UNIVERSIDAD NACIONAL DEL ALTIPLANO

Facultad Ingeniería Mecánica, Eléctrica, Electrónica y Sistemas

Escuela Profesional de Ingeniería de Sistemas

Trabajo 3 :

ESPACIOS MÉTRICOS

DOCENTE:

ING. COLLANQUI MARTINEZ FREDY

PRESENTADO POR:

CHOQUE CHURA OLIVER FRAN

SEXTO SEMESTRE 2024-I

PUNO - PERÚ

2024

LINK GITHUB

https://github.com/OliverChoque/Espacios-M-tricos

IMPLEMENTACIÓN EN 2D

```
<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

                <meta    name="viewport"    content="width=device-width,
initial-scale=1.0">

    <title>Representaci&#65533;n 2D de Espacios M&#65533;tricos</title>

                                                                    <link
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap
.min.css" rel="stylesheet">

    <style>

        body {

             background-color: #062745; /* Cambiar el fondo a un color
claro */

            display: flex;

            justify-content: center;

            align-items: center;

            height: 100vh;

            margin: 0;

            flex-direction: column;

        }

        .container {

                    background-color: white; /* Fondo blanco para el
contenedor */

            padding: 20px;

            border-radius: 10px;

            box-shadow: 0 0 10px rgba(0,0,0,0.1);

            position: relative; /* Agregado */

            left: -185px;

        }

        h1 {
```

```
            font-size: 2rem;
            font-weight: bold;
             background-image: linear-gradient(45deg, #ff0066, #ffcc00,
#00ccff, #33cc33);
            background-size: 400%;
            -webkit-background-clip: text;
            color: transparent;
            animation: gradient 5s infinite;
        }
        @keyframes gradient {
            0% { background-position: 0% 50%; }
            50% { background-position: 100% 50%; }
            100% { background-position: 0% 50%; }
        }
        canvas {
            border: 2px solid #333;
            display: block;
            margin: 20px auto;
        }
        .coordinate-input {
            width: 80px;
            display: inline-block;
        }
        .form-group {
            margin-bottom: 15px;
        }
        .properties {
            position: absolute;
            top: 0;
              left: calc(100% + 20px); /* Posicionamiento a la derecha
del contenedor */
            padding: 10px;
            background-color: #f9f9f9;
            border-radius: 10px;
```

```
            box-shadow: 0 0 10px rgba(0,0,0,0.1);
            display: none; /* Inicialmente oculto */
            align-items: center;
            width: 350px; /* Ajustar el ancho */
            height: 777px; /* Ajustar la altura */
        }
        .container.show-properties .properties {
             left: calc(100% + 20px); /* Mover las propiedades al lado
derecho del contenedor */
             display: block; /* Mostrar cuando la clase show-properties
est presente en el contenedor */
        }
        .properties ol {
            margin: 0; /* Agregado */
            padding-left: 20px; /* Agregado */
            text-align: left; /* Agregado */
        }
        .properties li {
            font-size: 14px; /* Tama o de fuente m s peque o */
        }
    </style>
</head>
<body>
    <div class="container text-center">
        <h1>Representaci n 2D de Espacios M tricos</h1>
                    <canvas id="myCanvas" width="500" height="500"
class="my-4"></canvas>
        <div class="form-group">
            <label for="punto1">Coordenadas Punto 1 (x, y):</label>
                    <input type="text" id="punto1" class="form-control
coordinate-input" placeholder="x1, y1" readonly>
        </div>
        <div class="form-group">
            <label for="punto2">Coordenadas Punto 2 (x, y):</label>
                    <input type="text" id="punto2" class="form-control
coordinate-input" placeholder="x2, y2" readonly>
```

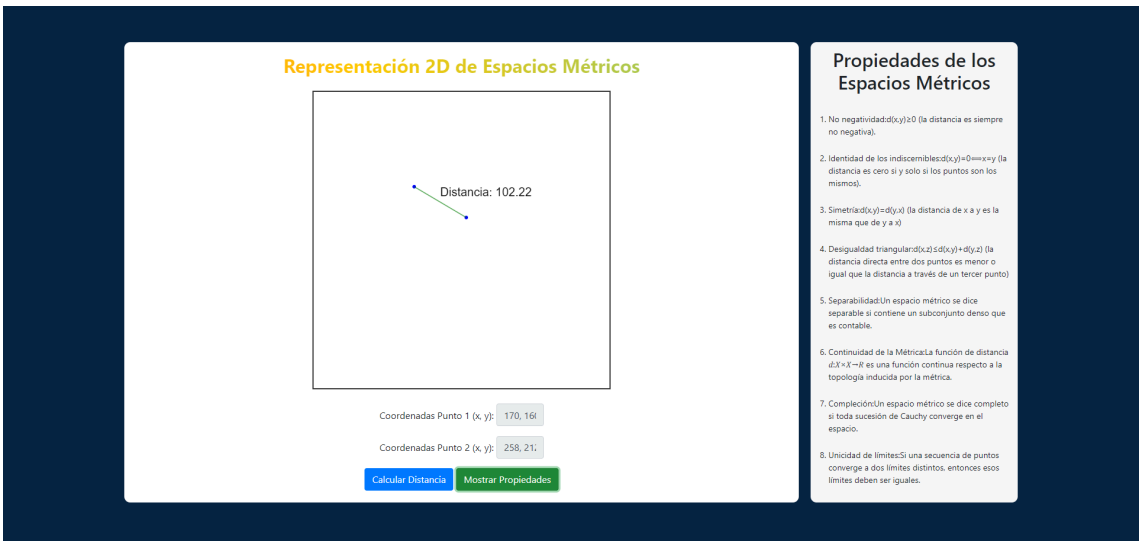
```
        </div>

                    <button  onclick="calcularDistancia()"  class="btn
btn-primary">Calcular Distancia</button>

                   <button  onclick="mostrarPropiedades()"  class="btn
btn-success">Mostrar Propiedades</button> <!-- Nuevo bot@n -->

        <div class="properties" id="properties">

            <h2>Propiedades de los Espacios M@tricos</h2><br>

            <ol>

                <li>No negatividad:d(x,y)=0 (la distancia es siempre
no negativa).</li><br>

                <li>Identidad de los indiscernibles:d(x,y)=0?x=y (la
distancia es cero si y solo si los puntos son los mismos).</li><br>

                <li>Simetr@a:d(x,y)=d(y,x) (la distancia de x a y es
la misma que de y a x)</li><br>

                 <li>Desigualdad triangular:d(x,z)=d(x,y)+d(y,z) (la
distancia directa entre dos puntos es menor o igual que la distancia a
trav@s de un tercer punto)</li><br>

                <li>Separabilidad:Un espacio m@trico se dice separable
si contiene un subconjunto denso que es contable.</li><br>

                <li>Continuidad de la M@trica:La funci@n de distancia
??:??@????? es una funci@n continua respecto a la topolog@a inducida
por la m@trica.</li><br>

                <li>Compleci@n:Un espacio m@trico se dice completo si
toda sucesi@n de Cauchy converge en el espacio.</li><br>

                  <li>Unicidad de l@mites:Si una secuencia de puntos
converge a dos l@mites distintos, entonces esos l@mites deben ser
iguales.</li>

            </ol>

        </div>

    </div>

                                                                <script
src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>

                                                                <script
src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.5.4/dist/umd/popper
.min.js"></script>

                                                                <script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.m
in.js"></script>

    <script>

        function mostrarPropiedades() {

            var properties = document.getElementById('properties');

             properties.style.display = 'block'; // Mostrar el elemento
properties

        }
```

```
        // Obtener el lienzo y el contexto 2D
        var canvas = document.getElementById('myCanvas');
        var ctx = canvas.getContext('2d');

            // Variables para almacenar los puntos ingresados por el
usuario
        var punto1 = null;
        var punto2 = null;

        // Funci□n para dibujar un punto en el plano cartesiano
        function dibujarPunto(x, y) {
            ctx.beginPath();
              ctx.arc(x, y, 3, 0, 2 * Math.PI);  // Tama□o reducido del
punto
            ctx.fillStyle = 'blue';
            ctx.fill();
        }

        // Funci□n para dibujar una l□nea entre dos puntos
        function dibujarLinea(p1, p2) {
            ctx.beginPath();
            ctx.moveTo(p1.x, p1.y);
            ctx.lineTo(p2.x, p2.y);
            ctx.strokeStyle = 'green';
            ctx.stroke();
        }

        // Funci□n para calcular la distancia entre dos puntos
        function calcularDistancia() {
            if (punto1 && punto2) {
                ctx.clearRect(0, 0, canvas.width, canvas.height);
                dibujarPunto(punto1.x, punto1.y);
                dibujarPunto(punto2.x, punto2.y);
```

```
                dibujarLinea(punto1, punto2);
                        var distancia = Math.sqrt(Math.pow(punto2.x - punto1.x, 2) + Math.pow(punto2.y - punto1.y, 2));
                mostrarDistancia(distancia.toFixed(2));
            } else {
                    alert('Por favor, ingresa coordenadas v□lidas para ambos puntos.');
            }
        }

        // Funci□n para mostrar la distancia en el lienzo
        function mostrarDistancia(distancia) {
            var x = (punto1.x + punto2.x) / 2;
            var y = (punto1.y + punto2.y) / 2;
            ctx.font = "20px Arial";
            ctx.fillStyle = "black";
            ctx.fillText("Distancia: " + distancia, x, y - 10);
        }
        // Manejar el evento del clic izquierdo del mouse en el lienzo
        canvas.addEventListener('click', function(event) {
            var rect = canvas.getBoundingClientRect();
                var mouseX = Math.round(event.clientX - rect.left); // Redondear las coordenadas
                var mouseY = Math.round(event.clientY - rect.top);  // Redondear las coordenadas

            if (!punto1) {
                punto1 = { x: mouseX, y: mouseY };
                dibujarPunto(mouseX, mouseY);
                  document.getElementById('punto1').value = mouseX + ', ' + mouseY;
            } else if (!punto2) {
                punto2 = { x: mouseX, y: mouseY };
                dibujarPunto(mouseX, mouseY);
                  document.getElementById('punto2').value = mouseX + ', ' + mouseY;
            }
```

```
        });

    </script>
</body>
</html>
```

RESULTADO



IMPLEMENTACIÓN EN 3D

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Representaci&#65533;n 3D de Espacios M&#65533;tricos</title>
    <style>
        body {
            display: flex;
            justify-content: center;
            align-items: center;
```

```
            height: 100vh;
            margin: 0;
            font-family: Arial, sans-serif;
        }
        #container {
            text-align: center;
        }
        canvas {
            border: 1px solid black;
        }
    </style>
</head>
<body>
    <div id="container">
        <h1>Representaci&#65533;n 3D de Espacios M&#65533;tricos</h1>
        <canvas id="myCanvas" width="500" height="500"></canvas>
        <div>
                <label for="coordenadas">Coordenadas Punto 1 (x, y, z):</label>
            <input type="text" id="punto1" placeholder="x1, y1, z1">
            <br>
                <label for="coordenadas">Coordenadas Punto 2 (x, y, z):</label>
            <input type="text" id="punto2" placeholder="x2, y2, z2">
        </div>
                    <button onclick="calcularDistancia()">Calcular Distancia</button>
    </div>

    <script>
        // Obtener el lienzo y el contexto 2D
        var canvas = document.getElementById('myCanvas');
        var ctx = canvas.getContext('2d');

            // Variables para almacenar los puntos ingresados por el usuario
```

```
        var punto1 = null;
        var punto2 = null;

        // Funci□n para dibujar un punto en el plano cartesiano
        function dibujarPunto(x, y, z) {
            if (x !== 0 || y !== 0) { // Verificar si las coordenadas son diferentes de (0,0)
                ctx.beginPath();
                ctx.arc(x, y, 5, 0, 2 * Math.PI);
                ctx.fillStyle = 'blue';
                ctx.fill();
            }
        }

        // Funci□n para dibujar una l□nea entre dos puntos
        function dibujarLinea(p1, p2) {
            ctx.beginPath();
            ctx.moveTo(p1.x, p1.y);
            ctx.lineTo(p2.x, p2.y);
            ctx.strokeStyle = 'green';
            ctx.stroke();
               // Mostrar la distancia encima de la l□nea verde si es mayor que 0
                var distancia = Math.sqrt(Math.pow(p2.x - p1.x, 2) + Math.pow(p2.y - p1.y, 2));
            if (distancia > 0) {
                var x = (p1.x + p2.x) / 2;
                var y = (p1.y + p2.y) / 2;
                ctx.fillStyle = 'black';
                    ctx.fillText(distancia.toFixed(2), x, y - 5); // Ajusta la posici□n del texto
            }
        }

        // Funci□n para calcular la distancia entre dos puntos en 3D
        function calcularDistancia() {
```

```
            if (punto1 && punto2) {

                // Calcular la distancia en 3D

                        var distancia = Math.sqrt(Math.pow(punto2.x -
punto1.x, 2) + Math.pow(punto2.y - punto1.y, 2) + Math.pow(punto2.z -
punto1.z, 2));



                // Dibujar la l□nea diagonal en 3D

                dibujarLinea(punto1, punto2);

                    dibujarLinea(punto1, {x: punto1.x, y: punto1.y, z:
punto2.z}); // L□nea en la dimensi□n Z

                    dibujarLinea(punto2, {x: punto2.x, y: punto2.y, z:
punto2.z}); // L□nea en la dimensi□n Z



                  // alert('La distancia entre los dos puntos en 3D es:
' + distancia.toFixed(2));

            } else {

                    alert('Por favor, ingresa coordenadas v□lidas para
ambos puntos.');

            }

        }



        // Manejar el evento del clic izquierdo del mouse en el lienzo

        canvas.addEventListener('click', function(event) {

            var rect = canvas.getBoundingClientRect();

            var mouseX = event.clientX - rect.left;

            var mouseY = event.clientY - rect.top;



            if (!punto1) {

                                punto1 = { x: mouseX, y: mouseY, z:
parseInt(prompt("Ingrese la coordenada Z para el punto 1")) || 0 }; //
Z-coordinate ser□ 0 si no se ingresa

                    dibujarPunto(mouseX, mouseY, punto1.z); // Pasa la
coordenada z

                document.getElementById('punto1').value = mouseX + ',
' + mouseY + ', ' + punto1.z;

            } else if (!punto2) {

                                punto2 = { x: mouseX, y: mouseY, z:
parseInt(prompt("Ingrese la coordenada Z para el punto 2")) || 0 }; //
Z-coordinate ser□ 0 si no se ingresa

                    dibujarPunto(mouseX, mouseY, punto2.z); // Pasa la
coordenada z
```

```
                document.getElementById('punto2').value = mouseX + ',
' + mouseY + ', ' + punto2.z;

            }

        });

    </script>

</body>

</html>
```

RESULTADO