```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Threading;
using System.Windows.Forms;

namespace SimplePendulum_Linear_NonLinear_All_Cases
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void idealToolStripMenuItem_Click(object sender, EventArgs e)
        {
            int size = 1000;                                                    //ideal linear by
euler method
            double[] th = new double[size];
            double[] w = new double[size];
            double[] t = new double[size];
            double g = 9.8, l = 1, dt = 0.04;
            th[0] = 2;
            w[0] = 5;
            for (int i = 0; i < size - 1; i++)
            {

                w[i + 1] = w[i] - (g / l) * th[i] * dt;
                th[i + 1] = th[i] + w[i] * dt;
                t[i + 1] = t[i] + dt;
            }
            Graphics gg = CreateGraphics();
            SolidBrush sb = new SolidBrush(Color.BlueViolet);
            Point O = new Point(150, 350);
            DrawAxes(O, 100, "t", "th");

            Application.DoEvents();
            Thread.Sleep(200);
            for (int i = 0; i < size; i++)
            {
                gg.FillEllipse(sb, O.X + (float)t[i] * 25, O.Y - (float)th[i] * 15, 5,
5);
            }
        }
        private void DrawAxes(Point O, int intercept, String xint, String yint)
        {
            Point p1 = new Point(O.X - intercept, O.Y);
            Point p2 = new Point(O.X + intercept, O.Y);
            Point p3 = new Point(O.X, O.Y - intercept);
            Point p4 = new Point(O.X, O.Y + intercept);
            Graphics gg = CreateGraphics();
            Pen pp = new Pen(Color.CornflowerBlue);
```

```
            SolidBrush sb = new SolidBrush(Color.Black);
            gg.DrawLine(pp, p1, p2);
            gg.DrawLine(pp, p3, p4);
            Font f = new Font("Arial", 12);
            gg.DrawString(xint, f, sb, O.X + 40, O.Y + 5);
            gg.DrawString(yint, f, sb, O.X - 40, O.Y - 50);
        }

        private void idealToolStripMenuItem1_Click(object sender, EventArgs e)
        {
            int size = 1500;                                    //ideal linear by euler
cromer..........click from nonlinear ideal
            double[] th = new double[size];
            double[] w = new double[size];
            double[] t = new double[size];
            double g = 9.8, l = 1, dt = 0.04;
            th[0] = 2;
            w[0] = 5;
            for (int i = 0; i < size - 1; i++)
            {

                w[i + 1] = w[i] - (g / l)*th[i] * dt;
                th[i + 1] = th[i] + w[i+1] * dt;
                t[i + 1] = t[i] + dt;
            }
            Graphics gg = CreateGraphics();
            SolidBrush sb = new SolidBrush(Color.BlueViolet);
            Point O = new Point(200, 250);
            DrawAxes(O, 200, "t", "th");

            Application.DoEvents();
            Thread.Sleep(200);
            for (int i = 0; i < size; i++)
            {
                gg.FillEllipse(sb, O.X + (float)t[i] * 25, O.Y- (float)th[i] * 15, 5, 5);
            }
        }

        private void dampedToolStripMenuItem_Click(object sender, EventArgs e)
        {
            int size = 1500;                                                //linear damped by
euler cromer for different values of q
            double[] th = new double[size];
            double[] w = new double[size];
            double[] t = new double[size];
            double q,g = 9.8, l = 1, dt = 0.04;
            th[0] = 2;
            w[0] = 5;
            q = double.Parse(textBox1.Text);
            for (int i = 0; i < size - 1; i++)
            {

                w[i + 1] = w[i] - (g / l) * th[i] * dt-q*w[i]*dt;
                th[i + 1] = th[i] + w[i+1] * dt;
                t[i + 1] = t[i] + dt;
            }
            Graphics gg = CreateGraphics();
            SolidBrush sb = new SolidBrush(Color.LightBlue);
```

```
            Point O = new Point(150, 150);
            DrawAxes(O, 200, "t", "th");

            Application.DoEvents();
            Thread.Sleep(200);
            for (int i = 0; i < size; i++)
            {
                gg.FillEllipse(sb, O.X + (float)t[i] * 25, O.Y - (float)th[i] * 15, 5,
5);
            }
        }

        private void dampedToolStripMenuItem1_Click(object sender, EventArgs e)
        {
            int size = 1500;                          //comparison between two different
values of theta.....click from non linear damped button
            double[] th = new double[size];
            double[] w = new double[size];
            double[] t = new double[size];
            double  g = 9.8, l = 1, dt = 0.01;

            w[0] = 2;

            th[0]= double.Parse(textBox3.Text);
            for (int i = 0; i <th.Length - 1; i++)
            {

                w[i + 1] = w[i] - (g / l) * Math.Sin(th[i]) * dt ;
                th[i + 1] = th[i] + w[i + 1] * dt;
                t[i + 1] = t[i] + dt;
            }
            Graphics gg = CreateGraphics();
            SolidBrush sb = new SolidBrush(Color.Goldenrod);
            Point O = new Point(350, 150);
            DrawAxes(O, 300, "t", "th");

            Application.DoEvents();
            Thread.Sleep(200);
            for (int i = 0; i < size; i++)
            {
                gg.FillEllipse(sb, O.X + (float)t[i] * 25, O.Y -(float)th[i] * 15, 5, 5);
            }
        }

        private void dampedDrivenToolStripMenuItem_Click(object sender, EventArgs e)
        {
            int size = 10000;                                                  //linear damped
driven by euler cromer
            double[] th = new double[size];
            double[] w = new double[size];
            double[] t = new double[size];
            double q,Fd, g = 9.8, l = 1, dt = 0.04,omega=2.0;
            th[0] = 2;
            w[0] = 5;
            t[0] = 0;
            q = double.Parse(textBox1.Text);
           Fd= double.Parse(textBox2.Text);
```

```
            for (int i = 0; i <th.Length - 1; i++)
            {

                w[i + 1] = w[i] - (g / l) *th[i] * dt - q * w[i] *
dt+Fd*Math.Sin(omega*t[i])*dt;
                th[i + 1] = th[i] + w[i + 1] * dt;
                t[i + 1] = t[i] + dt;
            }
            Graphics gg = CreateGraphics();
            SolidBrush sb = new SolidBrush(Color.DarkRed);
            Point O = new Point(250,250);
            DrawAxes(O, 300, "t", "th");

            Application.DoEvents();
            Thread.Sleep(200);
            for (int i = 0; i < size; i++)
            {
                gg.FillEllipse(sb, O.X + (float)t[i] * 10, O.Y - (float)th[i] * 155,5,5);
            }
        }

        private void dampedDrivenToolStripMenuItem1_Click(object sender, EventArgs e)
        {
            int size = 1500;                                                    //non linear
damped driven by cromer method in range -pi to pi
            double[] th = new double[size];
            double[] w = new double[size];
            double[] t = new double[size];
            double q, Fd, g = 9.8, l = 9.8, dt = 0.04, omega = 2.0/3.0;
            th[0] = 0.2;
            w[0] = 0;
            t[0] = 0;
            q = 1.0/2.0;
            Fd = double.Parse(textBox2.Text);
            for (int i = 0; i < th.Length - 1; i++)
            {

                w[i + 1] = w[i] - (g / l) *Math.Sin( th[i]) * dt - q * w[i] * dt + Fd *
Math.Sin(omega * t[i]) * dt;
                th[i + 1] = th[i] + w[i + 1] * dt;
                if (th[i + 1] < -Math.PI)
                {
                    th[i + 1] = th[i + 1] + 2 * Math.PI;
                }
                if (th[i + 1] > Math.PI)
                {
                    th[i + 1] = th[i + 1] - 2 * Math.PI;
                }
                t[i + 1] = t[i] + dt;
            }
            Graphics gg = CreateGraphics();
            SolidBrush sb = new SolidBrush(Color.CadetBlue);
            Point O = new Point(370, 300);
            DrawAxes(O, 300, "t", "th");

            Application.DoEvents();
            Thread.Sleep(200);
            for (int i = 0; i < size; i++)
```

```
            {
                gg.FillEllipse(sb, O.X + (float)t[i] * 5, O.Y - (float)th[i] * 40, 5, 5);
            }
        }

    }
}
```