



**UNIVERSIDAD NACIONAL
AUTÓNOMA DE MÉXICO**

**Facultad de Estudios Superiores
Aragón**

Cruz Ortiz Erick Jair
Maldonado Arrieta Johan Uriel
Tena Rodríguez Edson Alejandro
Zamudio Reyes Juan Pablo

Graficación por Computadoras
Salgado Rodríguez José Francisco

Fecha de entrega:
10 de Marzo de 2024

SCRIPT botonesnavegacion.cs

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;

public class botonesnavegacion : MonoBehaviour
{
    public void BotonInicio()
    {
        SceneManager.LoadScene("Programa");
    }

    public void Menu()
    {
        SceneManager.LoadScene("Titulo");
    }

    public void Info()
    {
        SceneManager.LoadScene("Informacion");
    }
}
```

SCRIPT conicas.cs

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;

public class conicas : MonoBehaviour
{
    public Text txtnombre;
    public Slider sl_a, sl_b, sl_h, sl_k, sl_t;
    public Text lba, lbb, lbh, lbk, lbt;
    public Material matLinea, matCircunferencia, matElipse, matParabola,
    matHiperbola;

    private int conica = 0;
    private int resolucion = 1000;
    private Vector3[] pospunto;
    private float a, b, h, k, t;
    void Start()
    {
        muestraSliderEtiqueta();
    }

    public void DibujaConicas()
    {
        if (conica != 0)
        {
            LineRenderer lr = GetComponent<LineRenderer>();
            #pragma warning disable CS0618 // El tipo o el miembro están obsoletos
            lr.SetVertexCount(resolucion + 1);

            a = sl_a.value;
            b = sl_b.value;
            h = sl_h.value;
            k = sl_k.value;
            t = sl_t.value;

            switch (conica)
            {
                case 1:
                    txtnombre.text = "Recta";
                    lr.material = matLinea;
                    muestraSliderEtiqueta();
                    lba.text = "ax";
                    lbb.text = "ay";
                    lbh.text = "bx";
                    lbk.text = "by";
                    sl_t.gameObject.SetActive(false);
                    lbt.gameObject.SetActive(false);
                    pospunto = CreaRecta(a, b, h, k, resolucion);
                    break;
```

```

        case 2:
            txtnombre.text = "Circunferencia";
            lr.material = matCircunferencia;
            muestraSliderEtiqueta();
            lba.gameObject.SetActive(false);
            sl_a.gameObject.SetActive(false);
            sl_t.gameObject.SetActive(false);
            lbt.gameObject.SetActive(false);
            lbb.text = "r";
            pospunto = crearCircunferencia(b, h, k, resolution);
            break;

        case 3:
            txtnombre.text = "Elipse";
            muestraSliderEtiqueta();
            pospunto = crearElipse(a, b, h, k, t, resolution);
            break;

        case 4:
            txtnombre.text = "Parábola";
            lr.material = matParabola;
            muestraSliderEtiqueta();
            lba.gameObject.SetActive(false);
            sl_a.gameObject.SetActive(false);
            lbb.text = "p";
            pospunto = CrearParabola(b, h, k, t, resolution);
            break;

        case 5:
            txtnombre.text = "Hipérbola";
            muestraSliderEtiqueta();
            pospunto = CrearHiperbola(a,b,h,k,t,resolucion);
            break;
    }

    for (int i = 0; i <= resolution; i++)
    {
        lr.SetPosition(i, pospunto[i]);
    }
}

public void btnRecta() {
    conica = 1;
    DibujaConicas();
}

Vector3[] CreaRecta(float ax, float ay, float bx, float by, int resolution)
{
    pospunto = new Vector3[resolution + 1];
    float dx = bx - ax;
    float dy = by - ay;

    for (int i = 0; i <= resolution; i++) {
        pospunto[i] = new Vector3(ax + dx * i / resolution, ay + dy * i /
resolution, 0);
    }
}

```

```

        return pospunto;
    }

    public void btnCircunferencia() {
        conica = 2;
        DibujaConicas();
    }

    Vector3[] crearCircunferencia(float r, float h, float k, int resolucion)
    {
        pospunto = new Vector3[resolucion + 1];
        Vector3 centro = new Vector3(h, k, 0);

        for (int i = 0; i <= resolucion; i++) {

            float angulo = ((float)i / (float)resolucion) * 2.0f * Mathf.PI;
            pospunto[i] = new Vector3(r * Mathf.Cos(angulo), r * Mathf.Sin(angulo),
0);
            pospunto[i] = pospunto[i] + centro;
        }

        return pospunto;
    }

    public void btnEllipse() {
        conica = 3;
        DibujaConicas();
    }

    Vector3[] crearEllipse(float a, float b, float h, float k, float t, int
resolucion)
    {
        pospunto = new Vector3[resolucion + 1];
        Vector3 centro = new Vector3(h, k, 0);
        for (int i = 0; i <= resolucion; i++)
        {
            float angulo = ((float)i / (float)resolucion) * 2.0f * Mathf.PI;
            pospunto[i] = new Vector3(h + a * Mathf.Cos(angulo) * Mathf.Cos(t) - b *
Mathf.Sin(angulo) * Mathf.Sin(t), k + a * Mathf.Cos(angulo) * Mathf.Sin(t) + b *
Mathf.Sin(angulo) * Mathf.Cos(t), 0);
        }

        return pospunto;
    }

    public void btnParabola(){
        conica = 4;
        DibujaConicas();
    }

    Vector3[] CrearParabola(float p, float h, float k, float theta, int resolucion)
    {
        pospunto = new Vector3[resolucion + 1];
        Quaternion q = Quaternion.AngleAxis(theta, Vector3.forward);
        Vector3 vertice = new Vector3(h, k, 0);

        for (int i = 0; i <= resolucion; i++)
        {

```

```

        float angulo = ((float)i / (float)resolucion) * 2.0f * Mathf.PI;
        pospunto[i] = new Vector3(i-(resolucion/2), (1/(4*p))*Mathf.Pow(i -
(resolucion/2),2),0);
        pospunto[i] = q * pospunto[i] + vertice;
    }

    return pospunto;
}
public void btnHiperbola(){
    conica = 5;
    DibujaConicas();
}
Vector3[] CrearHiperbola(float a, float b, float h, float k, float t, int
resolucion)
{
    pospunto = new Vector3[resolucion + 1];

    for (int i = 0; i <= resolucion; i++)
    {
        float theta = ((float)i / (float)resolucion) * Mathf.PI * 2.0f;
        float x = h + a / Mathf.Cos(theta) * Mathf.Cos(t) - b * Mathf.Tan(theta)
* Mathf.Sin(t);
        float y = k + a / Mathf.Cos(theta) * Mathf.Sin(t) + b * Mathf.Tan(theta)
* Mathf.Cos(t);
        pospunto[i] = new Vector3(x, y, 0);
    }

    return pospunto;
}

public void muestraSliderEtiqueta()
{
    sl_a.gameObject.SetActive(true);
    sl_b.gameObject.SetActive(true);
    sl_h.gameObject.SetActive(true);
    sl_k.gameObject.SetActive(true);
    sl_t.gameObject.SetActive(true);

    lba.gameObject.SetActive(true);
    lbb.gameObject.SetActive(true);
    lbh.gameObject.SetActive(true);
    lbk.gameObject.SetActive(true);
    lbt.gameObject.SetActive(true);

    lba.text = "a";
    lbb.text = "b";
    lbh.text = "h";
    lbk.text = "k";
    lbt.text = "t";
}
}

```

