



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° 01**

**NOMBRE COMPLETO:** Oropeza Sánchez Guadalupe

**N° de Cuenta:** 317144547

**GRUPO DE LABORATORIO:** 02

**GRUPO DE TEORÍA:** 06

**SEMESTRE 2024-2**

**FECHA DE ENTREGA LÍMITE:** 17 de febrero del 2024

**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.

INICIALES. Las iniciales se crearon con triángulos, en esta parte del código están cada todos los vértices utilizados para cada letra

```
//LETRA G
-2.0f, 0.8f, 0.0f,
-2.0f, -0.8f, 0.0f,
-1.8f, -0.8f, 0.0f,

-2.0f, 0.8f, 0.0f,
-1.8f, -0.8f, 0.0f,
-1.8f, 0.80f, 0.0f,
//
-1.8f, 0.80f, 0.0f,
-1.2f, 0.8f, 0.0f,
-1.2, 0.6f, 0.0f,

-1.8f, 0.80f, 0.0f,
-1.2, 0.6f, 0.0f,
-1.8f, 0.6f, 0.0f,
//
-1.8f, -0.6f, 0.0f,
-1.8f, -0.8f, 0.0f,
-0.9f, -0.8f, 0.0f,

-1.8f, -0.6f, 0.0f,
-0.9f, -0.8f, 0.0f,
-0.9f, -0.6, 0.0f,
//

-1.5f, -.25f,0.0f,
-1.5f, -0.45f, 0.0f,
-0.9f, -0.25f, 0.0f,

-0.9f, -0.25f, 0.0f,
-0.9f, -0.45f, 0.0f,
```

```
//LETRA O
/*r*/ -0.6f, 0.8f, 0.0f,
/*s*/ -0.4f, 0.8f, 0.0f,
/*q*/ -0.6f, -0.8f, 0.0f,

/*s*/ -0.4f, 0.8f, 0.0f,
/*q*/ -0.6f, -0.8f, 0.0f,
/*t*/ -0.4f, -0.8f, 0.0f,

/*r*/ -0.6f, 0.8f, 0.0f,
/*w*/ -0.6f, 0.6f, 0.0f,
/*u*/ 0.4f, 0.8f, 0.0f,

/*w*/ -0.6f, 0.6f, 0.0f,
/*u*/ 0.4f, 0.8f, 0.0f,
/*v*/ 0.4f, 0.6f, 0.0f,

/*q*/ -0.6f, -0.8f, 0.0f,
/*b1*/ -0.6f, -0.6f, 0.0f,
/*a1*/ 0.4f, -0.6f, 0.0f,

/*q*/ -0.6f, -0.8f, 0.0f,
/*a1*/ 0.4f, -0.6f, 0.0f,
/*z*/ 0.4f, -0.8f, 0.0f,

/*u*/ 0.4f, 0.8f, 0.0f,
/*z*/ 0.4f, -0.8f, 0.0f,
/*d1*/ 0.2f, 0.8f, 0.0f,

/*z*/ 0.4f, -0.8f, 0.0f,
/*d1*/ 0.2f, 0.8f, 0.0f,
/*c1*/ 0.2f, -0.8f, 0.0f,
```

```
//LETRA S
/*f1*/ 0.8f, 0.8f, 0.0f,
/*g1*/ 1.8f, 0.8f, 0.0f,
/*e1*/ 0.8f, 0.6f, 0.0f,

/*g1*/ 1.8f, 0.8f, 0.0f,
/*e1*/ 0.8f, 0.6f, 0.0f,
/*h1*/ 1.8f, 0.6f, 0.0f,

/*f1*/ 0.8f, 0.8f, 0.0f,
/*i1*/ 0.8f, 0.0f, 0.0f,
/*j1*/ 1.0f, 0.0f, 0.0f,

/*f1*/ 0.8f, 0.8f, 0.0f,
/*j1*/ 1.0f, 0.0f, 0.0f,
/*k1*/ 1.0f, 0.8f, 0.0f,

/*n1*/ 0.8f, 0.2f, 0.0f,
/*i1*/ 0.8f, 0.0f, 0.0f,
/*l1*/ 1.8f, 0.2f, 0.0f,

/*i1*/ 0.8f, 0.0f, 0.0f,
/*l1*/ 1.8f, 0.2f, 0.0f,
/*m1*/ 1.8f, 0.0f, 0.0f,

/*l1*/ 1.8f, 0.2f, 0.0f,
/*q1*/ 1.6f, 0.2f, 0.0f,
/*o1*/ 1.6f, -0.8f, 0.0f,

/*l1*/ 1.8f, 0.2f, 0.0f,
/*o1*/ 1.6f, -0.8f, 0.0f,
/*p1*/ 1.8f, -0.8f, 0.0f,
```

Le pasamos los datos al VBO

```
//LETRA O
glGenBuffers(11, &O1);
glBindBuffer(GL_ARRAY_BUFFER, O1);
glBufferData(GL_ARRAY_BUFFER, sizeof(vertices), vertices, GL_STATIC_DRAW);

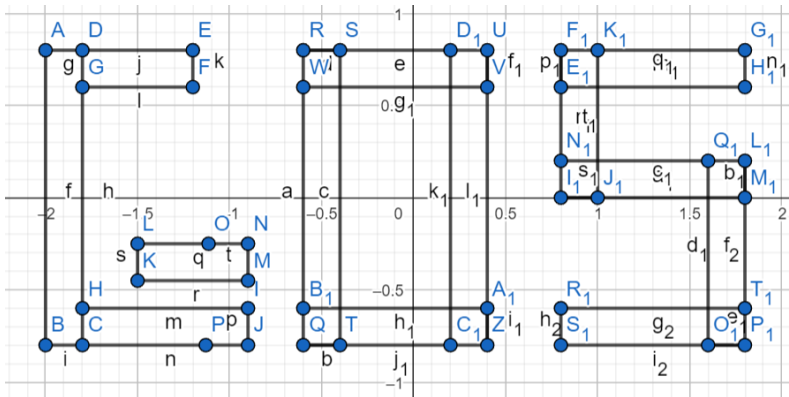
glGenBuffers(12, &O2);
glBindBuffer(GL_ARRAY_BUFFER, O2);
glBufferData(GL_ARRAY_BUFFER, sizeof(vertices), vertices, GL_STATIC_DRAW);

glGenBuffers(13, &O3);
glBindBuffer(GL_ARRAY_BUFFER, O3);
glBufferData(GL_ARRAY_BUFFER, sizeof(vertices), vertices, GL_STATIC_DRAW);
```

Dibujamos los triángulos:

```
//LETRA O  
glDrawArrays(GL_TRIANGLES, 30, 33);  
glDrawArrays(GL_TRIANGLES, 33, 36);  
glDrawArrays(GL_TRIANGLES, 36, 39);  
glDrawArrays(GL_TRIANGLES, 39, 42);  
glDrawArrays(GL_TRIANGLES, 42, 45);  
glDrawArrays(GL_TRIANGLES, 45, 48);
```

Para la creación de las letras se utilizó GeoGebra



Ejecutando el código, se tiene esta salida:



Y para la salida aleatoria de colores para el fondo de pantalla:

```
float rojo = 0.0f;
float azul = 0.0f;
float verde = 0.0f;
float tiempoUltimoCambio = glfwGetTime();
float intervaloCambio = 2.0f;

//Loop mientras no se cierra la ventana
while (!glfwWindowShouldClose(mainWindow))
{
    float tiempoActual = glfwGetTime();
    float tiempoDesdeUltimoCambio = tiempoActual - tiempoUltimoCambio;

    // Si ha pasado el intervalo de cambio de color, actualizar los colores
    if (tiempoDesdeUltimoCambio >= intervaloCambio)
    {
        tiempoUltimoCambio = tiempoActual;

        // Generar valores aleatorios para los componentes de color entre 0 y 1
        rojo = generarAleatorio(0.0f, 1.0f);
        verde = generarAleatorio(0.0f, 1.0f);
        azul = generarAleatorio(0.0f, 1.0f);
    }

    //Recibir eventos del usuario
    glfwPollEvents();
    //Limpiar la ventana
    glClearColor(rojo, verde, azul, 1.0f);
    glClear(GL_COLOR_BUFFER_BIT);
}
```

Se inicializan los variables rojo, verde y azul (RGB). Se ocupa la función ***generarAleatorio***

```
float generarAleatorio(float min, float max) {
    return min + static_cast<float>(rand()) / (static_cast<float>(RAND_MAX / (max - min)));
}
```

La cual nos da un número aleatorio de tipo flotante, con dicho valor generamos valores para el fondo de pantalla



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

En esta práctica lo único que me costó un poco fue la cantidad de vértices con los que trabajaba, por eso en las últimas dos letras todos los vértices tienen un comentario al principio para identificar con que vértice se estaba trabajando.

3.- Conclusión:

No tuvo gran complejidad esta práctica, se lograron ambos ejercicios exitosamente, tanto el modelado de las letras como el cambio de color en la pantalla, la explicación fue detallada y concisa.

4.- Bibliografía.

\* Apuntes del semestre pasado