



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° 06

NOMBRE COMPLETO: Reyes Romero Luis Fernando

N° de Cuenta: 318155320

GRUPO DE LABORATORIO: 11

GRUPO DE TEORÍA: 06

SEMESTRE 2024-2

FECHA DE ENTREGA LÍMITE: 27 de marzo de 2024

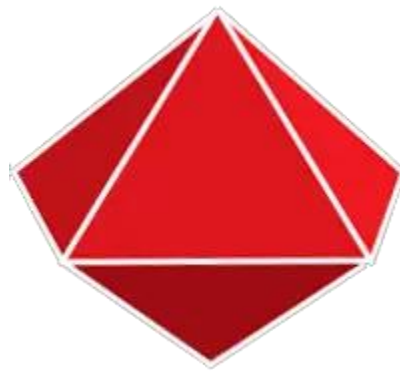
CALIFICACIÓN: _____

REPORTE DE PRÁCTICA:

Actividades

- Crear un dado de 10 caras y texturizarlo por medio de código.
- Importar el modelo de su coche con sus 4 llantas acomodadas y tener texturizadas las 4 llantas (diferenciar caucho y rin)
- Texturizar el logo de la Facultad de ingeniería en el cofre de su propio modelo de coche

Para la primera actividad lo que tenemos que hacer es un dado de 10 caras y ese dado lo tenemos que crear por medio de código, así como su textura. Entonces, lo que buscamos hacer es recrear esta figura por medio de código.



Para esto necesitamos indicar los vértices y las coordenadas de los vértices para que se pueda crear la figura. Cabe mencionar, que se deben de declarar todos los vértices de las caras porque después de crear la figura vamos a texturizar cada una de las caras del dado. A continuación, se muestran las modificaciones del código.

Creación de la textura

```
Texture Dado10Texture;
```

Índices del dado

```
void CrearDado10()
{
    unsigned int Dado10_indices[] = {
        //top
        0, 1, 2,
        3, 4, 5,
        6, 7, 8,
        9, 10, 11,
        12, 13, 14,

        //bottom
        15, 16, 17,
        18, 19, 20,
        21, 22, 23,
        24, 25, 26,
        27, 28, 29,
    };
}
```

Vértices del cubo

```
GLfloat Dado10_vertices[] = {
    // Parte de arriba
    //Cara 1
    //x      y      z      S      T      NX      NY      NZ
    -0.5f, -0.5f, 0.5f, 0.0f, 0.55f, 0.0f, 0.0f, -1.0f,
    0.5f, -0.5f, 0.5f, 0.2f, 0.55f, 0.0f, 0.0f, -1.0f,
    0.0f, 0.5f, -0.35f, 0.1f, 1.0f, 0.0f, 1.0f, 0.0f,

    //Cara 2
    0.5f, -0.5f, 0.5f, 0.38f, 0.55f, 0.0f, 0.0f, -1.0f,
    1.0f, -0.5f, -0.5f, 0.6f, 0.55f, 1.0f, 0.0f, 0.0f,
    0.0f, 0.5f, -0.35f, 0.50f, 1.0f, 0.0f, 1.0f, 0.0f,

    //Cara 3
    1.0f, -0.5f, -0.5f, 0.6f, 0.0f, 1.0f, 0.0f, 0.0f,
    0.0f, -0.5f, -1.2f, 0.8f, 0.0f, 0.0f, 0.0f, 1.0f,
    0.0f, 0.5f, -0.35f, 0.7f, 0.5f, 0.0f, 1.0f, 0.0f,

    //Cara 4
    0.0f, -0.5f, -1.2f, 0.8f, 0.55f, 0.0f, 0.0f, 1.0f,
    -1.0f, -0.5f, -0.5f, 1.00f, 0.55f, -1.0f, 0.0f, 0.0f,
    0.0f, 0.5f, -0.35f, 0.9f, 1.00f, 0.0f, 1.0f, 0.0f,

    //Cara 5
    -1.0f, -0.5f, -0.5f, 0.2f, 0.00f, -1.0f, 0.0f, 0.0f,
    -0.5f, -0.5f, 0.5f, 0.4f, 0.00f, 0.0f, 0.0f, -1.0f,
    0.0f, 0.5f, -0.35f, 0.3f, 0.50f, 0.0f, 1.0f, 0.0f,
}
```

Vértices del cubo

```
//Parte de Abajo
//Cara 1
//x      y      z      S      T      NX      NY      NZ
-0.5f, -0.5f,  0.5f,  0.4f,  0.55f,  0.0f,  0.0f, -1.0f,
 0.5f, -0.5f,  0.5f,  0.2f,  0.55f,  0.0f,  0.0f, -1.0f,
 0.0f, -1.5f, -0.35f,  0.3f,  1.0f,  0.0f,  1.0f,  0.0f,

//Cara 2
 0.5f, -0.5f,  0.5f,  0.38f,  0.0f,  0.0f,  0.0f, -1.0f,
 1.0f, -0.5f, -0.5f,  0.6f,  0.0f,  1.0f,  0.0f,  0.0f,
 0.0f, -1.5f, -0.35f,  0.5f,  0.5f,  0.0f,  1.0f,  0.0f,

//Cara 3
 1.0f, -0.5f, -0.5f,  1.0f,  0.0f,  1.0f,  0.0f,  0.0f,
 0.0f, -0.5f, -1.2f,  0.8f,  0.0f,  0.0f,  0.0f,  1.0f,
 0.0f, -1.5f, -0.35f,  0.9f,  0.6f,  0.0f,  1.0f,  0.0f,

//Cara 4
 0.0f, -0.5f, -1.2f,  0.80f,  0.55f,  0.0f,  0.0f,  1.0f,
-1.0f, -0.5f, -0.5f,  0.60f,  0.55f, -1.0f,  0.0f,  0.0f,
 0.0f, -1.5f, -0.35f,  0.70f,  1.00f,  0.0f,  1.0f,  0.0f,

//Cara 5
-1.0f, -0.5f, -0.5f,  0.20f,  0.00f, -1.0f,  0.0f,  0.0f,
-0.5f, -0.5f,  0.5f,  0.00f,  0.00f,  0.0f,  0.0f, -1.0f,
 0.0f, -1.5f, -0.35f,  0.10f,  0.55f,  0.0f,  1.0f,  0.0f,
```

Se llama a la función

```
CrearDado10();
```

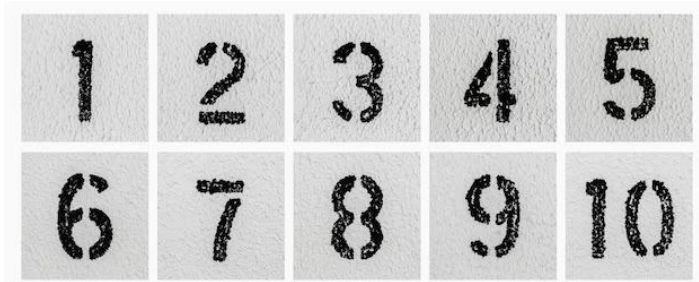
Ruta de la textura

```
Dado10Texture = Texture("Textures/dado_colores.tga");
Dado10Texture.LoadTextureA();
```

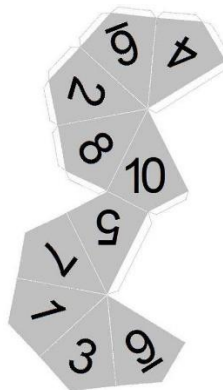
Se dibuja el dado ya texturizado

```
model = glm::mat4(1.0);
model = glm::translate(model, glm::vec3(1.0f, 1.0f, -2.5f));
glUniformMatrix4fv(uniformModel, 1, GL_FALSE, glm::value_ptr(model));
Dado10Texture.UseTexture();
meshList[5]->RenderMesh();
```

Como vimos en el ejercicio de esta práctica las coordenadas S y T, son las coordenadas que se van a usar para la textura. La textura que usamos para el dado fue la siguiente.



Escogí esta textura debido a que la textura para armar un cubo de 10 lados es muy complicada intenté texturizar el dado con la imagen que se muestra a continuación, pero fue un fracaso total.



Ejercicio del coche

Para el texturizado del coche vamos a usar el software 3ds Max. Vamos a hacer algo parecido a lo que ya habíamos hecho cuando texturizamos el dado por medio de 3ds Max. Cargamos la imagen de la llanta y vamos acomodando los colores de la llanta correspondiente a cada elemento. Pero antes de hacer eso dentro del código vamos a crear nuestros modelos y vamos a escribir las rutas donde se encuentran nuestros modelos para que se carguen de forma correcta.

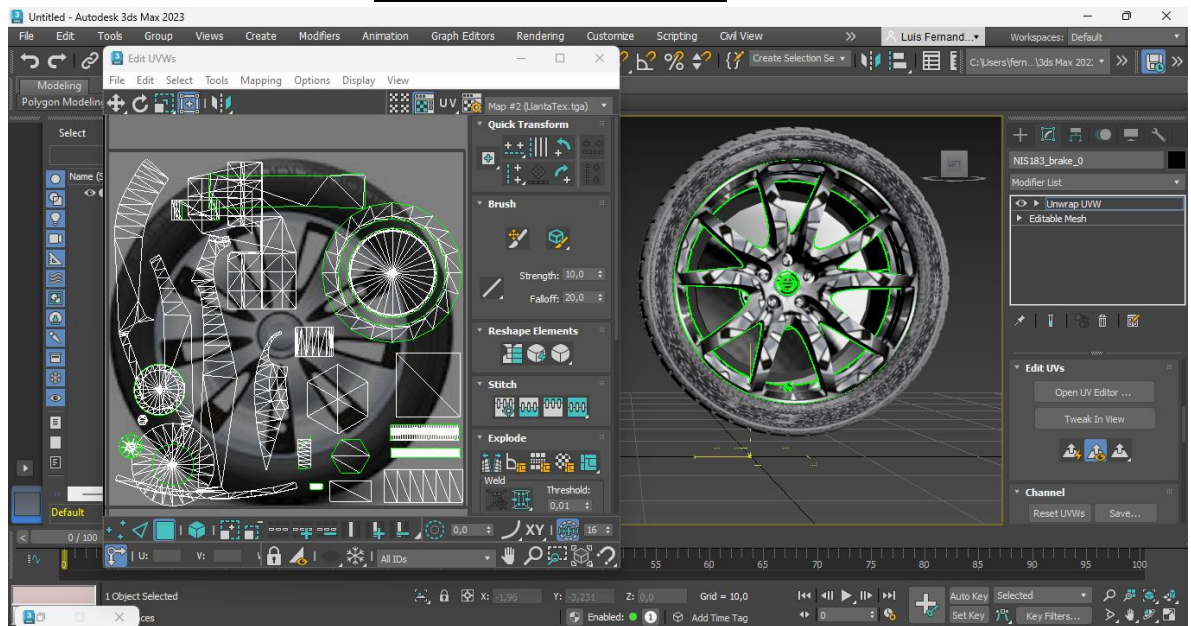
Creación de modelos

```
Model Dado_M;
Model Kitt_M;
Model Llanta_M;
Model Blackhawk_M;
Model Coche;
Model Cofre;
Model Llanta1;
Model Llanta2;
Model Llanta3;
Model Llanta4;
```

Rutas

```
brickTexture = Texture("Textures/brick.png");
brickTexture.LoadTextureA();
dirtTexture = Texture("Textures/dirt.png");
dirtTexture.LoadTextureA();
plainTexture = Texture("Textures/plain.png");
plainTexture.LoadTextureA();
pisoTexture = Texture("Textures/piso.tga");
pisoTexture.LoadTextureA();
dadoTexture = Texture("Textures/dado_animales_recortado.tga");
dadoTexture.LoadTextureA();
logofiTexture = Texture("Textures/escudo_fi_color.tga");
logofiTexture.LoadTextureA();
Dado10Texture = Texture("Textures/dado_colores.tga");
Dado10Texture.LoadTextureA();
```

Texturizado por 3ds Max

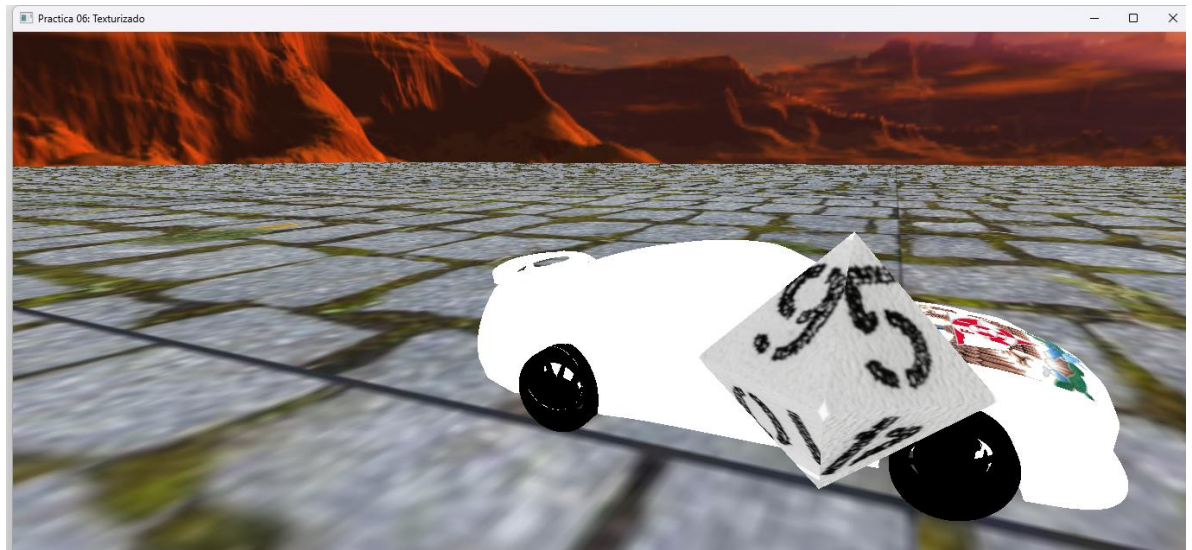


Para la textura del coche se hace algo similar solo que aquí vamos a texturizar el cofre con el logo de la facultad. Una vez hecho los modelos e importarlos con la extensión .obj vamos a modificar los archivos .mtl para que la textura se pueda ver de forma correcta. Después vamos a dibujar los modelos y ejecutamos el código.

Ejecución del código

Dado de 10 lados





Coche texturizado



Algo que me parece curioso es que no sale la textura de las llantas, pero del cofre si y esto me paso al revés en el ejercicio de la práctica de iluminación. Las llantas con textura salían bien, pero el cofre no.

Lista de problemas

- Textura. Más que nada fue la textura porque en el 3ds Max las llantas si salen con la textura ya implementada, pero a la hora de renderizar las llantas dentro del código salen de color totalmente negro. No sé si es mi computadora, pero me paso algo similar en el ejercicio de la práctica de iluminación solo que ahí después de un tiempo ya se lograba ver la textura de las llantas, pero en ese ejercicio el cofre se veía totalmente blanco y en este reporte ya se puede ver el logo de la facultad, pero las llantas no. Entonces no sé si es mi computadora o es un error que cometí al importar los objetos.

Conclusión

Lo único que sentí complejo fue hacer el dado de 10 caras por el manejo de vértices e índices, ya que al colocar de forma errónea estos datos la figura puede estar chueca, los lados pueden ser distintos, no tiene la simetría correcta, entre otras cosas. Asimismo, la textura de este dado es muy compleja y más si se requiere hacer la textura del armado del dado de 10 lados. Además, observe que se tienen que implementar un dado con unas buenas dimensiones para que la textura no quede tan justa como la mía. Por parte del coche la dificultad estuvo bien, siento que texturizar por medio de 3ds Max es demasiado sencillo siento que lo único complejo es acomodar los triángulos con su correspondiente color para que la textura este correcta. Gracias a la explicación en clase se pudo resolver esta práctica de forma correcta, de igual forma, los vídeos son de gran ayuda por si te llegas a perder en la explicación de la clase.

Bibliografía en formato APA

No se uso ninguna fuente externa para esta práctica.