# РГПУ им. А. И. Герцена

Лабораторные работы №1-6

Выполнила: Шеховцова Е. Г.

Лабораторная работа №1

Цель: Знакомство с интегрированной средой разработки программного обеспечения Lazarus. Создание нового проекта. Знакомство с объектами формы Button, TextBox, Label, PictureBox.

Задача 1: Создать простейший пользовательский интерфейс, для программы сложения двух чисел (a + b = c), содержащий объекты Button, TextBox - для ввода значений переменных a и b и вывода значения с, Label - для поясняющих надписей

Задача 2: Написать процедуру обработки события - щелчок по кнопке - сложение двух чисел. Для этого необходимо создать процедуру обработки события (двойным щелчком по объекту кнопка), объявить переменные, ввести значение переменных a и b, вычислить значение переменной с и вывести значение с на экран.

Задача 3: Нарисовать блок схему с помощью сервиса Google – рисунки

Код:

var c, a, b :integer;

begin

a:=Strtoint(Edit1.Text);

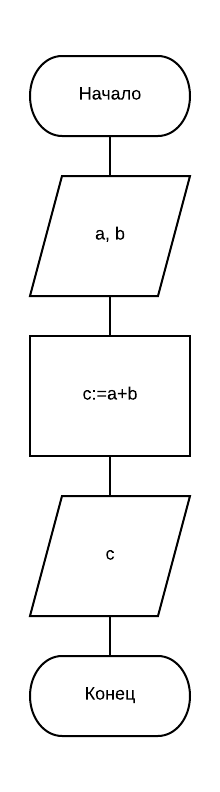
b:=Strtoint(Edit2.Text);

c:= a + b;

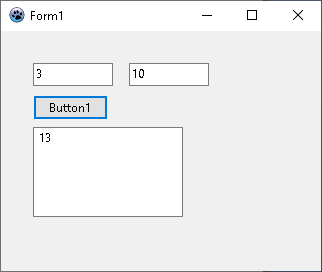
Memo1.Text:=inttostr(c);

end;

Блок-схема:



Скриншот:



Лабораторная работа №2

Задача 1: Нарисовать в объекте TImage треугольник, одна сторона которого черная, другая красная и третья - синяя.

Задача 2: Создать программу рисующую треугольник, координаты вершин которого можно изменять через пользовательский интерфейс.

Код:

var x,y,x1,y1,x2,y2:integer;

begin

x:=strtoint(Edit2.Text);

y:=strtoint(Edit1.Text);

x1:=strtoint(Edit3.Text);

y1:=strtoint(Edit4.Text);

x2:=strtoint(Edit5.Text);

y2:=strtoint(Edit6.Text);

with PaintBox1.Canvas do begin

Pen.Color := clRed;

MoveTo(x,y);

LineTo(x1,y1);

Pen.Color:=clBlue;

LineTo(x2,y2);

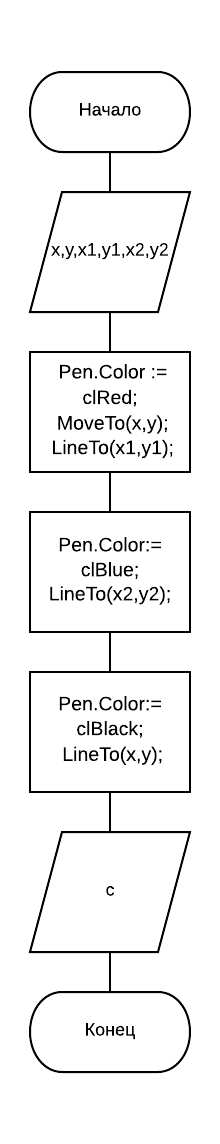
Pen.Color:=clBlack;

LineTo(x,y);

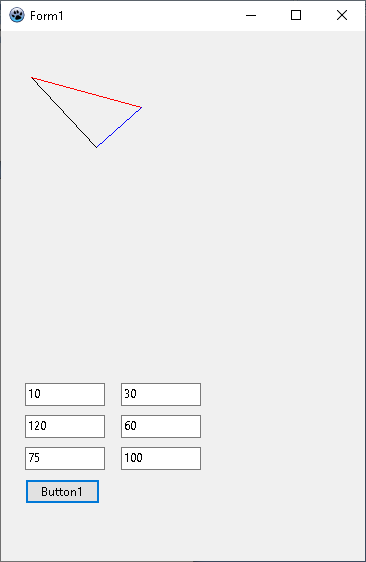
end;

end;

Блок-схема:



Скриншот:



Лабораторная работа №3

Цель: Разработать алгоритм и программу решения прикладной задачи – построение графика заданной функции. Научиться переводить декартовые координаты в экранные.

Задача : Построить график функции y = Ax^2 + Bx + C в диапазоне ( x min , y min ) – ( x max , y max ). Постоянные А, В, С, x min , y min , x max , y max вводятся через пользовательский интерфейс.

Математическая модель:

Блок-схемы:

Код:

var xmid, ymid, xmax, x, y, a, b, c, xmin, ymin: integer;

begin

a:=strtoint(Edit1.Text);

b:=strtoint(Edit2.Text);

c:=strtoint(Edit3.Text);

xmin:=strtoint(Edit4.Text);

ymin:=strtoint(Edit5.Text);

xmax:=strtoint(Edit6.Text);

with PaintBox1.Canvas do begin

xmid := PaintBox1.Width div 2;

ymid := PaintBox1.Height div 2;

MoveTo(0,ymid);

LineTo(PaintBox1.Width,ymid);

MoveTo(xmid,0);

LineTo(xmid,PaintBox1.Height);

x:=xmin;

y:=0;

MoveTo(xmin+xmid,-ymin+ymid);

Pen.Color:=clRed;

while (x<xmax) do

begin

y:=a\*x\*x+b\*x+c;

LineTo(x+xmid, -y+ymid);

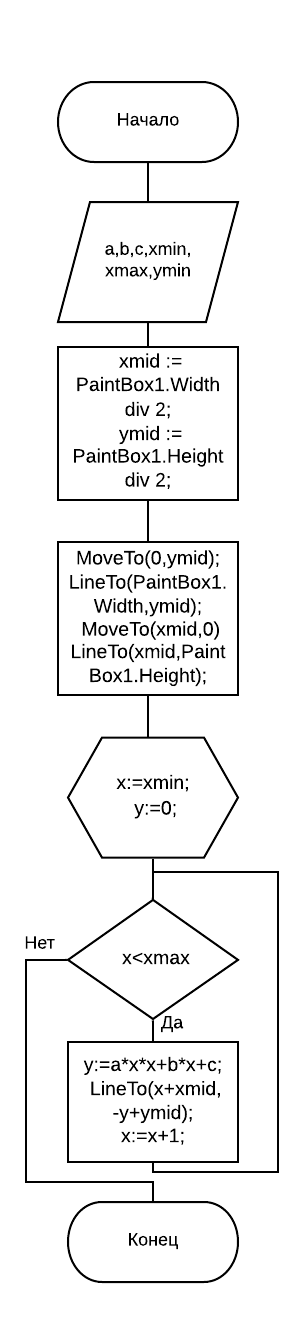
x:=x+1;

end;

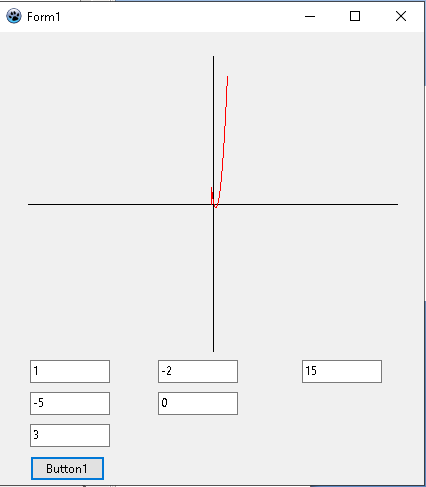
end;

end;

Блок-схема:



Скриншот:



Лабораторная работа №4

Цель: Разработать алгоритм и программу решения прикладной задачи – построение траектории движения тела брошенного под углом к горизонту. Научиться переводить мировые координаты в экранные.

Задача : Изобразить местонахождение тела, брошенного с земли с начальной скорость U0 под углом к горизонту α, через равные интервалы времени ( N – число равных временных интервалов)

Математическая модель:

Код:

var xmid, ymid, xmax, x, y, xmin, ymin: integer;

v0,alpha: real;

begin

alpha:=strtoint(Edit1.Text)\*3.14/180;

v0:=strtoint(Edit2.Text);

with PaintBox1.Canvas do begin

xmid := PaintBox1.Width div 2;

ymid := PaintBox1.Height div 2;

MoveTo(0,PaintBox1.Height-1);

LineTo(0,0);

MoveTo(0,PaintBox1.Height-1);

LineTo(PaintBox1.Width+200,PaintBox1.Height);

x:=0;

y:=0;

MoveTo(0,PaintBox1.Height);

Pen.Color:=clRed;

while (x<PaintBox1.Width) do

begin

y:=round(x\*sin(alpha)/cos(alpha)-10\*x\*x/(2\*v0\*v0\*cos(alpha)\*cos(alpha)));

LineTo(x,-y+PaintBox1.Height-1);

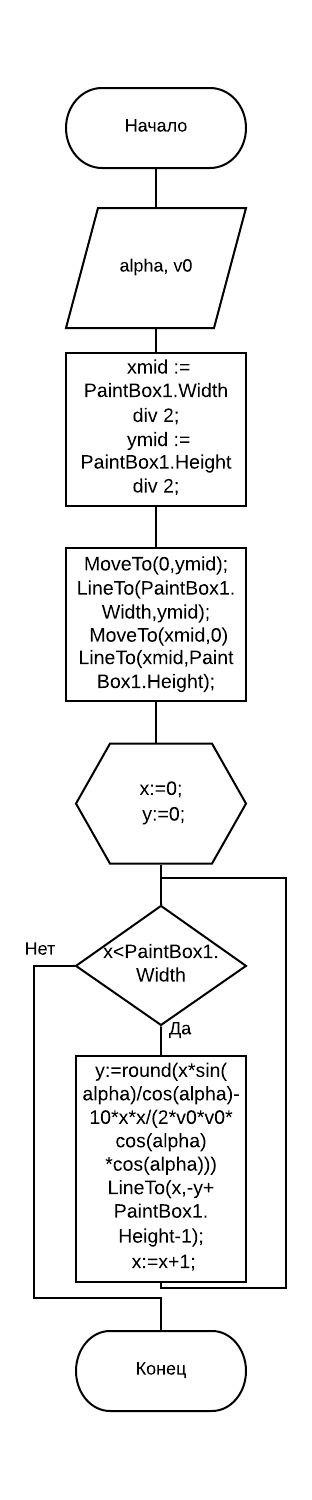
x:=x+1;

end;

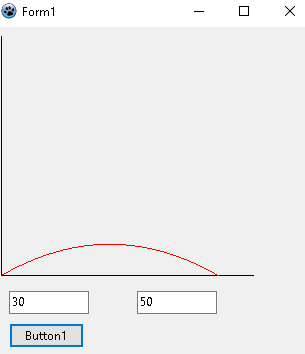
end;

end;

Блок-схема:



Скриншот:



Лабораторная работа №5

Задача : Построить правильный n -угольник, количество углов которого задано и вводиться через пользовательский интерфейс.

Математическая модель:

Код:

procedure TForm1.lineAngle(x, y, R, angle,i :integer);

begin

a[i]:=Round(x+cos(angle/180\*pi)\*R);

b[i]:=Round(y-sin(angle/180\*pi)\*R);

end;

procedure TForm1.Button1Click(Sender: TObject);

var xmid, ymid, r, z, n,i,angle: integer;

begin

n := strtoint(Edit1.Text);

with PaintBox1.Canvas do begin

xmid := PaintBox1.Width div 2;

ymid := PaintBox1.Height div 2;

r := PaintBox1.Height - 100;

angle := 360 div n;

z:=0;

for i:=0 to n do

begin

lineAngle(xmid,ymid,r,z,i);

z := z+ angle;

end;

MoveTo(a[0],b[0]);

for i:=0 to n do

begin

LineTo(a[i],b[i]);

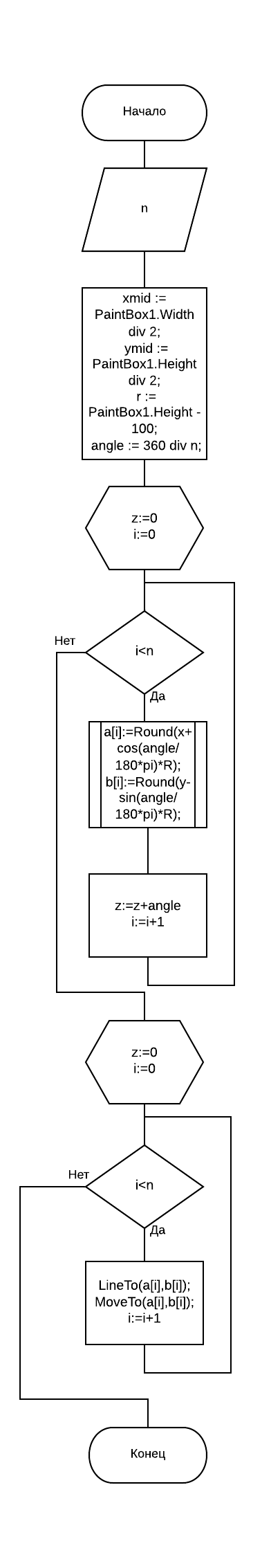
MoveTo(a[i],b[i]);

end;

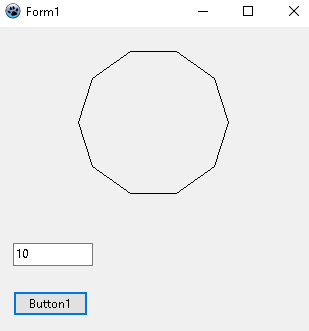
end;

end;

Блок-схема:



Скриншот:



Лабораторная работа №6

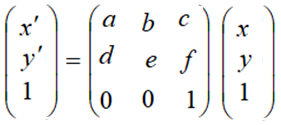
Задача: Построить произвольный объект в первой четверти системы координат (например, флажок. кол-во вершин объекта должно быть на меньше 7) и организовать следующие преобразования объекта:

- общее преобразование (изменение масштаба, симметричное отражение относительно осей, сдвиг),

- вращение объекта относительно начала координат

- комбинированное преобразование (преобразование общего вида + вращение на произвольный угол относительно начала координат)

Математическая модель:



Матрица отражения относительно OY

Матрица отражения относительно OX

Матрица изменения масштаба

Матрица поворота

Код:

public

var c,mt:matrix;

end;

function TForm1.matrixMult(a,b,c:matrix;n, m :integer):matrix;

var i, j, k: integer;

begin

for i := 1 to m do

for j := 1 to m do begin

c[i,j] := 0;

for k := 1 to n do

c[i,j] := c[i,j] + a[i,k] \* b[k,j];

end;

matrixMult:=c;

end;

procedure TForm1.Button1Click(Sender: TObject);

var xmid, ymid: integer;

alpha:real;

var a,b:matrix;

begin

with PaintBox1.Canvas do begin

xmid := PaintBox1.Width div 2;

ymid := PaintBox1.Height div 2;

MoveTo(0,ymid);

LineTo(PaintBox1.Width,ymid);

MoveTo(xmid,0);

LineTo(xmid,PaintBox1.Height);

a[1,1]:=xmid+20; a[2,1]:=ymid-20; a[3,1]:=0;

a[1,2]:=xmid+25; a[2,2]:=ymid-25; a[3,2]:=0;

a[1,3]:=xmid+20; a[2,3]:=ymid-30; a[3,3]:=0;

a[1,4]:=xmid+45; a[2,4]:=ymid-25; a[3,4]:=0;

a[1,5]:=xmid+70; a[2,5]:=ymid-30; a[3,5]:=0;

a[1,6]:=xmid+65; a[2,6]:=ymid-25; a[3,6]:=0;

a[1,7]:=xmid+70; a[2,7]:=ymid-20; a[3,7]:=0;

MoveTo(a[1,1],a[2,1]);

LineTo(a[1,2],a[2,2]);

LineTo(a[1,3],a[2,3]);

LineTo(a[1,4],a[2,4]);

LineTo(a[1,5],a[2,5]);

LineTo(a[1,6],a[2,6]);

LineTo(a[1,7],a[2,7]);

LineTo(a[1,4],a[2,4]);

LineTo(a[1,1],a[2,1]);

a[1,1]:=a[1,1]-xmid; a[2,1]:=ymid-a[2,1]; a[3,1]:=0;

a[1,2]:=a[1,2]-xmid; a[2,2]:=ymid-a[2,2]; a[3,2]:=0;

a[1,3]:=a[1,3]-xmid; a[2,3]:=ymid-a[2,3]; a[3,3]:=0;

a[1,4]:=a[1,4]-xmid; a[2,4]:=ymid-a[2,4]; a[3,4]:=0;

a[1,5]:=a[1,5]-xmid; a[2,5]:=ymid-a[2,5]; a[3,5]:=0;

a[1,6]:=a[1,6]-xmid; a[2,6]:=ymid-a[2,6]; a[3,6]:=0;

a[1,7]:=a[1,7]-xmid; a[2,7]:=ymid-a[2,7]; a[3,7]:=0;

b[1,1]:=-1; b[1,2]:=0; b[1,3]:=0;

b[2,1]:=0; b[2,2]:=1; b[2,3]:=0;

b[3,1]:=0; b[3,2]:=0; b[3,3]:=1;

mt:=matrixMult(b,a,c,3,7);

mt[1,1]:=mt[1,1]+xmid; mt[2,1]:=ymid-mt[2,1]; c[3,1]:=0;

mt[1,2]:=mt[1,2]+xmid; mt[2,2]:=ymid-mt[2,2]; c[3,2]:=0;

mt[1,3]:=mt[1,3]+xmid; mt[2,3]:=ymid-mt[2,3]; c[3,3]:=0;

mt[1,4]:=mt[1,4]+xmid; mt[2,4]:=ymid-mt[2,4]; c[3,4]:=0;

mt[1,5]:=mt[1,5]+xmid; mt[2,5]:=ymid-mt[2,5]; c[3,5]:=0;

mt[1,6]:=mt[1,6]+xmid; mt[2,6]:=ymid-mt[2,6]; c[3,6]:=0;

mt[1,7]:=mt[1,7]+xmid; mt[2,7]:=ymid-mt[2,7]; c[3,7]:=0;

Pen.Color := clRed;

MoveTo(mt[1,1],mt[2,1]);

LineTo(mt[1,2],mt[2,2]);

LineTo(mt[1,3],mt[2,3]);

LineTo(mt[1,4],mt[2,4]);

LineTo(mt[1,5],mt[2,5]);

LineTo(mt[1,6],mt[2,6]);

LineTo(mt[1,7],mt[2,7]);

LineTo(mt[1,4],mt[2,4]);

LineTo(mt[1,1],mt[2,1]);

b[1,1]:=-2; b[1,2]:=0; b[1,3]:=0;

b[2,1]:=0; b[2,2]:=-2; b[2,3]:=0;

b[3,1]:=0; b[3,2]:=0; b[3,3]:=1;

mt:=matrixMult(b,a,c,3,7);

mt[1,1]:=mt[1,1]+xmid; mt[2,1]:=ymid-mt[2,1]; c[3,1]:=0;

mt[1,2]:=mt[1,2]+xmid; mt[2,2]:=ymid-mt[2,2]; c[3,2]:=0;

mt[1,3]:=mt[1,3]+xmid; mt[2,3]:=ymid-mt[2,3]; c[3,3]:=0;

mt[1,4]:=mt[1,4]+xmid; mt[2,4]:=ymid-mt[2,4]; c[3,4]:=0;

mt[1,5]:=mt[1,5]+xmid; mt[2,5]:=ymid-mt[2,5]; c[3,5]:=0;

mt[1,6]:=mt[1,6]+xmid; mt[2,6]:=ymid-mt[2,6]; c[3,6]:=0;

mt[1,7]:=mt[1,7]+xmid; mt[2,7]:=ymid-mt[2,7]; c[3,7]:=0;

Pen.Color := clRed;

MoveTo(mt[1,1],mt[2,1]);

LineTo(mt[1,2],mt[2,2]);

LineTo(mt[1,3],mt[2,3]);

LineTo(mt[1,4],mt[2,4]);

LineTo(mt[1,5],mt[2,5]);

LineTo(mt[1,6],mt[2,6]);

LineTo(mt[1,7],mt[2,7]);

LineTo(mt[1,4],mt[2,4]);

LineTo(mt[1,1],mt[2,1]);

alpha:=220\*3.14/180;

b[1,1]:=round(cos(alpha)); b[1,2]:=round(sin(alpha)); b[1,3]:=0;

b[2,1]:=-round(sin(alpha)); b[2,2]:=round(cos(alpha)); b[2,3]:=0;

b[3,1]:=0; b[3,2]:=0; b[3,3]:=1;

mt:=matrixMult(b,a,c,3,7);

mt[1,1]:=mt[1,1]+xmid; mt[2,1]:=ymid-mt[2,1]; c[3,1]:=0;

mt[1,2]:=mt[1,2]+xmid; mt[2,2]:=ymid-mt[2,2]; c[3,2]:=0;

mt[1,3]:=mt[1,3]+xmid; mt[2,3]:=ymid-mt[2,3]; c[3,3]:=0;

mt[1,4]:=mt[1,4]+xmid; mt[2,4]:=ymid-mt[2,4]; c[3,4]:=0;

mt[1,5]:=mt[1,5]+xmid; mt[2,5]:=ymid-mt[2,5]; c[3,5]:=0;

mt[1,6]:=mt[1,6]+xmid; mt[2,6]:=ymid-mt[2,6]; c[3,6]:=0;

mt[1,7]:=mt[1,7]+xmid; mt[2,7]:=ymid-mt[2,7]; c[3,7]:=0;

Pen.Color := clRed;

MoveTo(mt[1,1],mt[2,1]);

LineTo(mt[1,2],mt[2,2]);

LineTo(mt[1,3],mt[2,3]);

LineTo(mt[1,4],mt[2,4]);

LineTo(mt[1,5],mt[2,5]);

LineTo(mt[1,6],mt[2,6]);

LineTo(mt[1,7],mt[2,7]);

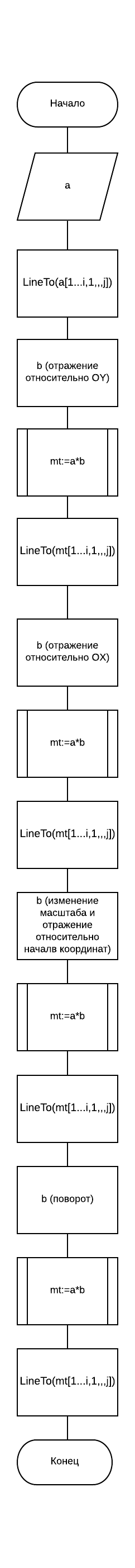
LineTo(mt[1,4],mt[2,4]);

LineTo(mt[1,1],mt[2,1]);

end;

end;

Блок-схема:



Скриншот:

