**Московский авиационный институт**

**(Национальный исследовательский университет)**

Факультет №8 «Информационные технологии и прикладная математика»

Кафедра 806 «Вычислительной математики и программирования»

**Лабораторная работа № 1**

по курсу «Компьютерная графика»

Тема: Построение изображений 2D-кривых.

Студент: Сорокин Д.М.

Группа: 80-304Б

Преподаватель: Филлипов Г.С.

Оценка:

Москва, 2017

Задание:

Написать и отладить программу, строящую изображение заданной замечательной кривой. Обеспечить автоматическое масштабирование и центрирование кривой при изменении размеров окна.

Вариант:

15. ρ = (a\*cos2ϕ)/cos ϕ -π<A<=ϕ<=B<π

Код:

**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** lab1

{

**public** **partial** **class** Form1 : Form

{

**int** MAX = 1000000;

**float** scale = 1;

Point offset;

Point zero;

**double** pi = Math.PI;

**private** Point CalcCentr()

{

**return** new Point(drawBox.Width / 2, drawBox.Height / 2);

}

**public** Form1()

{

InitializeComponent();

zero = CalcCentr();

drawBox.MouseWheel += DrawBox\_MouseWheel;

}

**bool** drawcurve = **false**;

**private** **double** myFunc(**double** a, **double** t)

{

**return** a \* Math.Cos(2\*t) / Math.Cos(t);*/\* > MAX ? MAX : a \* Math.Cos(2 \* t) / Math.Cos(t)\*/*

}

Graphics g;

**private** **void** drawBox\_Paint(**object** sender, PaintEventArgs e)

{

g = e.Graphics;

g.ScaleTransform(scale,scale);

g.TranslateTransform(zero.X + offset.X, zero.Y + offset.Y);

DrawCoordinates(g, Pens.Red);

**if** (drawcurve) DrawCurve(g);

}

**private** **void** DrawCurve(Graphics g)

{

Pen linePen = new Pen(Color.Blue, 1);

**double** aBorder;

**double** bBorder;

**double** paramA;

**double** step;

**try**

{

aBorder = Convert.ToDouble(textBox1.Text);

bBorder = Convert.ToDouble(textBox2.Text);

paramA = Convert.ToDouble(textBox3.Text);

step = Convert.ToDouble(textBox4.Text);

}

**catch** (Exception)

{

MessageBox.Show("Неверный формат ввода параметров");

**return**;

}

step /= 1000;

**var** points = new List<PointF>();

**for** (**double** i = aBorder; i < bBorder; i += step)

{

**var** r = myFunc(paramA, i);

PointF begin = ToDecart(r, i);

*//PointF end = ToDecart(r, i + step);*

points.**Add**(begin);

*//g.DrawLine(linePen, begin, end);*

}

**if** (points.Count < 2)

{

MessageBox.Show("Задан слишком большой шаг");

**return**;

}

**var** ARR = points.ToArray();

**var** last = ARR[0];

**for** (**int** i = 1; i < ARR.Length; i++)

{

**if** (ARR[i].Y - last.Y > 10000\*step)

{

last = ARR[i];

**continue**;

}

g.DrawLine(linePen, last, ARR[i]);

last = ARR[i];

}

*//g.DrawLines(linePen, points.ToArray());*

*//g.DrawCurve(linePen, points.ToArray());*

}

**private** PointF ToDecart(**double** r, **double** d)

{

**float** x = (**float**)(r \* Math.Cos(d));

**if** (Math.Abs(x) < 0.001) x = 0;

**if** (Math.Abs(x) > 100000) x = 100000;

**float** y = (**float**)(r \* Math.Sin(d));

**if** (Math.Abs(y) < 0.001) y = 0;

**if** (Math.Abs(y) > 1000)

{

**if** (y >= 0)

y = 1000;

**else**

y = -1000;

}

**return** new PointF(x, y);

}

**private** **void** DrawCoordinates(Graphics g, Pen p)

{

g.DrawLine(p, new Point(-MAX, 0), new Point(MAX, 0));

g.DrawLine(p, new Point(0, -MAX), new Point(0, MAX));

**for** (**int** i = -1000; i < 1000; i += 10)

{

g.DrawLine(p, new Point(i, -2), new Point(i, 2));

**if** (i % 100 == 0) g.DrawString((i / 10).ToString(), DefaultFont, Brushes.Green, new Point(i, 10));

g.DrawLine(p, new Point(-2, i), new Point(2, i));

}

}

**private** **void** DrawBox\_MouseWheel(**object** sender, MouseEventArgs e)

{

**if** ( e.Delta > 0)

scale \*= 1.1f;

**else**

{

scale \*= 0.9f;

}

drawBox.Invalidate();

}

**private** **void** drawBox\_SizeChanged(**object** sender, EventArgs e)

{

zero = CalcCentr();

}

**private** **void** button1\_Click\_1(**object** sender, EventArgs e)

{

scale = 1;

drawcurve = !drawcurve;

offset = new Point();

drawBox.Invalidate();

}

**bool** mousePress = **false**;

Point mousePressPoint;

Point oldoffset;

**private** **void** drawBox\_MouseDown(**object** sender, MouseEventArgs e)

{

mousePress = **true**;

mousePressPoint = e.Location;

oldoffset = new Point(offset.X, offset.Y);

}

**private** **void** drawBox\_MouseUp(**object** sender, MouseEventArgs e)

{

mousePress = **false**;

*//mousePressPoint = new Point();*

}

**private** **void** drawBox\_MouseMove(**object** sender, MouseEventArgs e)

{

**if** (mousePress)

{

**int** dx = e.X - mousePressPoint.X;

**int** dy = e.Y - mousePressPoint.Y;

offset = new Point(oldoffset.X + (**int**)(dx/scale), oldoffset.Y + (**int**)(dy/scale));

drawBox.Invalidate();

}

}

**private** **void** textBox1\_TextChanged\_1(**object** sender, EventArgs e)

{

drawcurve = **false**;

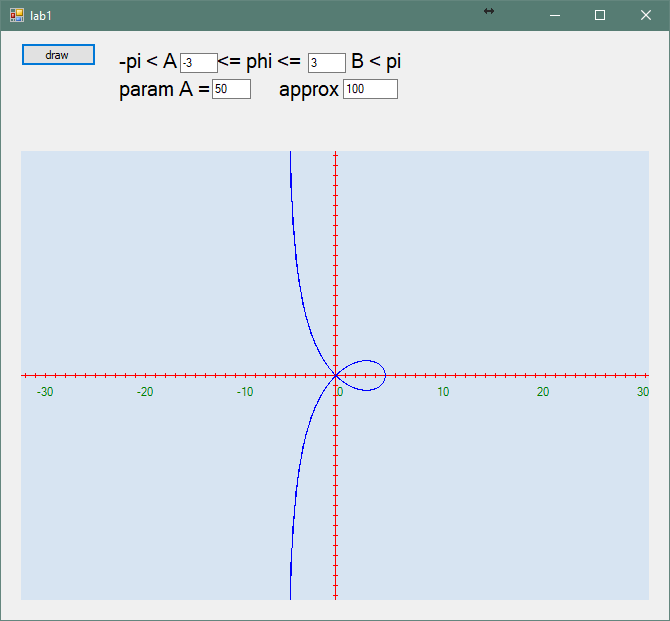
drawBox.Invalidate();

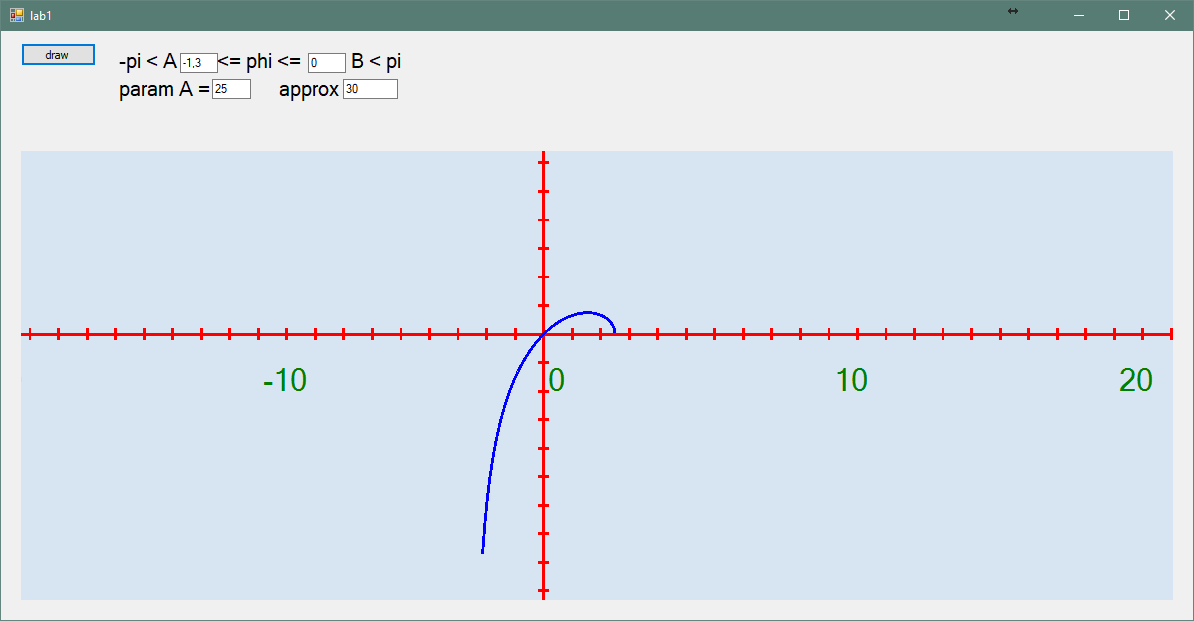
}

}

}

Примеры работы программы:





Выводы:

Выполнив данную лабораторную работу, я приобрел навыки в практических применениях знаний основ компьютерной графики. Я освоил базовые умения создания изображений с помощью языка С#.