



FACULTAD DE INGENIERÍA - Course 2019/ 2019

SECRETARÍA/DIVISIÓN: DIVISIÓN DE INGENIERÍA ELÉCTRICA  
ÁREA/DEPARTAMENTO: INGENIERÍA EN COMPUTACIÓN

LABORATORIO DE COMPUTACIÓN GRÁFICA E INTERACCIÓN HUMANO  
COMPUTADORA:

**Modelado Geométrico.**

Reynaldo Martell Avila

**PRÁCTICA 4**

## Contents

<b>1</b>	<b>Objetivos de aprendizaje</b>	<b>2</b>
1.1	Objetivos generales: . . . . .	2
1.2	Objetivos específicos: . . . . .	2
<b>2</b>	<b>Recursos a emplear</b>	<b>2</b>
2.1	Software . . . . .	2
2.2	Equipos . . . . .	2
2.3	Instrumentos . . . . .	2
<b>3</b>	<b>Fundamento Teórico</b>	<b>2</b>
3.1	Desarrollo de actividades . . . . .	2
3.2	Ejercicios . . . . .	4
<b>4</b>	<b>Observaciones y Conclusiones</b>	<b>4</b>
<b>5</b>	<b>Anexos</b>	<b>4</b>

# 1 Objetivos de aprendizaje

## 1.1 Objetivos generales:

El alumno repasará como crear buffers de OpenGL, leer archivos, comprenderá los diferentes tipos de proyección y las funciones de la librería glm para crear éstas, así como comprenderá los diferentes sistemas de referencia de OpenGL. Del mismo modo practicará como colocar en la escena diferentes geometrías.

## 1.2 Objetivos específicos:

El alumno practicará crear geometrías con índices, revisará los sistemas de referencia que se aplican en OpenGL, comprenderá la utilización de la matriz de modelo, vista, proyección y la zona de dibujo.

# 2 Recursos a emplear

## 2.1 Software

Sistema Operativo: Windows 7 Ambiente de Desarrollo: Visual Studio 2017.

## 2.2 Equipos

Los equipos de cómputo con los que cuenta el laboratorio de Computación Gráfica

## 2.3 Instrumentos

# 3 Fundamento Teórico

- **Presentación de conceptos.**

El Modelado Geométrico consiste en la construcción de un modelo a partir de primitivas, es decir, elementos más sencillo. Un Modelo Geométrico es la representación de las características geométrica de una entidad concreta o abstracta.

El Modelado Geométrico y el Modelado Jerárquico hace uso de la composición de operaciones matriciales anidando transformaciones geométricas, por lo cual es importante repasar esto, además de que se explica a grandes rasgos como se forman las primitivas geométricas.

- **Datos necesarios.** Librería OpenGL 3.3, librería de creación de ventanas, IDE de desarrollo (Visual Studio 2017).

## 3.1 Desarrollo de actividades

1. Ejecutar el código base de la práctica **04-Modelado Geométrico**, observar la ejecución.
2. Explicar el código de la Clase **AbstractModel.h** y sus implementaciones **Cylinder.cpp**, **Sphere.cpp** y **Box.cpp**.

3. Agregar las cabeceras de los modelos geométricos Esfera, Caja y Cilindro que se muestran a continuación 1.

Ejemplo 1: Inclusión de cabeceras de las formas geométricas.

```
1 // Model geometric includes
2 #include "Headers/Sphere.h"
3 #include "Headers/Cylinder.h"
4 #include "Headers/Box.h"
```

4. Declarar dos objetos de tipo Sphere, Cylinder y Box como se muestra en el ejemplo 2

Ejemplo 2: Declarar objetos de tipo esfera, cilindro y caja.

```
1 Sphere sphere1(20, 20);
2 Cylinder cylinder1(20, 20, 0.5, 0.5);
3 Box box1;
```

5. En el método **init** agregar las inicializaciones de los buffers de los modelos geométricos 4

Ejemplo 3: Inicialización de esfera, cilindro y caja.

```
1 sphere1.init();
2 sphere1.setShader(&shader);
3 sphere1.setColor(glm::vec4(0.3, 0.3, 1.0, 1.0));
4
5 cylinder1.init();
6 cylinder1.setShader(&shader);
7 cylinder1.setColor(glm::vec4(0.3, 0.3, 1.0, 1.0));
8
9 box1.init();
10 box1.setShader(&shader);
11 box1.setColor(glm::vec4(0.3, 0.3, 1.0, 1.0));
```

6. Agregar en el método **destroy** la eliminación explícita de los objetos, con el fin de liberar espacio de memoria (Buffers y atributos de vertices creados) como se muestra en el ejemplo ??

Ejemplo 4: Inicialización de esfera, cilindro y caja.

```
1 sphere1.destroy();
2 cylinder1.destroy();
3 box1.destroy();
```

7. Agregar en el método **applicationLoop** después de la matriz de modelo el render de una esfera **sphere1.render(model);** y ejecutar el programa.
8. Agregar enseguida la llamada al método **sphere1.enableWireMode();**.  
¿Para que sirve ésta?

¿Para que sirve el método **enableFillMode**?

Abrir esta función y reporte que funciones son llamadas internamente.

9. Colocar otros modelos geométricos en diferentes posiciones y tamaños.
10. Definir un modelo a construir a partir de primitivas geométricas. El modelo debe ser en tres dimensiones y consistir de, al menos, seis elementos, de tal forma que se practique el uso de la composición de operaciones matriciales. Cada alumno define el modelo a generar.

### 3.2 Ejercicios

1. Agregue las transformaciones necesarias para que los elementos en pantalla puedan ser trasladados en el eje X y en el eje Y, al presionar teclas.
2. Agregue las transformaciones necesarias para que los elementos en pantalla puedan ser manipulados mediante una rotación sobre el eje Y, al presionar alguna tecla.

## 4 Observaciones y Conclusiones

## 5 Anexos

1. Cuestionario previo.
  - (a) ¿Qué es modelado geométrico?.
  - (b) Investigue el algoritmo para generar una esfera.
  - (c) Investigue el algoritmo para generar un cilindro.
2. Actividad de investigación previa.
  - (a) Realizar un **git pull origin master** y un **git pull myrepo master**, antes de comenzar la práctica.