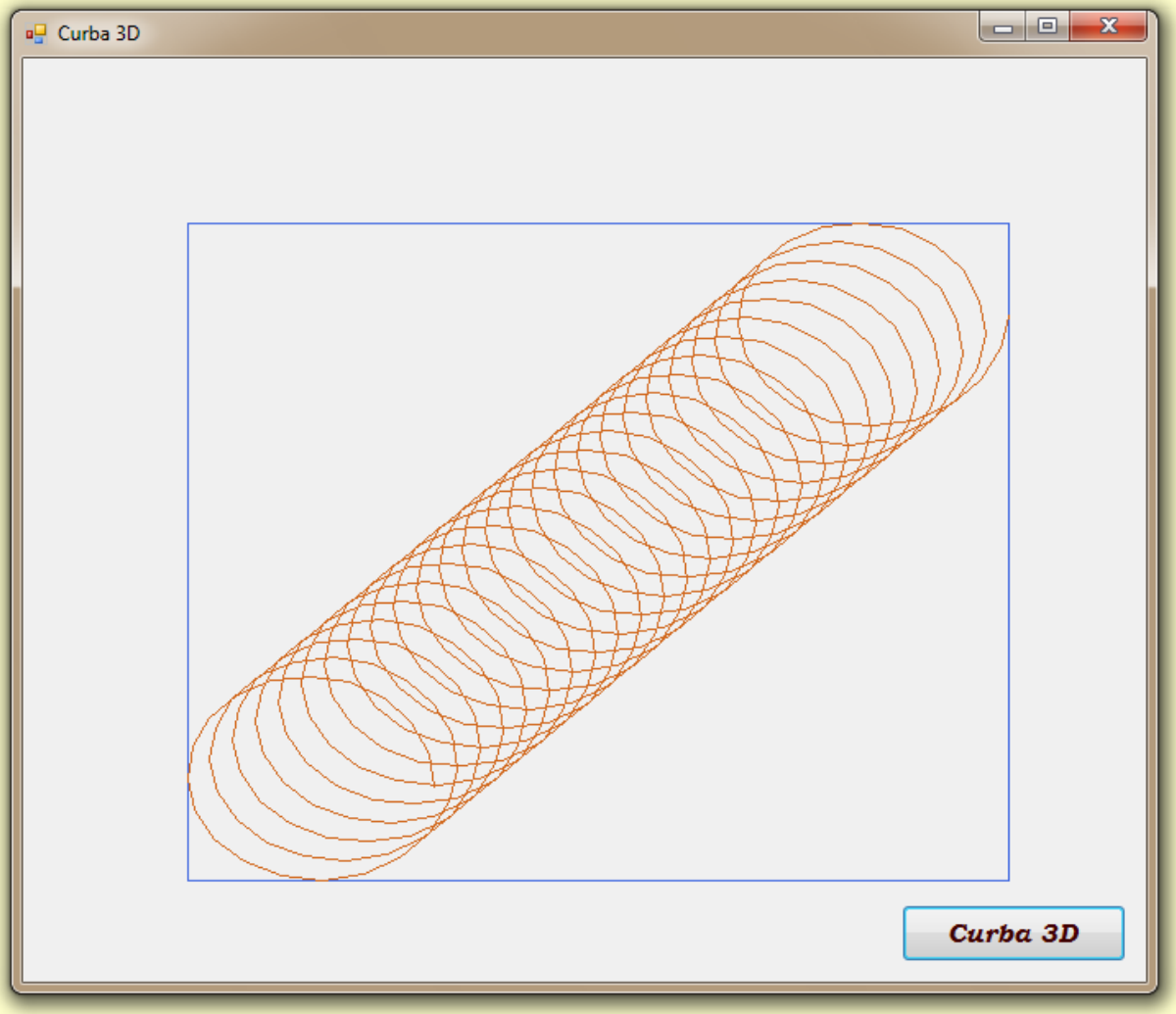
```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace Curba_3D
{
   public partial class Form1 : Form
   {

      double x (double t) { return Math.Cos(t); }        // x=f(t); y=g(t); z=h(t)
      double y (double t) { return Math.Sin(t); }
      double z (double t) { return       t/25; }

      int    u1, v1, u2, v2;           // ViewPort
      double a,  b,  c,  d;            // Window
      double Raza, Alfa;               // Pr. Par.
      int    Lu, Lv;                   // Lpr;

      int u(double x) { return (int)((x - a) / (b - a) * (u2 - u1) + u1); }
      int v(double y) { return (int)((y - d) / (c - d) * (v2 - v1) + v1); }

      void ViewPort(int    x1, int    y1, int    x2, int    y2) { u1 = x1; v1 = y1; u2 = x2; v2 = y2; }
      void Window  (double x1, double y1, double x2, double y2) { a  = x1; d  = y1; b  = x2;  c = y2; }

      void DefPr   (double r, double a) { Raza = r; Alfa = a; }  //  r=1; a=0.8;  // = Pi/4  \\

      double PrX  (double x, double z)   { return x+Raza*z*Math.Cos(Alfa); }
      double PrY  (double y, double z)   { return y+Raza*z*Math.Sin(Alfa); }

      void MoveTo(int u1, int v1) { Lu = u1; Lv = v1; }
      void LineTo  (int u1, int v1, System.Drawing.Graphics Gr, System.Drawing.Pen Pen)
                                    {  Gr.DrawLine(Pen, Lu, Lv, u1, v1); Lu = u1; Lv = v1; }

      public Form1()
      {
         InitializeComponent();
      }
```

```
    private void button1_Click(object sender, EventArgs e)  // Curba 3D
    {
       System.Drawing.Graphics Drept;
       Drept = this.CreateGraphics();
       Pen myPen = new Pen(System.Drawing.Color.RoyalBlue);
       Rectangle myRectangle = new Rectangle(100, 100, 500, 400);
       Drept.DrawRectangle(myPen, myRectangle);

       myPen = new System.Drawing.Pen(System.Drawing.Color.Chocolate);
       System.Drawing.Graphics formGraphics = this.CreateGraphics();
       ViewPort (100, 100, 600, 500);
       DefPr(1, 3.14/4);   int n=500;

       double t1=0, t2=50*3.1416;                          //     Domeniul de definiție

       double a=PrX(x(t1),z(t1)), b=PrX(x(t1),z(t1)),     //  Determinarea ferestrei reale
              c=PrY(y(t1),z(t1)), d=PrY(y(t1),z(t1));
       for (int i=1; i<=n; i++)
       {
          double t=(t2-t1)/n*i+t1;
          double Xp=PrX(x(t),z(t)), Yp=PrY(y(t),z(t));
          if (Xp<a) a=Xp;  else if (Xp>b) b=Xp;
          if (Yp<c) c=Yp;  else if (Yp>d) d=Yp;
       }
       Window  (a,d, b,c);
       MoveTo(u(PrX(x(t1),z(t1))),v(PrY(y(t1),z(t1))));       // Desenarea Curbei
       for (int i=1; i<=n; i++)
       {
          double t=(t2-t1)/n*i+t1;
          LineTo(u(PrX(x(t),z(t))),v(PrY(y(t),z(t))),formGraphics, myPen);
       }
    }
  }
}
```


Curba 3D
Curba 3D