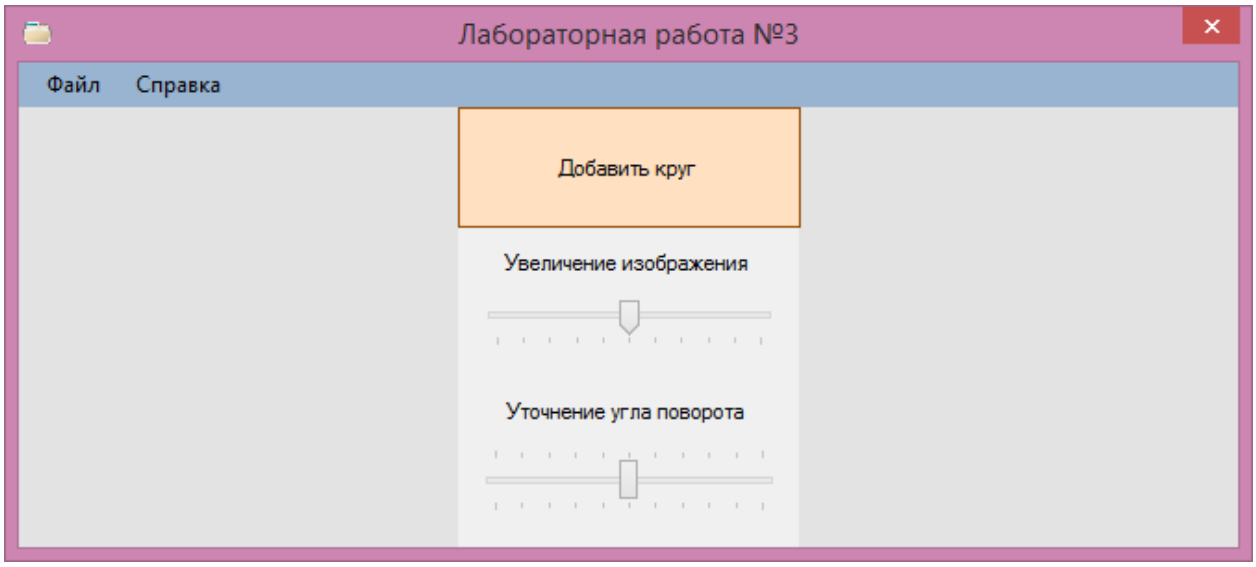
Загрузить из файла изображение. На вторую форму вывести преобразованное изображение. В качестве преобразований использовать:

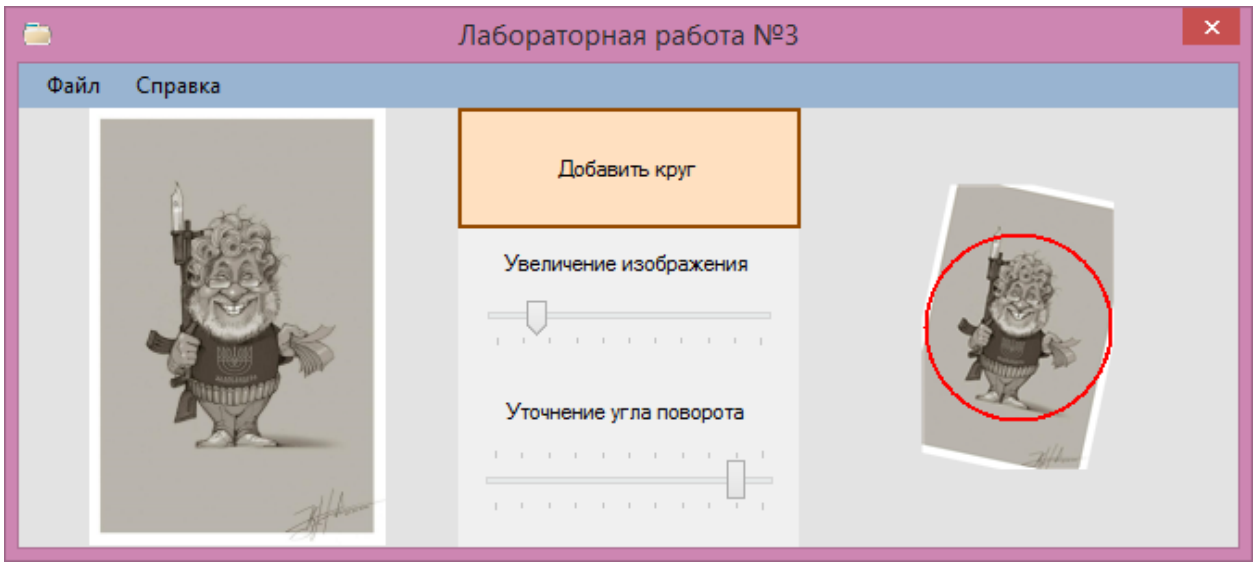
a. Поворот;
b. Масштаб;
c. В центре преобразованного изображения поместить окружность радиуса 100.

```
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;
using System.IO;
using System.Drawing.Drawing2D;

namespace SHLb_3
{
    public partial class ImageEditor : Form
    {
        private Image img;
        private int degree = 0;
        private int size = 100;

        public ImageEditor()
        {
            InitializeComponent();
        }

        private void OpenImageMenuItem_Click(object sender, EventArgs e)
        {
            if (OpenImage.ShowDialog() == DialogResult.OK)
            {
                string fileName = OpenImage.FileName;
                FileStream stream = File.Open(fileName, FileMode.Open, FileAccess.Read);
                if (stream != null)
                {
                    img = Image.FromStream(stream);
                    Original.Image = img;
                    this.DrawImageFromImg();
                    stream.Close();
                }
            }
        }

        private void Rotate_Scroll(object sender, EventArgs e)
        {
            this.degree = Rotate.Value;
            this.DrawImageFromImg();
        }

        private void Modernize_MouseClick(object sender, MouseEventArgs e)
        {
            if (this.img != null)
            {
                if (e.Button == MouseButtons.Left)
```

```
            {
                this.img.RotateFlip(RotateFlipType.Rotate270FlipNone);
            }
            else if (e.Button == MouseButtons.Right)
            {
                this.img.RotateFlip(RotateFlipType.Rotate90FlipNone);
            }
            this.DrawImageFromImg();
        }
        else
        {
            OpenImageMenuItem_Click(sender, e);
        }
    }

    private void Original_Click(object sender, EventArgs e)
    {
        if (Modernize.Image == null)
        {
            OpenImageMenuItem_Click(sender, e);
        }
        else
        {
            this.img = Original.Image;
            Zoom.Value = 100;
            size = 100;
            Rotate.Value = 0;
            degree = 0;
            DrawImageFromImg();
        }
    }

    private void DrawImageFromImg()
    {
        if(img != null)
        {
            Image temp = img;
            if (this.degree != 0)
            {
                temp = this.rotateImage((Bitmap)temp, this.degree);
            }

            int sizeWidth = (Modernize.Width * this.size) / 100;
            int sizeHeight = (Modernize.Height * this.size) / 100;
            temp = this.ResizeImg(temp, sizeWidth, sizeHeight);

            Modernize.Image = temp;
            Modernize.Invalidate();
        }
    }

    private Bitmap rotateImage(Bitmap b, float angle)
    {
        Bitmap returnBitmap = new Bitmap(b.Width, b.Height);
        returnBitmap.SetResolution(b.HorizontalResolution, b.VerticalResolution);
        Graphics g = Graphics.FromImage(returnBitmap);
        g.TranslateTransform((float)b.Width / 2, (float)b.Height / 2);
        g.RotateTransform(angle);
        g.TranslateTransform(-(float)b.Width / 2, -(float)b.Height / 2);
        g.DrawImage(b, new Point(0, 0));
        return returnBitmap;
    }

    public Image ResizeImg(Image imgToResize, int nWidth, int nHeight)
    {
```

```
            int sourceWidth = imgToResize.Width;
            int sourceHeight = imgToResize.Height;

            float nPercent = 0;
            float nPercentW = 0;
            float nPercentH = 0;

            nPercentW = ((float)nWidth / (float)sourceWidth);
            nPercentH = ((float)nHeight / (float)sourceHeight);

            if (nPercentH < nPercentW)
                nPercent = nPercentH;
            else
                nPercent = nPercentW;

            int destWidth = (int)(sourceWidth * nPercent);
            int destHeight = (int)(sourceHeight * nPercent);

            Bitmap b = new Bitmap(destWidth, destHeight);
            Graphics g = Graphics.FromImage((Image)b);
            g.InterpolationMode = InterpolationMode.HighQualityBicubic;

            g.DrawImage(imgToResize, 0, 0, destWidth, destHeight);
            g.Dispose();

            return (Image)b;
        }

        private void Zoom_Scroll(object sender, EventArgs e)
        {
            this.size = Zoom.Value;
            this.DrawImageFromImg();
        }

        private void Circle_Click(object sender, EventArgs e)
        {
            if(img != null)
            {
                 Graphics g = Graphics.FromImage(Modernize.Image);
                g.DrawEllipse(new Pen(Color.Red, 2), new Rectangle(Modernize.Image.Width
/ 2 - 50, Modernize.Image.Height / 2 - 50, 100, 100));
                Modernize.Invalidate();
            }
        }

        private void ExitMenuItem_Click(object sender, EventArgs e)
        {
            Application.Exit();
        }
    }
}
```


Лабораторная работа №3
Файл
Справка
Добавить круг
Увеличение изображения
Уточнение угла поворота


Лабораторная работа №3
Файл
Справка
Добавить круг
Увеличение изображения
Уточнение угла поворота


Лабораторная работа №3
Файл
Справка
Добавить круг
Увеличение изображения
Уточнение угла поворота