotated(90, 0, 1, 0);  
Eje.draw_();  
glTranslated(4, -2, 0);  
glRotated(90, 0, 1, 0);  
Eje.draw();
```

El resultado final, tras la implementación del código resultante es el siguiente



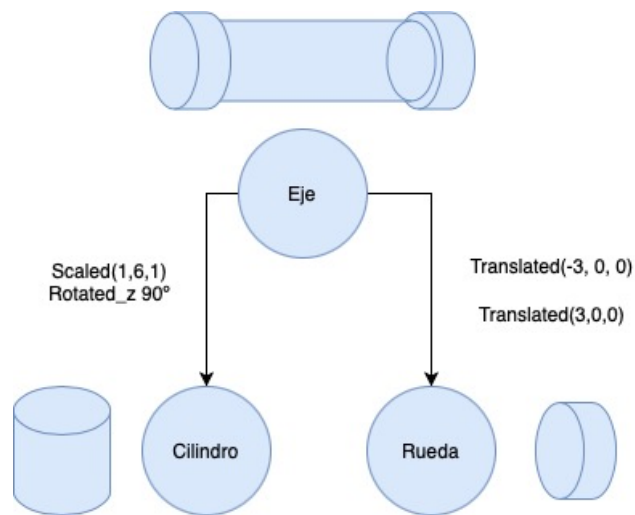


Figura 3.4: Eje

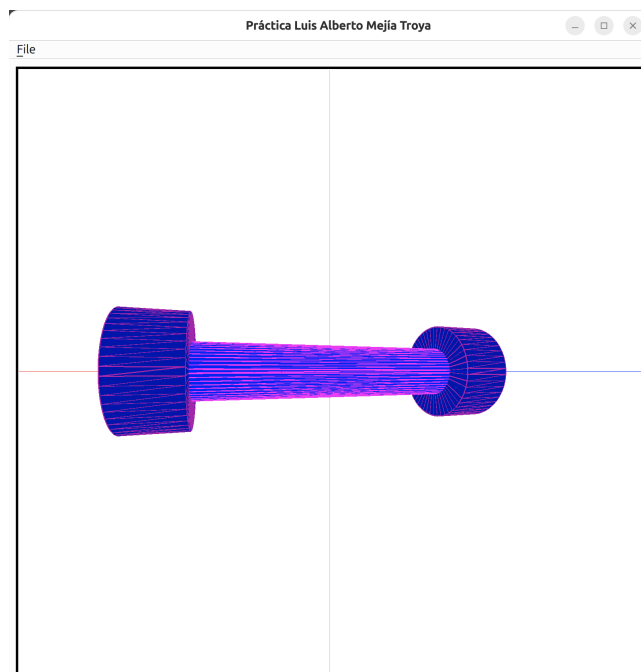


Figura 3.5: Eje

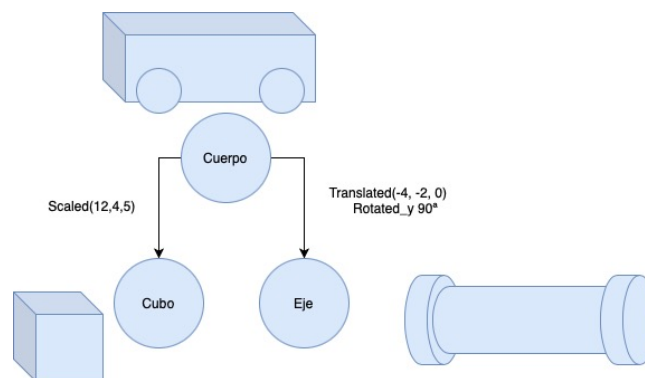


Figura 3.6: Cuerpo

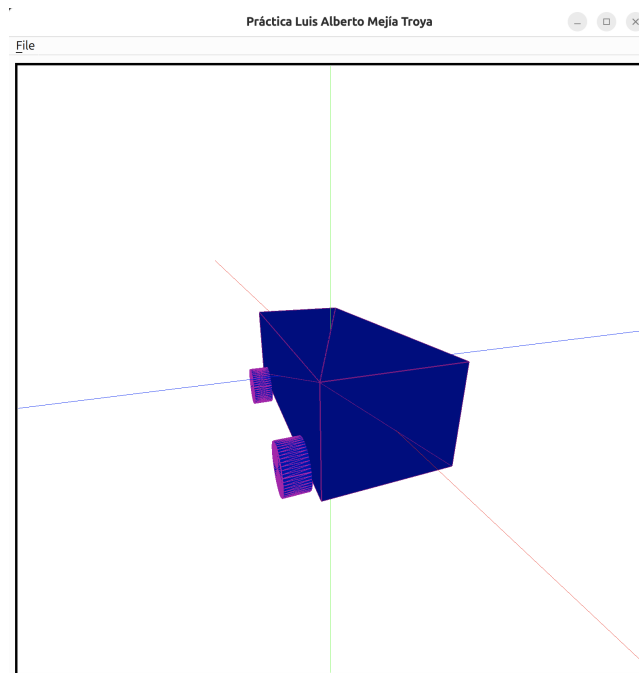


Figura 3.7: Cuerpo

### 3.2.4. Cabina

Para la cabina, hemos realizado unicamente un escalado, la translaticio que vemos en el diagrama siguiente, es para indicar el punto con respecto va a girar la cabina.

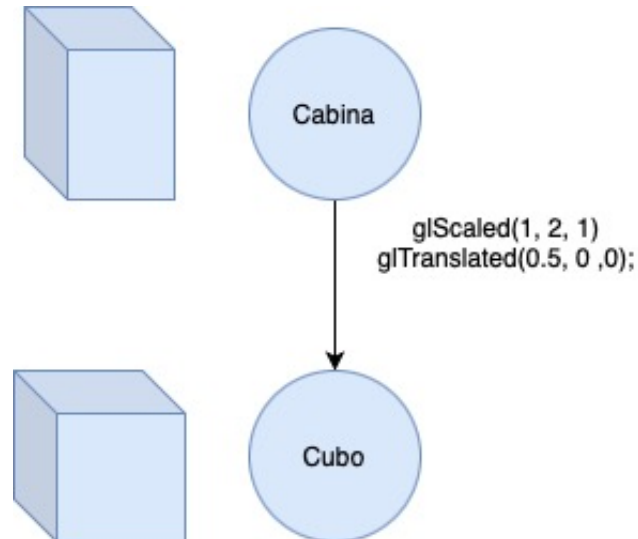


Figura 3.8: Cabina

Finalmente, el código que hemos utilizado para la implementación es el siguiente.

#### Cabina

```
glTranslated(0.5, 0 ,0);  
glScaled(1, 2, 1);  
Cube.draw();
```

### 3.2.5. Brazo1

Para la implementación del primer brazo, hemos usado la “Cabina” y hemos utilizado un Cubo. El código asociado a esta parte de la implementación es el siguiente

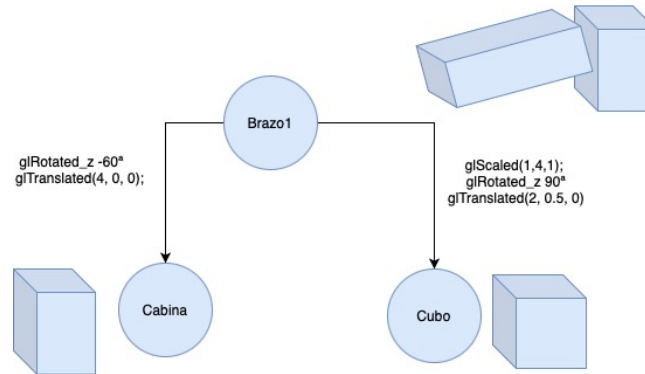


Figura 3.9: Brazo1

#### Brazo1

```
glTranslated(2, 0.5, 0);
glRotated(90, 0, 0, 1);
glScaled(1, 4, 1);
Cube.draw();
glTranslated(4, 0, 0);
glRotated(-60, 0, 0, 1);
Cabina.draw();
```

Hemos girado el objeto modelado, realmente, la cabina esta tumbada hacia para que, cuando lo juntemos con el siguiente brazo, este en la posición correcta. Asi, el objeto modelado queda de la siguiente forma

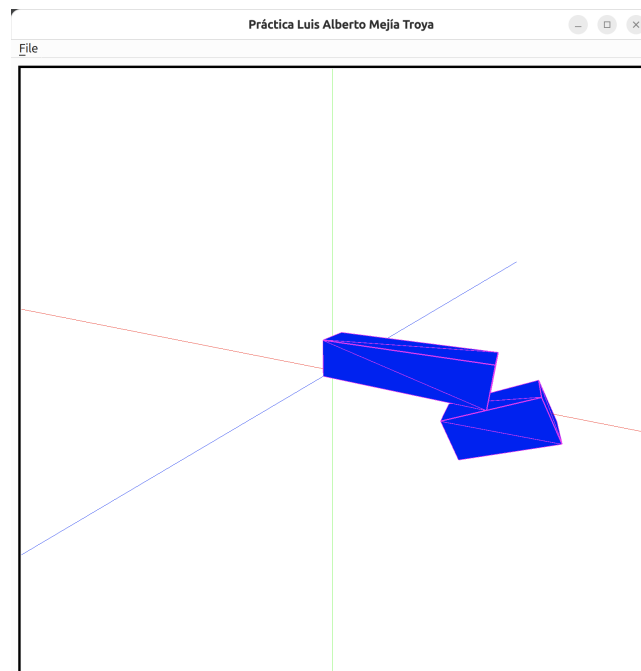


Figura 3.10: Brazo1

### 3.2.6. Brazo2

Para la realización del segundo brazo, hemos seguido un proceso similar al primero. En este caso, hemos usado un cubo y el primer brazo para encajar dichas partes.

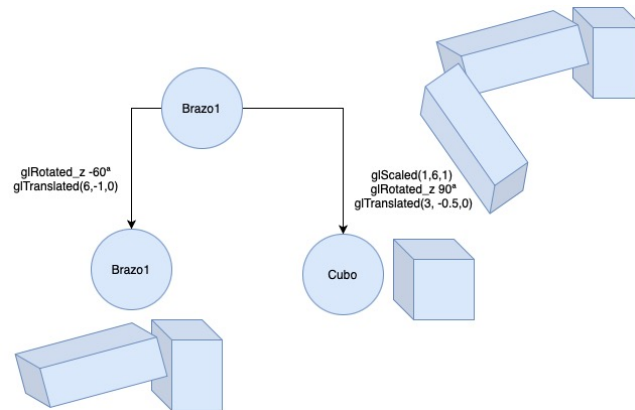


Figura 3.11: Brazo2

Por lo que el código asociado a la implementación es el siguiente

#### Brazo2

```
glTranslated(3, -0.5, 0);
glRotated(90, 0, 0, 1);
glScaled(1, 6, 1);
Cube.draw();
glTranslated(6, -1, 0);
glRotated(-60, 0, 0, 1);
Brazo.draw();
```

Y finalmente el modelo que obtenemos de esta implementación es el siguiente

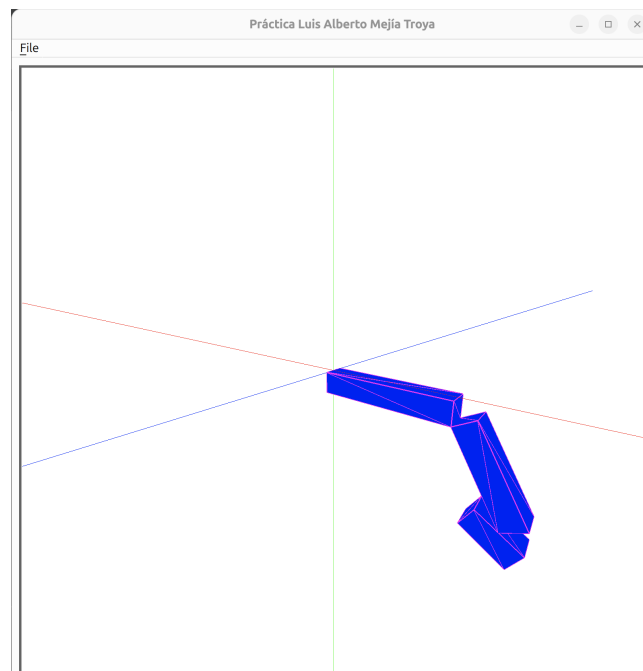


Figura 3.12: Brazo2

### 3.2.7. Grúa

Esta parte consiste en juntar la “carrocería” con los brazos, para ello, tenemos que directamente dibujar el Cuerpo y luego realizaremos una serie de transformaciones para poner los brazos en la posición correcta. El código asociado para poder modelar el objeto final es el siguiente

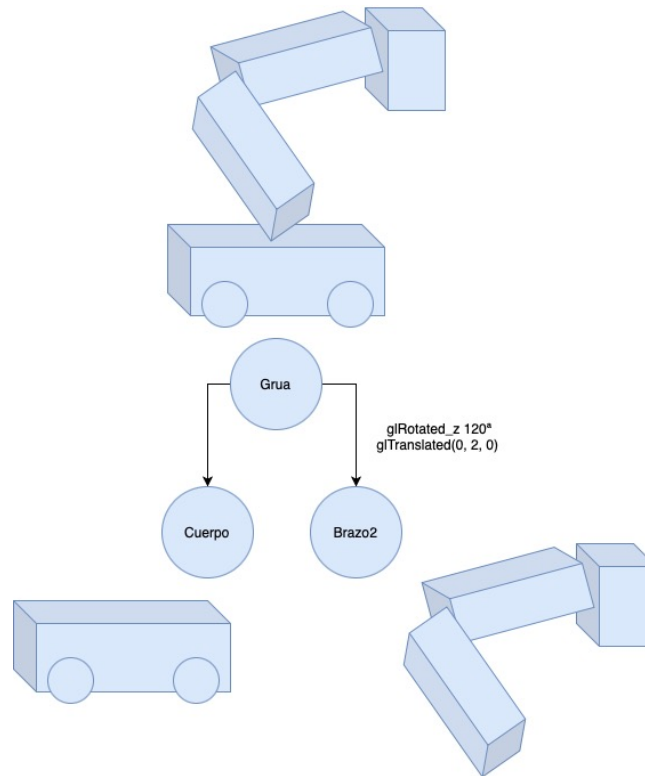


Figura 3.13: Grúa

#### Grúa

```
Cuerpo.draw();  
glTranslated(0, 2, 0);  
glRotated(120, 0, 0, 1);  
Brazo.draw();
```

El objeto final es el que hemos enseñado en el capítulo 1