

Московский авиационный институт
(Национальный исследовательский университет)

**Институт «Информационных технологий и прикладной
математики»**

Лабораторная работа №1
Построение изображений 2D-кривых

Работу выполнил:
Рябыкин Алексей Сергеевич
Группа: М8О-309Б-18

Преподаватель: Филиппов Г.С.
Оценка:
Дата:

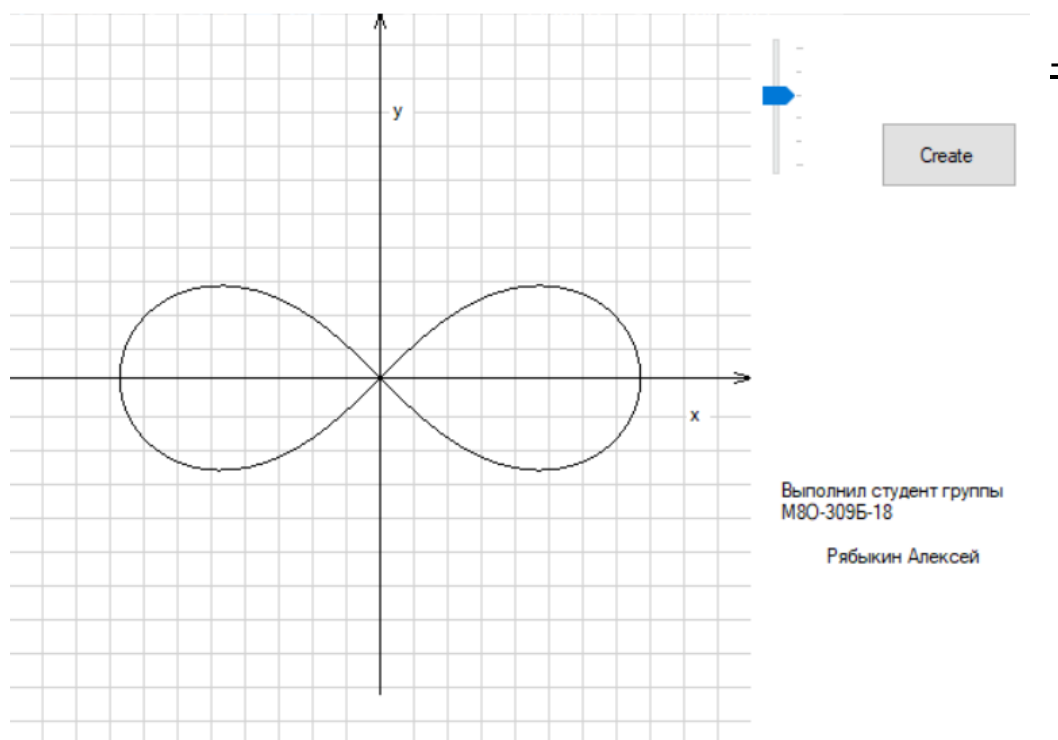
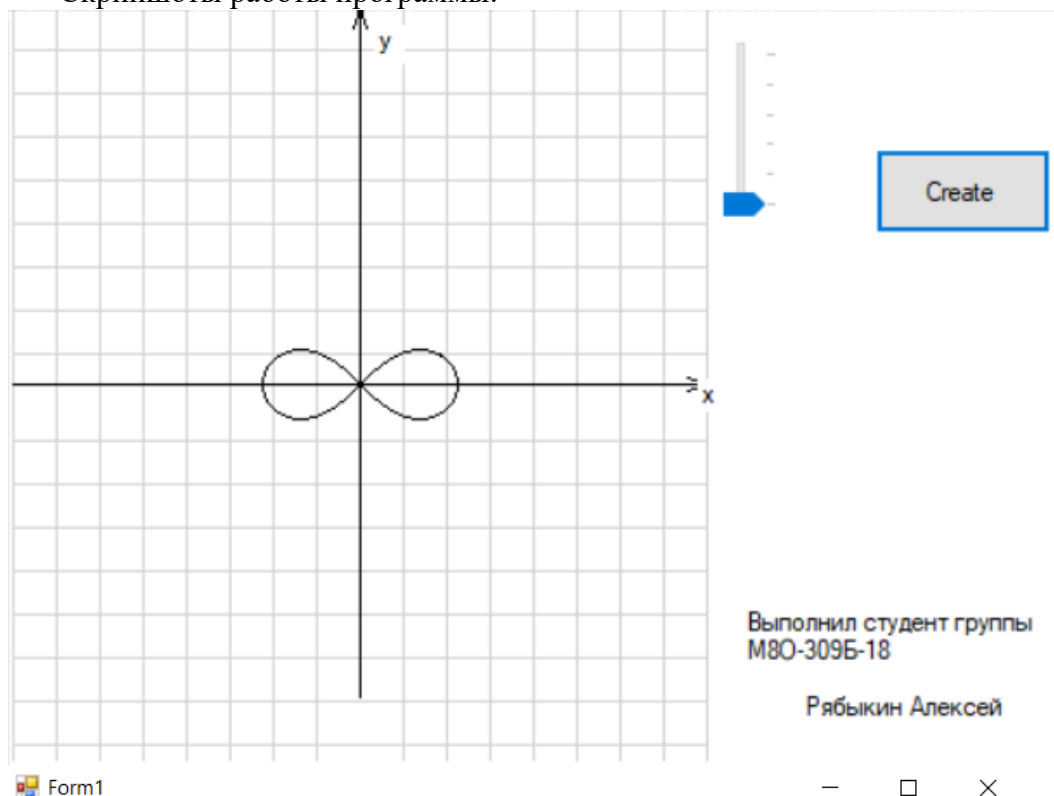
Москва, 2020

Постановка задачи

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

Вариант 4: $\rho^2 = a^2 \cos 2\varphi$ ρ, φ - полярные координаты

Скриншоты работы программы:



Фрагменты кода:

Функция Solve, создающая массивы точек:

```
private void Solve()
{
    a = 10;
    while (phi <= (float)Math.PI)
    {
        R = (float)Math.Sqrt(a * a * (float)Math.Cos(2 * phi));
        X.Add(R * (float)Math.Cos(phi));
        Y.Add(R * (float)Math.Sin(phi));
        phi += 0.01f;
    }
}
```

Функция Draw, выполняющая отрисовку:

```
private void Draw()
{
    for (float l = 0; l <= screen.Width; l += 20)
    {
        if (Math.Abs(l - screen.Width / 2) > 10)
            Graphics.FromHwnd(screen.Handle).DrawLine(os, l, 0, l,
screen.Height);
    }
    for (float l = 0; l <= screen.Height; l += 20)
    {
        if (Math.Abs(l - screen.Height / 2) > 10)
            Graphics.FromHwnd(screen.Handle).DrawLine(os, 0, l, screen.Width, l);
    }
    osx.Visible = true;
    Point x = new Point(screen.Height - 30, screen.Width / 2 + 10);
    osx.Location = x;
    osy.Visible = true;
    k = (float)Size.Value / 4;
    Graphics.FromHwnd(screen.Handle).DrawLine(penx, screen.Width / 2, 0,
screen.Width / 2, screen.Height - 30);
    Graphics.FromHwnd(screen.Handle).DrawLine(penx, 0, screen.Height / 2,
screen.Width, screen.Height / 2);
    Graphics.FromHwnd(screen.Handle).DrawLine(penx, screen.Width / 2, 0,
screen.Width / 2 + 3, 10);
    Graphics.FromHwnd(screen.Handle).DrawLine(penx, screen.Width / 2, 0,
screen.Width / 2 - 3, 10);
    Graphics.FromHwnd(screen.Handle).DrawLine(penx, screen.Width, screen.Height /
2, screen.Width - 10, screen.Height / 2 - 3);
    Graphics.FromHwnd(screen.Handle).DrawLine(penx, screen.Width, screen.Height /
2, screen.Width - 10, screen.Height / 2 + 3);
    k1 = (float)screen.Width / 428;
    Pen pen = new Pen(Color.Black);
    for (int i = 0; i < X.Count() - 1; i++)
    {
        Graphics.FromHwnd(screen.Handle).DrawLine(pen, k1 * k * X[i] * 120 / a +
screen.Width / 2,
            k1 * k * Y[i] * 120 / a + screen.Height / 2, k1 * k * X[i + 1] * 120
/ a + screen.Width / 2,
            k1 * k * Y[i + 1] * 120 / a + screen.Height / 2);
        Graphics.FromHwnd(screen.Handle).DrawLine(pen, k1 * k * X[i] * 120 / a +
screen.Width / 2,
            k1 * k * -1 * Y[i] * 120 / a + screen.Height / 2, k1 * k * X[i + 1] *
120 / a + screen.Width / 2,
            k1 * k * -1 * Y[i + 1] * 120 / a + screen.Height / 2);
    }
}
```

Среда разработки: Microsoft Visual Studio 2019

Вывод: В процессе выполнения лабораторной работы научился отрисовывать, масштабировать и центрировать при изменении окна 2D-кривые различной кривизны.