МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Вятский государственный университет»

Факультет автоматики и вычислительной техники Кафедра электронных вычислительных машин

АФИННЫЕ ПРЕОБРАЗОВАНИЯ НА ПЛОСКОСТИ

Отчет

Лабораторной работе №3 по дисциплине

«Компьютерная графика»

Выполнил студент группы ИВТб-21\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Жеребцов К. А. /

Проверил преподаватель\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Коржавина А.С./

2021 г.

**Цель:** закрепить лекционный материал по изучению одноименной темы, реализовав матрицы переноса, масштабирования, отражения и вращения применительно к координатам описанной в программе плоской фигуры (многоугольника) с целью демонстрации движения и преобразования формы этой фигуры на плоскости.

**Задание на лабораторную работу:**

Написать на языке PASCAL программу:

1. Рисующую многоугольник.

2. Смещающую его на n пикселов вправо и m вниз.

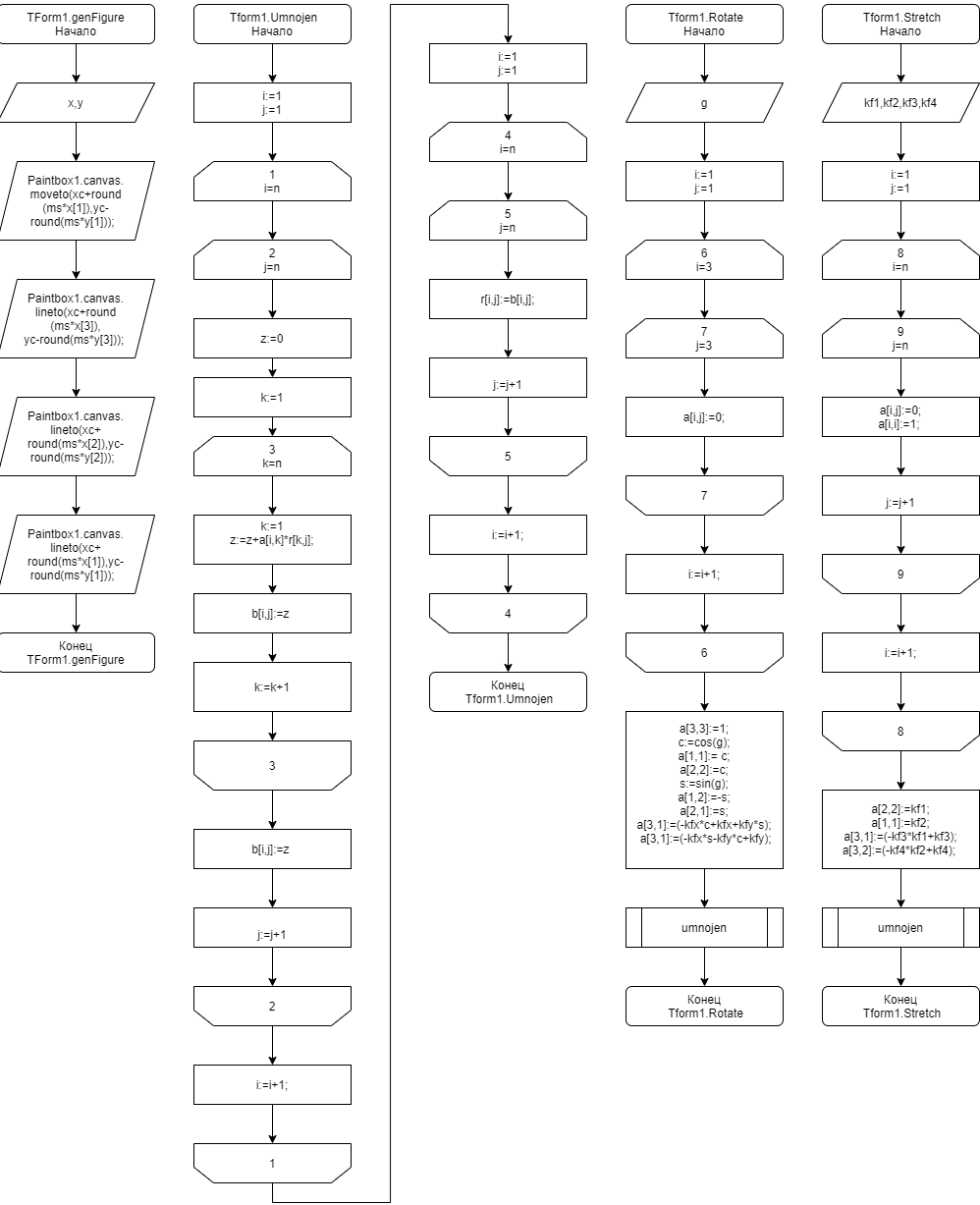
3. Зеркально отражающую его относительно осей координат.

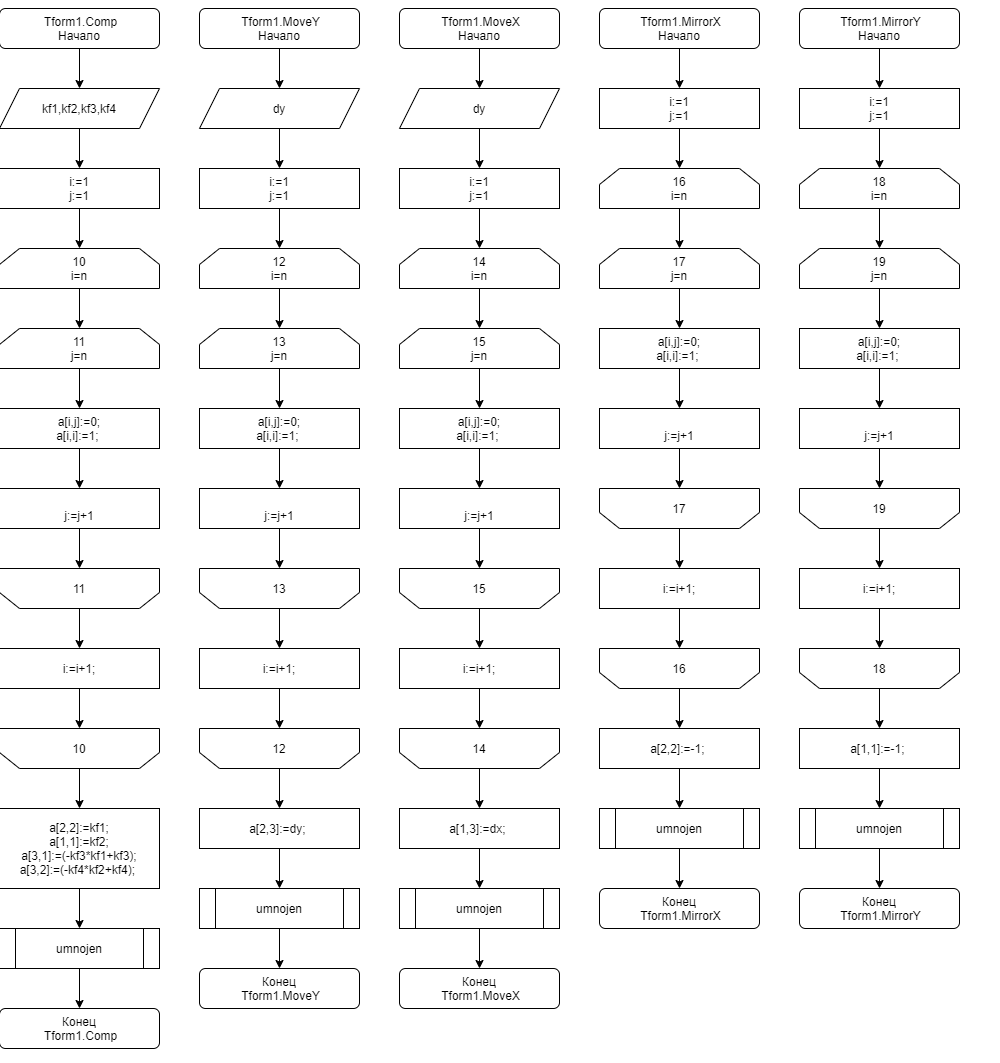
4. Растягивающую (сжимающую) его вдоль координатных осей относительно некоторой заданной точки.

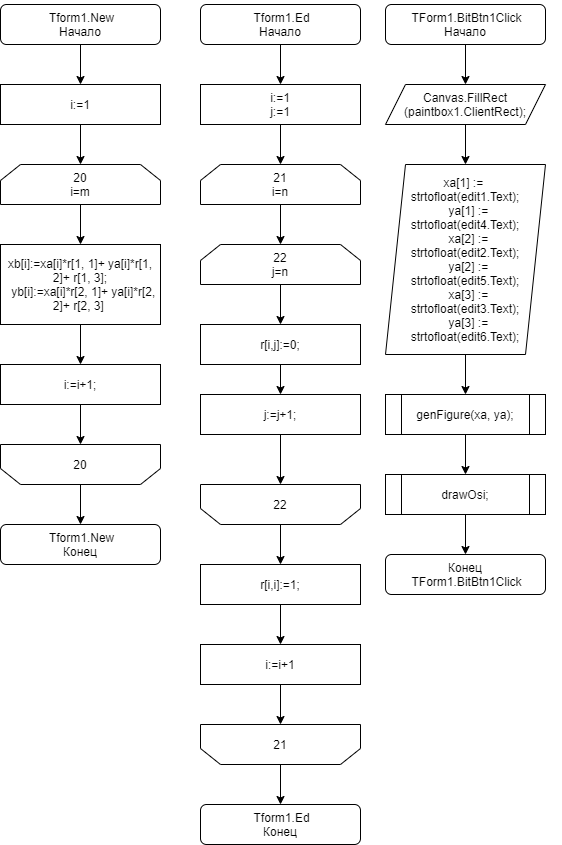
5. Вращающую его относительно центра с координатами (k,l) по часовой стрелке с увеличением размеров, против - с уменьшением.

6. Реализовать интерактивную анимацию (взаимодействующую с пользователем), на которой выполняется поворот и масштабирование объектов.

**Схемы алгоритмов:**

****

****



**Исходный код:**

unit Unit1;

{$mode objfpc}{$H+}

interface

uses

Classes, SysUtils, Forms, Controls, Graphics, Dialogs, ExtCtrls, Buttons,

StdCtrls;

const

m = 4;

n = 3;

kfst = 1.1;{коэффициент растяжения}

kfcm = 0.9; {коэффициент сжатия}

kfx = 960; {точка сжатия/растяжения/поворота}

kfy = 540;

type

mas = array[1..m] of real;

mtr = array[1..n, 1..n] of real;

{ TForm1 }

TForm1 = class(TForm)

BitBtn1: TBitBtn;

BitBtn10: TBitBtn;

BitBtn11: TBitBtn;

BitBtn12: TBitBtn;

BitBtn13: TBitBtn;

BitBtn2: TBitBtn;

BitBtn3: TBitBtn;

BitBtn4: TBitBtn;

BitBtn5: TBitBtn;

BitBtn6: TBitBtn;

BitBtn7: TBitBtn;

BitBtn8: TBitBtn;

BitBtn9: TBitBtn;

Edit1: TEdit;

Edit2: TEdit;

Edit3: TEdit;

Edit4: TEdit;

Edit5: TEdit;

Edit6: TEdit;

PaintBox1: TPaintBox;

procedure BitBtn10Click(Sender: TObject);

procedure BitBtn11Click(Sender: TObject);

procedure BitBtn12Click(Sender: TObject);

procedure BitBtn13Click(Sender: TObject);

procedure BitBtn1Click(Sender: TObject);

procedure BitBtn2Click(Sender: TObject);

procedure BitBtn4Click(Sender: TObject);

procedure BitBtn5Click(Sender: TObject);

procedure BitBtn6Click(Sender: TObject);

procedure BitBtn7Click(Sender: TObject);

procedure BitBtn8Click(Sender: TObject);

procedure BitBtn9Click(Sender: TObject);

procedure FormCreate(Sender: TObject);

procedure drawOsi;

procedure genFigure(x, y: mas);

procedure Umnojen;

procedure MoveY(dy: integer);

procedure New;

procedure Ed;

procedure BitBtn3Click(Sender: TObject);

procedure MoveX(dx: integer);

procedure MirrorX;

procedure MirrorY;

procedure comp(kf1, kf2, kf3, kf4: real);

procedure Stretch(kf1, kf2, kf3, kf4: real);

procedure Rotate(g: real);

private

public

end;

var

Form1: TForm1;

xa, ya, xb, yb: mas;{фигуры после перемещения}

a, r: mtr;{матрицы преобразований}

xc, yc: integer;

ms: real;

i, j, k: integer;

x: array[1..10] of integer;

y: array[1..10] of integer;

implementation

{$R \*.lfm}

{ TForm1 }

procedure TForm1.FormCreate(Sender: TObject);

begin

xc := 960;

yc := 540;

ms := 70;

end;

procedure Tform1.drawOsi;

begin

paintbox1.Canvas.Pen.Color := ClBlack;

paintbox1.Canvas.Pen.Width := 2;

Paintbox1.canvas.line(0, 540, 1920, 540);

Paintbox1.canvas.line(1920, 540, 1870, 530);

Paintbox1.canvas.line(1920, 540, 1870, 550);

Paintbox1.canvas.line(960, 0, 960, 1080);

Paintbox1.canvas.line(960, 0, 950, 50);

Paintbox1.canvas.line(960, 0, 970, 50);

end;

procedure TForm1.genFigure(x, y: mas);

begin

paintbox1.Canvas.Pen.Width := 2;

paintbox1.Canvas.Pen.Color := ClRed;

Paintbox1.canvas.moveto(xc + round(ms \* x[1]), yc - round(ms \* y[1]));

Paintbox1.canvas.lineto(xc + round(ms \* x[3]), yc - round(ms \* y[3]));

Paintbox1.canvas.lineto(xc + round(ms \* x[2]), yc - round(ms \* y[2]));

Paintbox1.canvas.lineto(xc + round(ms \* x[1]), yc - round(ms \* y[1]));

end;

procedure TForm1.BitBtn1Click(Sender: TObject);

begin

Canvas.FillRect(paintbox1.ClientRect);

{координаты вершин фигуры}

xa[1] := strtofloat(edit1.Text);

ya[1] := strtofloat(edit4.Text);

xa[2] := strtofloat(edit2.Text);

ya[2] := strtofloat(edit5.Text);

xa[3] := strtofloat(edit3.Text);

ya[3] := strtofloat(edit6.Text);

genFigure(xa, ya);

drawOsi;

end;

procedure TForm1.BitBtn3Click(Sender: TObject); {перемещение по Y}

begin

Canvas.FillRect(paintbox1.ClientRect);

drawOsi;

Ed;

MoveY(-1);

New;

genFigure(xb, yb);

xa := xb;

ya := yb;

end;

procedure TForm1.BitBtn2Click(Sender: TObject);

begin

Canvas.FillRect(paintbox1.ClientRect);

drawOsi;

Ed;

MoveY(1);

New;

genFigure(xb, yb);

xa := xb;

ya := yb;

end;

procedure TForm1.BitBtn4Click(Sender: TObject); {перемещение по X}

begin

Canvas.FillRect(paintbox1.ClientRect);

drawOsi;

Ed;

MoveX(-1);

New;

genFigure(xb, yb);

xa := xb;

ya := yb;

end;

procedure TForm1.BitBtn5Click(Sender: TObject);

begin

Canvas.FillRect(paintbox1.ClientRect);

drawOsi;

Ed;

MoveX(1);

New;

genFigure(xb, yb);

xa := xb;

ya := yb;

end;

procedure TForm1.BitBtn6Click(Sender: TObject); {отражение по Х}

begin

Canvas.FillRect(paintbox1.ClientRect);

drawOsi;

Ed;

MirrorX;

New;

genfigure(xb, yb);

xa := xb;

ya := yb;

end;

procedure TForm1.BitBtn7Click(Sender: TObject); {отражение по Y}

begin

Canvas.FillRect(paintbox1.ClientRect);

drawOsi;

Ed;

MirrorY;

New;

genfigure(xb, yb);

xa := xb;

ya := yb;

end;

procedure TForm1.BitBtn8Click(Sender: TObject); {растяжение}

begin

Canvas.FillRect(paintbox1.ClientRect);

drawOsi;

Ed;

Stretch(1, kfst, kfx, kfy);

New;

genfigure(xb, yb);

xa := xb;

ya := yb;

end;

procedure TForm1.BitBtn11Click(Sender: TObject);

begin

Canvas.FillRect(paintbox1.ClientRect);

drawOsi;

Ed;

Stretch(kfst, 1, kfy, kfx);

New;

genfigure(xb, yb);

xa := xb;

ya := yb;

end;

procedure TForm1.BitBtn12Click(Sender: TObject); {поворот по часовой}

begin

Canvas.FillRect(paintbox1.ClientRect);

drawOsi;

Ed;

Rotate(pi / 6);

New;

genfigure(xb, yb);

xa := xb;

ya := yb;

Canvas.FillRect(paintbox1.ClientRect);

drawOsi;

Ed;

Stretch(kfst, 1, 1, 1);

;

New;

genfigure(xb, yb);

xa := xb;

ya := yb;

end;

procedure TForm1.BitBtn13Click(Sender: TObject);

begin

Canvas.FillRect(paintbox1.ClientRect);

drawOsi;

Ed;

Rotate(11 \* pi / 6);

New;

genfigure(xb, yb);

xa := xb;

ya := yb;

Canvas.FillRect(paintbox1.ClientRect);

drawOsi;

Ed;

Stretch(kfcm, 1, 1, 1);

;

New;

genfigure(xb, yb);

xa := xb;

ya := yb;

end;

procedure TForm1.BitBtn9Click(Sender: TObject); {сжатие}

begin

Canvas.FillRect(paintbox1.ClientRect);

drawOsi;

Ed;

comp(1, kfcm, kfx, kfy);

New;

genfigure(xb, yb);

xa := xb;

ya := yb;

end;

procedure TForm1.BitBtn10Click(Sender: TObject);

begin

Canvas.FillRect(paintbox1.ClientRect);

drawOsi;

Ed;

comp(kfcm, 1, kfy, kfx);

New;

genfigure(xb, yb);

xa := xb;

ya := yb;

end;

procedure Tform1.Umnojen; {умножение матриц}

var

b: mtr;

z: real;

begin

for i := 1 to n do

for j := 1 to n do

begin

z := 0;

for k := 1 to n do

z := z + a[i, k] \* r[k, j];

b[i, j] := z;

end;

for i := 1 to n do

for j := 1 to n do

r[i, j] := b[i, j];

end;

procedure Tform1.Rotate(g: real);

var

c, s: real;

begin

for i := 1 to 3 do

for j := 1 to 3 do

a[i, j] := 0;

a[3, 3] := 1;

c := cos(g);

a[1, 1] := c;

a[2, 2] := c;

s := sin(g);

a[1, 2] := -s;

a[2, 1] := s;

a[3, 1] := (-kfx \* c + kfx + kfy \* s);

a[3, 1] := (-kfx \* s - kfy \* c + kfy);

umnojen;

end;

procedure Tform1.Stretch(kf1, kf2, kf3, kf4: real);

begin

for i := 1 to n do

begin

for j := 1 to n do

a[i, j] := 0;

a[i, i] := 1;

end;

a[2, 2] := kf1;

a[1, 1] := kf2;

a[3, 1] := (-kf3 \* kf1 + kf3);

a[3, 2] := (-kf4 \* kf2 + kf4);

umnojen;

end;

procedure Tform1.comp(kf1, kf2, kf3, kf4: real);

begin

for i := 1 to n do

begin

for j := 1 to n do

a[i, j] := 0;

a[i, i] := 1;

end;

a[2, 2] := kf1;

a[1, 1] := kf2;

a[3, 1] := (-kf3 \* kf1 + kf3);

a[3, 2] := (-kf4 \* kf2 + kf4);

umnojen;

end;

procedure Tform1.MoveY(dy: integer);

begin

for i := 1 to n do

begin

for j := 1 to n do

a[i, j] := 0;

a[i, i] := 1;

end;

a[2, 3] := dy;

umnojen;

end;

procedure Tform1.MoveX(dx: integer);

begin

for i := 1 to n do

begin

for j := 1 to n do

a[i, j] := 0;

a[i, i] := 1;

end;

a[1, 3] := dx;

umnojen;

end;

procedure Tform1.MirrorX;

begin

for i := 1 to n do

begin

for j := 1 to n do

a[i, j] := 0;

a[i, i] := 1;

end;

a[2, 2] := -1;

umnojen;

end;

procedure Tform1.MirrorY;

begin

for i := 1 to n do

begin

for j := 1 to n do

a[i, j] := 0;

a[i, i] := 1;

end;

a[1, 1] := -1;

umnojen;

end;

procedure Tform1.New;

begin

for i := 1 to m do

begin

xb[i] := xa[i] \* r[1, 1] + ya[i] \* r[1, 2] + r[1, 3];

yb[i] := xa[i] \* r[2, 1] + ya[i] \* r[2, 2] + r[2, 3];

end;

end;

procedure Tform1.Ed;

begin

for i := 1 to n do

begin

for j := 1 to n do

r[i, j] := 0;

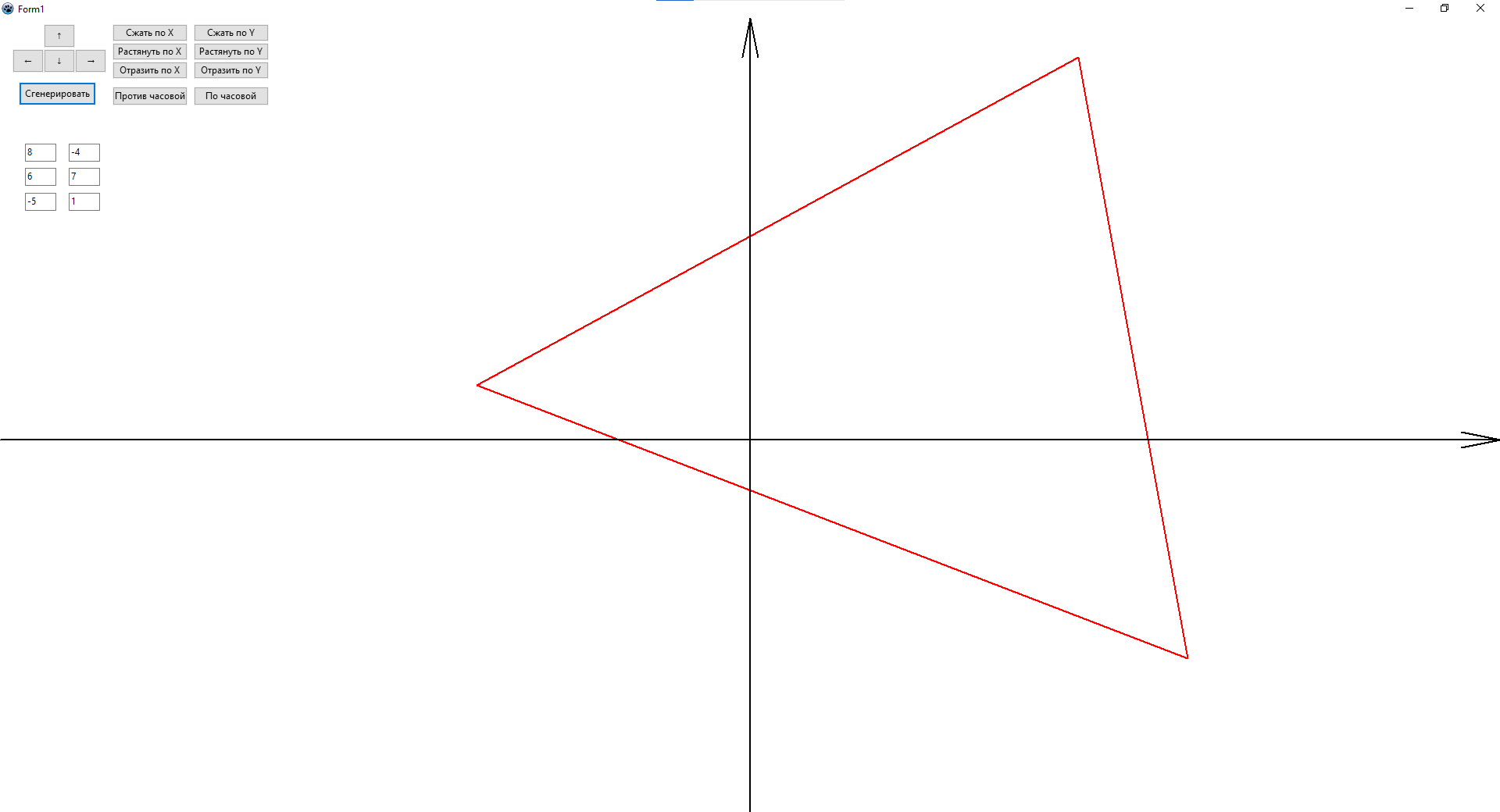
r[i, i] := 1;

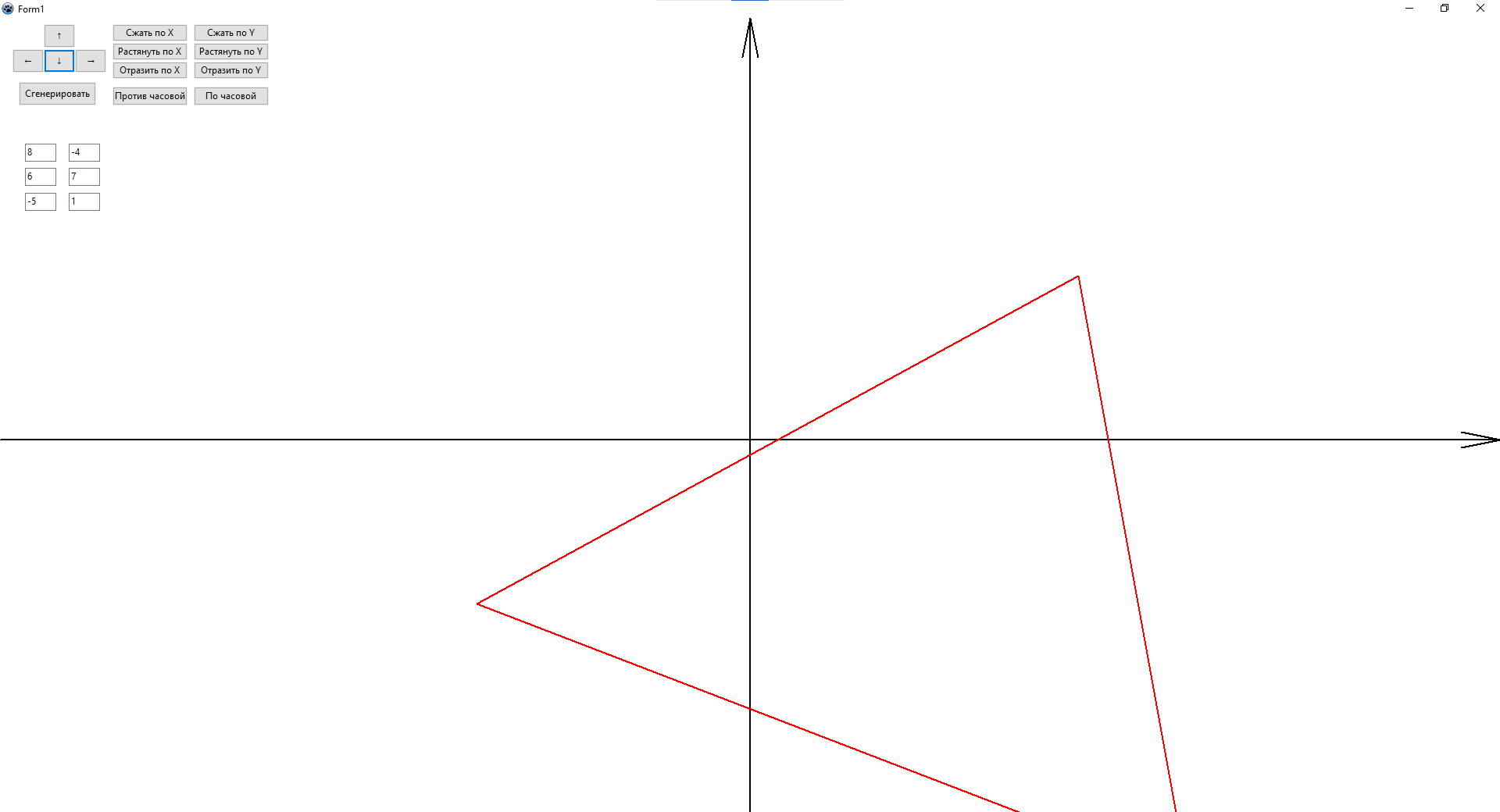
end;

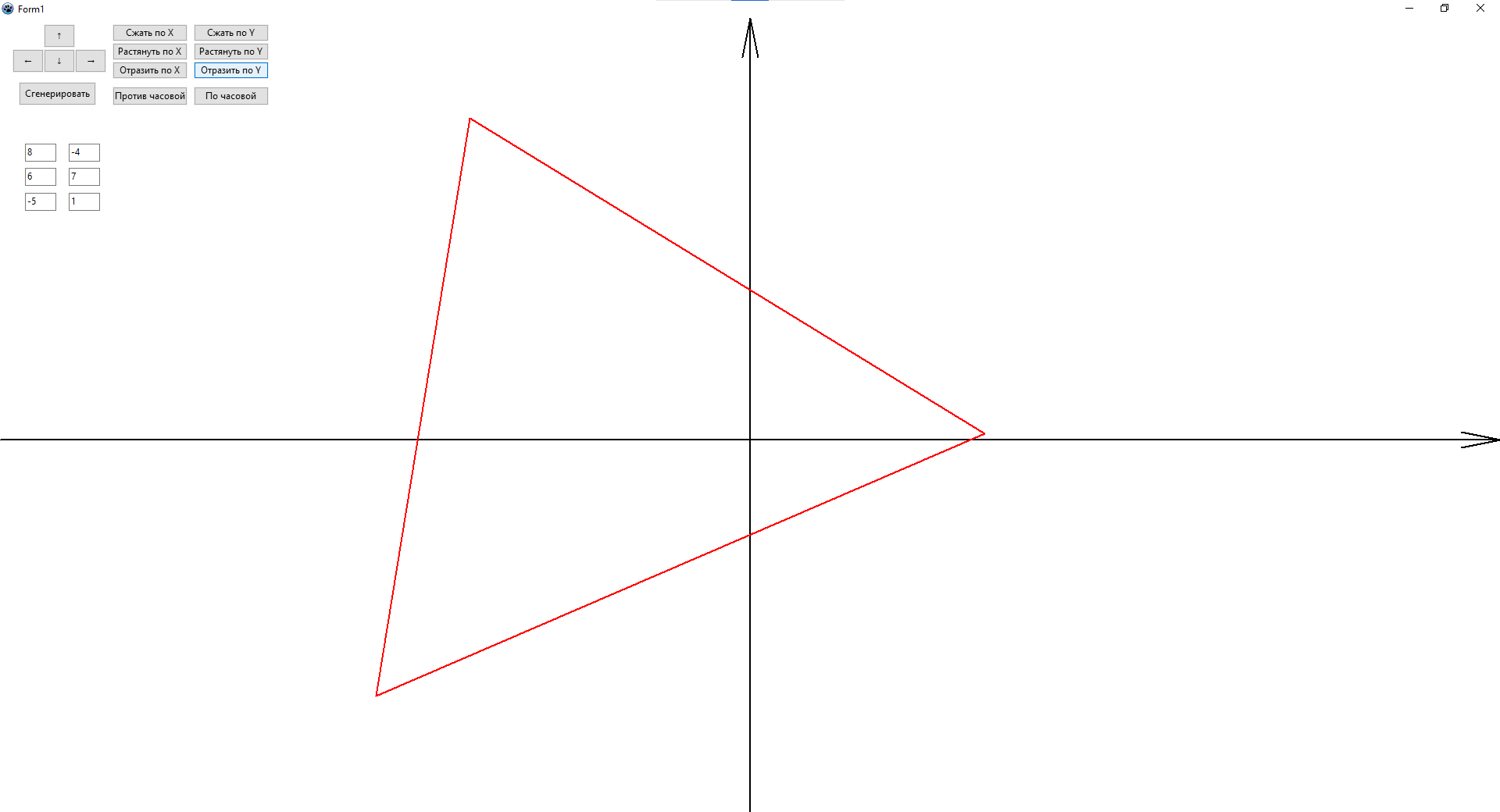
end;

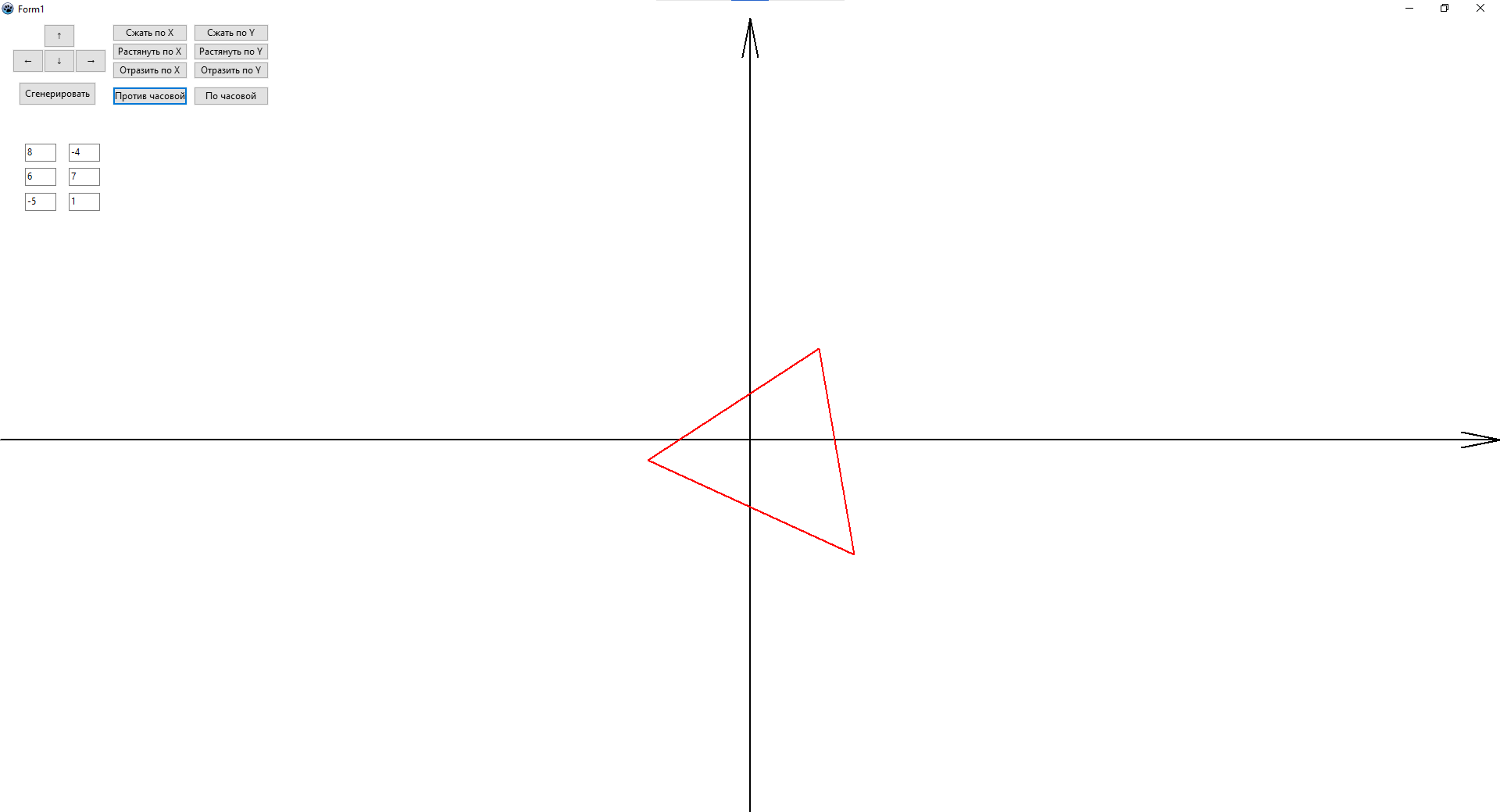
end.

**Экранные формы:**









**Вывод:**

Был закреплен лекционный материал по изучению одноименной темы, были реализованы матрицы переноса, масштабирования, отражения и вращения применительно к координатам описанной в программе плоской фигуры (многоугольника) с целью демонстрации движения и преобразования формы этой фигуры на плоскости.

Была написана на языке PASCAL программа, которая рисует многоугольник, смещает его на n пикселов вправо и m вниз, зеркально отражает его относительно осей координат, растягивает (сжимает) его вдоль координатных осей относительно некоторой заданной точки, вращает его относительно центра с координатами (k,l) по часовой стрелке с увеличением размеров, против - с уменьшением.