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

АФФИНЫЕ ПРЕОБРАЗОВАНИЯ В ПРОСТРАНСТВЕ

Отчет

Лабораторная работа №7 по дисциплине

«Компьютерная графика»

Выполнил студент группы ИВТб-21\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Боров К.Э /

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

2021 г.

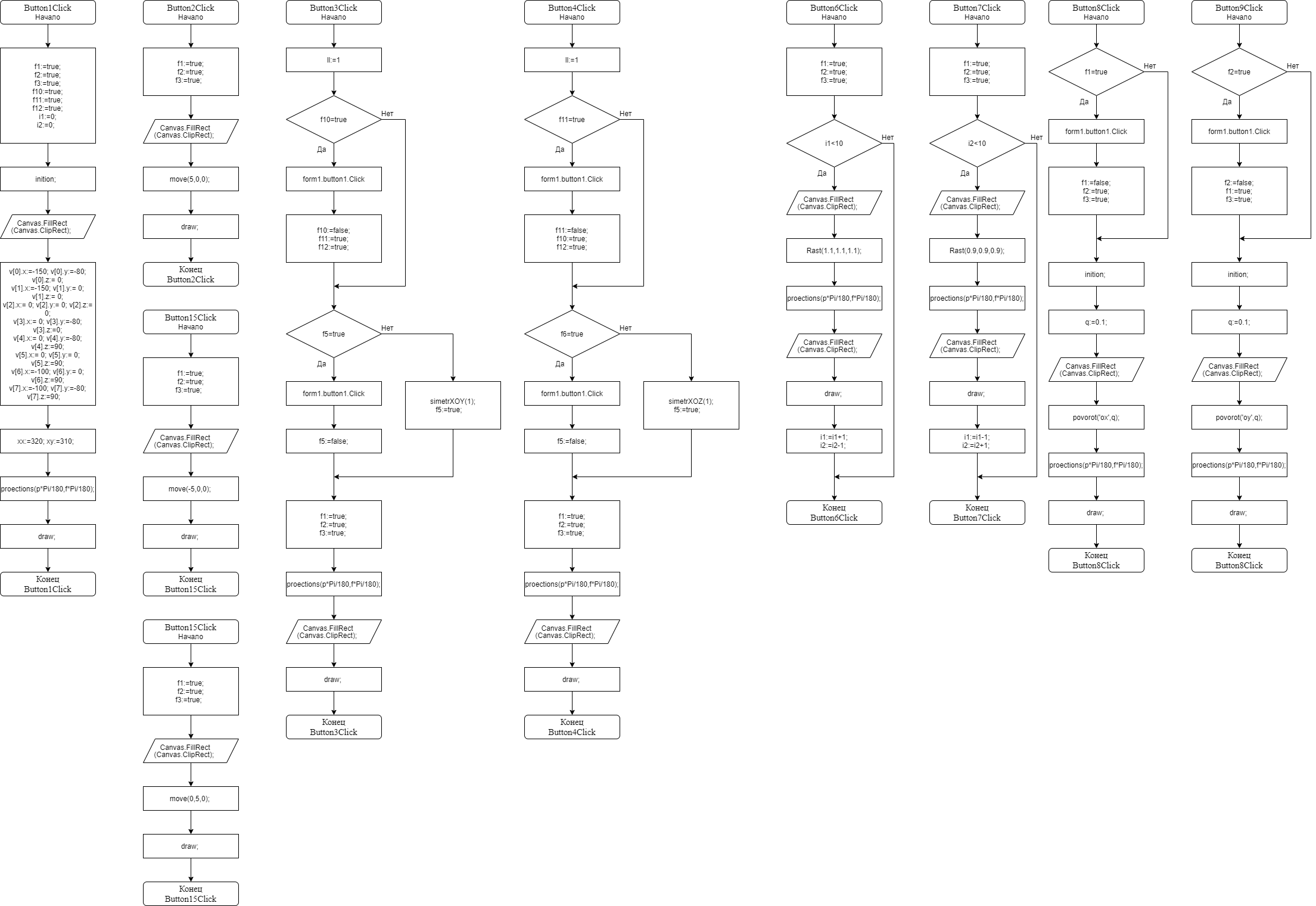
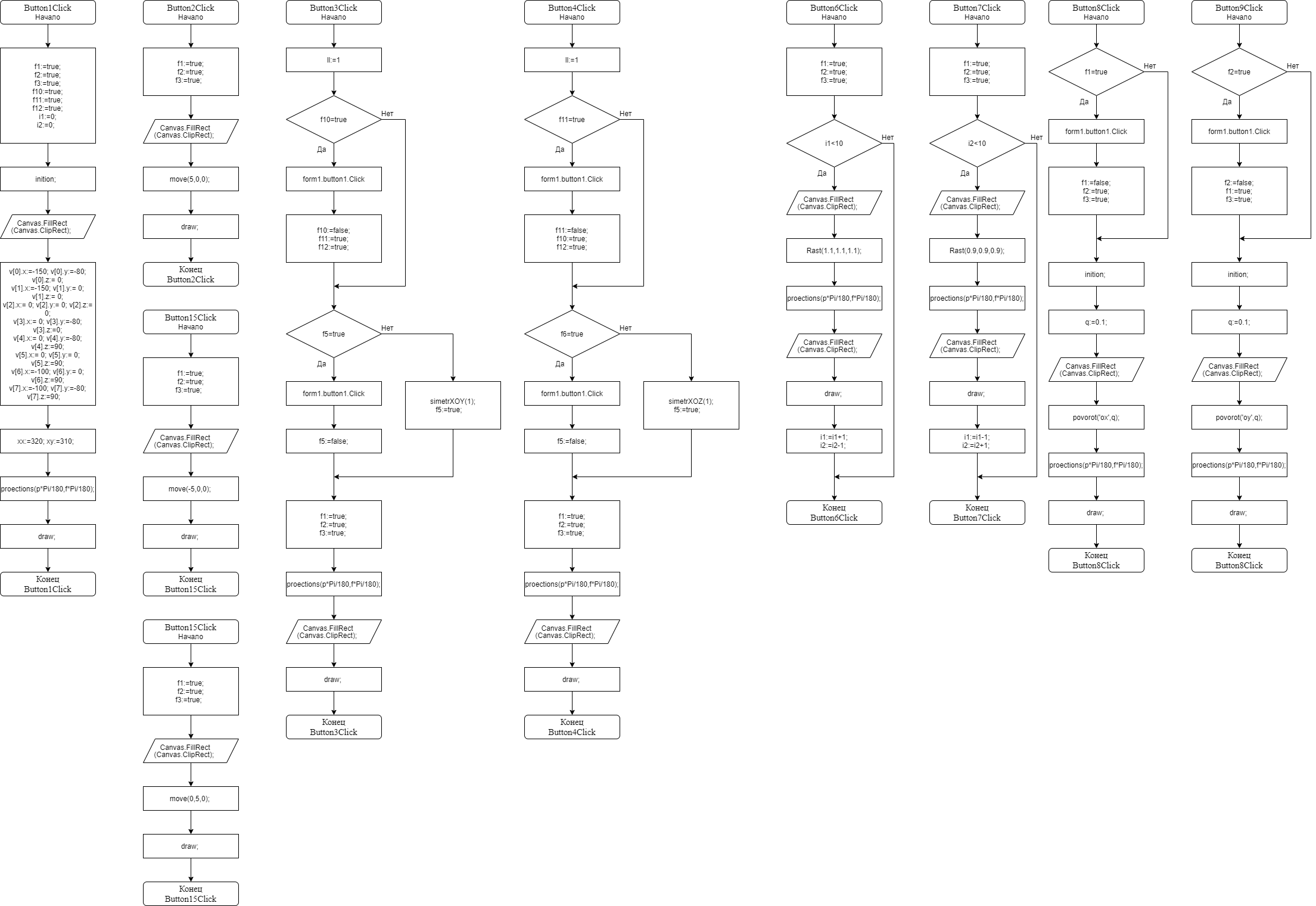
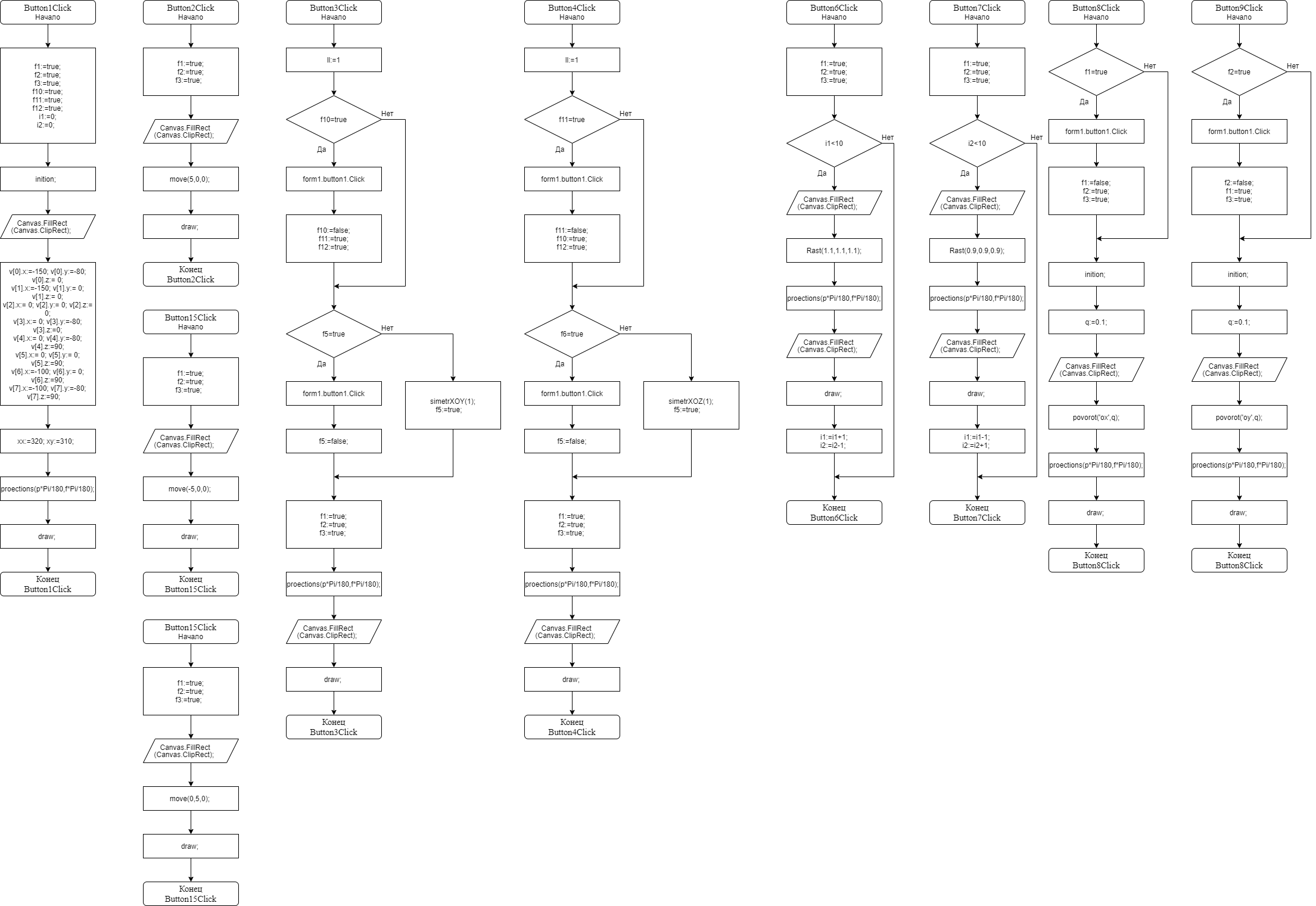
**Цель:** закрепить лекционный материал по изучению материала одноименной темы, реализовав матрицы переноса, масштабирования, отражения и вращения применительно к координатам описанной в программе объемной фигуры (многогранника) с целью демонстрации движения и преобразования формы этой фигуры в пространстве.

***Задание на лабораторную работу:***

Написать на языке Pascal программу:

1. Описывающую многогранник (куб) в приборной системе координат.
2. Смещающую его на n пикселов вправо, m - вниз, p - вглубь.
3. Зеркально отражающую относительно плоскостей координат.
4. Растягивающую (сжимающую) его вдоль координатных осей относительно некоторой заданной точки.
5. Вращающую его относительно линии, проходящей через начало координат (относительно координатных осей, диагонали многогранника).
6. Реализовать интерактивную 3Д анимацию (взаимодействующую с пользователем), на которой выполняется поворот и масштабирование объектов.

**Схемы алгоритмов:**



**Исходный код**

**unit** Unit1;

{$mode objfpc}{$H+}

**interface**

**uses**

Classes, SysUtils, FileUtil, Forms, Controls, Graphics, Dialogs, StdCtrls,

ExtCtrls;

**type**

TXYZ=**record**

x,y,z:real;

**end**;

tcub=**record**

x,y,z:real;

**end**;

TXY=**record**

x,y:integer;

**end**;

{ TForm1 }

TForm1 = **class**(TForm)

Button1: TButton;

Button10: TButton;

Button11: TButton;

Button13: TButton;

Button14: TButton;

Button15: TButton;

Button16: TButton;

Button17: TButton;

Button2: TButton;

Button3: TButton;

Button4: TButton;

Button5: TButton;

Button6: TButton;

Button7: TButton;

Button8: TButton;

Button9: TButton;

Label1: TLabel;

Label2: TLabel;

Label3: TLabel;

Panel1: TPanel;

Timer1: TTimer;

**procedure** Button10Click(Sender: TObject);

**procedure** Button11Click(Sender: TObject);

**procedure** Button13Click(Sender: TObject);

**procedure** Button14Click(Sender: TObject);

**procedure** Button1Click(Sender: TObject);

**procedure** Button2Click(Sender: TObject);

**procedure** Button3Click(Sender: TObject);

**procedure** Button4Click(Sender: TObject);

**procedure** Button5Click(Sender: TObject);

**procedure** Button6Click(Sender: TObject);

**procedure** Button7Click(Sender: TObject);

**procedure** Button8Click(Sender: TObject);

**procedure** Button9Click(Sender: TObject);

**procedure** Button15Click(Sender: TObject);

**procedure** Button16Click(Sender: TObject);

**procedure** Button17Click(Sender: TObject);

**procedure** FormCreate(Sender: TObject);

**private**

{ private declarations }

**public**

**procedure** draw;

**procedure** helper;

**procedure** inition;

**procedure** move(dx,dy,dz:integer);

**procedure** proections(k, p: real);

**procedure** Rast(kx,ky,kz:real);

**procedure** simetrXOY(a:integer);

**procedure** simetrXOZ(a:integer);

**procedure** simetrYOZ(a:integer);

**procedure** vrash(x1,y1,z1:integer; q:real);

**procedure** povorot(per:string; q:real);

**procedure** blakcStar;

{ public declarations }

**end**;

**Const** p=45;f=35.264;

**var**

ff1,ff2,ff3,ll:integer;

flag1, flag2, flag3,f1,f2,f3,f5,f6,f7,f10,f11,f12:boolean;

Form1: TForm1;

q:real;

i1,i2: integer;

tkub:**array**[0..7] **of** tcub;

xx,xy,xz,i:integer;

v1:**array** [0..7] **of** TXY;

help,temp:**array**[0..7] **of** TXYZ;

v:**array**[0..7] **of** TXYZ=

((x: -100; y: -80; z: 0),

(x: -100; y: 0; z: 0),

(x: 0; y: 0; z: 0),

(x: 0; y: -80; z: 0),

(x: 0; y: -80; z: 90),

(x: 0; y: 0; z: 90),

(x: -100; y: 0; z: 90),

(x: -100; y: -80; z: 90));

**implementation**

{$R \*.lfm}

**procedure** TForm1.draw;

**begin**

**with** Canvas **do**

**begin**

Pen.Color:=clBlack;

Pen.Width:=1;

MoveTo(v1[0].x + xx, v1[0].y + xy);

LineTo(v1[1].x + xx, v1[1].y + xy);

MoveTo(v1[1].x + xx, v1[1].y + xy);

LineTo(v1[2].x + xx, v1[2].y + xy);

MoveTo(v1[2].x + xx, v1[2].y + xy);

LineTo(v1[3].x + xx, v1[3].y + xy);

MoveTo(v1[3].x + xx, v1[3].y + xy);

LineTo(v1[4].x + xx, v1[4].y + xy);

MoveTo(v1[4].x + xx, v1[4].y + xy);

LineTo(v1[5].x + xx, v1[5].y + xy);

MoveTo(v1[5].x + xx, v1[5].y + xy);

LineTo(v1[6].x + xx, v1[6].y + xy);

MoveTo(v1[6].x + xx, v1[6].y + xy);

LineTo(v1[7].x + xx, v1[7].y + xy);

MoveTo(v1[7].x + xx, v1[7].y + xy);

LineTo(v1[0].x + xx, v1[0].y + xy);

MoveTo(v1[0].x + xx, v1[0].y + xy);

LineTo(v1[3].x + xx, v1[3].y + xy);

MoveTo(v1[2].x + xx, v1[2].y + xy);

LineTo(v1[5].x + xx, v1[5].y + xy);

MoveTo(v1[1].x + xx, v1[1].y + xy);

LineTo(v1[6].x + xx, v1[6].y + xy);

MoveTo(v1[4].x + xx, v1[4].y + xy);

LineTo(v1[7].x + xx, v1[7].y + xy);

**end**;

**end**;

**procedure** TForm1.helper;

**begin**

**for** i:=0 **to** 7 **do begin**

help[i].x:=v[i].x;

help[i].y:=v[i].y;

help[i].z:=v[i].z;

**end**;

**end**;

**procedure** TForm1.inition;

**begin**

**for** i:=0 **to** 7 **do begin**

temp[i].x:=v[i].x;

temp[i].y:=v[i].y;

temp[i].z:=v[i].z;

**end**;

**end**;

**procedure** TForm1.blakcStar;

**begin**

**for** i:=0 **to** 7 **do begin**

v[i].x:=temp[i].x;

v[i].y:=temp[i].y;

v[i].z:=temp[i].z;

**end**;

proections(p\*Pi/180,f\*Pi/180);

**end**;

**procedure** TForm1.move(dx,dy,dz:integer);

**begin**

**for** i:=0 **to** 7 **do begin**

v[i].x:=v[i].x+dx;

v[i].y:=v[i].y+dy;

v[i].z:=v[i].z+dz;

**end**;

proections(p\*Pi/180,f\*Pi/180);

**end**;

**procedure** TForm1.proections(k, p: real);

**begin**

**for** i:=0 **to** 7 **do begin**

v1[i].x:=round(V[i].x\*cos(p)+V[i].z\*sin(p));

v1[i].y:=round(-V[i].x\*sin(k)\*sin(p)+V[i].y\*cos(k)+V[i].z\*sin(k)\*cos(p));

**end**;

**end**;

**procedure** TForm1.Rast(kx,ky,kz:real);

**begin**

**for** i:=0 **to** 7 **do begin**

v[i].x:=round(v[i].x\*kx);

v[i].y:=round(v[i].y\*ky);

v[i].z:=round(v[i].z\*kz);

**end**;

**end**;

**procedure** TForm1.simetrXOY(a:integer);

**begin**

**for** i:=0 **to** 7 **do begin**

v[i].x:=v[i].x;

v[i].y:=v[i].y;

v[i].z:=a\*(-v[i].z);

{v[i].z:=-v[i].z;}

**end**;

**end**;

**procedure** TForm1.simetrXOZ(a:integer);

**begin**

**for** i:=0 **to** 7 **do begin**

v[i].x:=v[i].x;

v[i].y:=-v[i].y;

v[i].z:=v[i].z;

**end**;

**end**;

**procedure** TForm1.simetrYOZ(a:integer);

**begin**

**for** i:=0 **to** 7 **do begin**

v[i].x:=-v[i].x;

v[i].y:=v[i].y;

v[i].z:=v[i].z;

**end**;

**end**;

**procedure** TForm1.vrash(x1,y1,z1:integer; q:real);

**var** n1,n2,n3,x,y,z:real;

**begin**

n1:=x1/sqrt(sqr(x1)+sqr(y1)+sqr(z1));

n2:=y1/sqrt(sqr(x1)+sqr(y1)+sqr(z1));

n3:=z1/sqrt(sqr(x1)+sqr(y1)+sqr(z1));

**for** i:=0 **to** 7 **do**

**begin**

tkub[i].x:=v[i].x;

tkub[i].y:=v[i].y;

tkub[i].z:=v[i].z;

**end**;

**for** i:=0 **to** 7 **do**

**begin**

x:=tkub[i].x;

y:=tkub[i].y;

z:=tkub[i].z;

tkub[i].x:=(x\*(n1\*n1+(1-n1\*n1)\*cos(q))+ y\*(n1\*n2\*(1-cos(q))-n3\*sin(q))+z\*(n1\*n3\*(1-cos(q))+n2\*sin(q)));

tkub[i].y:=(x\*(n1\*n2\*(1-cos(q))+n3\*sin(q))+y\*(n2\*n2+(1-n2\*n2)\*cos(q))+ z\*(n2\*n3\*(1-cos(q))-n1\*sin(q)));

tkub[i].z:=(x\*(n1\*n3\*(1-cos(q))-n2\*sin(q))+y\*(n2\*n3\*(1-cos(q))+n1\*sin(q))+z\*(n3\*n3+(1-n3\*n3)\*cos(q)));

v[i].x:=tkub[i].x;

v[i].y:=tkub[i].y;

v[i].z:=tkub[i].z;

**end**;

**end**;

**procedure** TForm1.povorot(per:string; q:real);

**begin**

**case** per **of**

'ox':**begin**

helper;

**for** i:=0 **to** 7 **do begin**

v[i].x:=help[i].x;

v[i].y:=round( help[i].y\*Cos(q)-help[i].z\*Sin(q) );

v[i].z:=round( help[i].y\*Sin(q)+help[i].z\*Cos(q) );

**end**;

**end**;

'oz':**begin**

helper;

**for** i:=0 **to** 7 **do begin**

v[i].x:=round( help[i].x\*Cos(q)+help[i].z\*Sin(q) );

v[i].y:=help[i].y;

v[i].z:=round( -help[i].x\*Sin(q)+help[i].z\*Cos(q) );

**end**;

**end**;

'oy':**begin**

helper;

**for** i:=0 **to** 7 **do begin**

v[i].x:=round( help[i].x\*Cos(q)-help[i].y\*Sin(q) );

v[i].y:=round( help[i].x\*Sin(q)+help[i].y\*Cos(q) );

v[i].z:=help[i].z;

**end**;

**end**;

**end**;

**end**;

**procedure** TForm1.Button1Click(Sender: TObject);

**begin**

f1:=true;

f2:=true;

f3:=true;

f10:=true;

f11:=true;

f12:=true;

i1:=0;

i2:=0;

inition;

Canvas.Brush.Color:=clWhite;

Canvas.FillRect(Canvas.ClipRect);

v[0].x:=-150; v[0].y:=-80; v[0].z:= 0;

v[1].x:=-150; v[1].y:= 0; v[1].z:= 0;

v[2].x:= 0; v[2].y:= 0; v[2].z:= 0;

v[3].x:= 0; v[3].y:=-80; v[3].z:=0;

v[4].x:= 0; v[4].y:=-80; v[4].z:=90;

v[5].x:= 0; v[5].y:= 0; v[5].z:=90;

v[6].x:=-100; v[6].y:= 0; v[6].z:=90;

v[7].x:=-100; v[7].y:=-80; v[7].z:=90;

xx:=320; xy:=310;

proections(p\*Pi/180,f\*Pi/180);

Canvas.Pen.Color:=clBlack;

Canvas.Pen.Width:=1;

draw;

**end**;

//сдвиг

**procedure** TForm1.Button2Click(Sender: TObject);

**begin**

f1:=true;

f2:=true;

f3:=true;

Canvas.Brush.Color:=clWhite;

Canvas.FillRect(Canvas.ClipRect);

Canvas.Pen.Color:=clBlack;

move(5,0,0);

draw;

**end**;

**procedure** TForm1.Button15Click(Sender: TObject);

**begin**

f1:=true;

f2:=true;

f3:=true;

Canvas.Brush.Color:=clWhite;

Canvas.FillRect(Canvas.ClipRect);

Canvas.Pen.Color:=clBlack;

move(-5,0,0);

draw;

**end**;

**procedure** TForm1.Button13Click(Sender: TObject);

**begin**

f1:=true;

f2:=true;

f3:=true;

Canvas.Brush.Color:=clWhite;

Canvas.FillRect(Canvas.ClipRect);

Canvas.Pen.Color:=clBlack;

move(0,5,0);

draw;

**end**;

**procedure** TForm1.Button16Click(Sender: TObject);

**begin**

f1:=true;

f2:=true;

f3:=true;

Canvas.Brush.Color:=clWhite;

Canvas.FillRect(Canvas.ClipRect);

Canvas.Pen.Color:=clBlack;

move(0,-5,0);

draw;

**end**;

**procedure** TForm1.Button14Click(Sender: TObject);

**begin**

f1:=true;

f2:=true;

f3:=true;

Canvas.Brush.Color:=clWhite;

Canvas.FillRect(Canvas.ClipRect);

Canvas.Pen.Color:=clBlack;

move(0,0,5);

draw;

**end**;

**procedure** TForm1.Button17Click(Sender: TObject);

**begin**

f1:=true;

f2:=true;

f3:=true;

Canvas.Brush.Color:=clWhite;

Canvas.FillRect(Canvas.ClipRect);

Canvas.Pen.Color:=clBlack;

move(0,0,-5);

draw;

**end**;

//отражение

**procedure** TForm1.Button3Click(Sender: TObject);

**begin**

ll:=1;

**if** f10=true **then**

**begin**

form1.button1.Click;

f10:=false;

f11:=true;

f12:=true;

**end**;

**if** f5=true **then begin**

form1.button1.Click;

f5:=false;

**end**

**else begin**

simetrXOY(1);

f5:=true;

**end**;

f1:=true;

f2:=true;

f3:=true;

Canvas.Brush.Color:=clWhite;

Canvas.FillRect(Canvas.ClipRect);

proections(p\*Pi/180,f\*Pi/180);

Canvas.Brush.Color:=clWhite;

Canvas.FillRect(Canvas.ClipRect);

draw;

**end**;

**procedure** TForm1.Button4Click(Sender: TObject);

**begin**

**if** f11=true **then**

**begin**

form1.button1.Click;

f11:=false;

f10:=true;

f12:=true;

**end**;

f1:=true;

f2:=true;

f3:=true;

**if** f6=true **then begin**

form1.button1.Click;

f6:=false;

**end**

**else begin**

simetrXOZ(1);

f6:=true;

**end**;

Canvas.Brush.Color:=clWhite;

Canvas.FillRect(Canvas.ClipRect);

proections(p\*Pi/180,f\*Pi/180);

Canvas.Brush.Color:=clWhite;

Canvas.FillRect(Canvas.ClipRect);

draw;

**end**;

**procedure** TForm1.Button5Click(Sender: TObject);

**begin**

**if** f12=true **then**

**begin**

form1.button1.Click;

f12:=false;

f11:=true;

f10:=true;

**end**;

f1:=true;

f2:=true;

f3:=true;

**if** f7=true **then begin**

form1.button1.Click;

f7:=false;

**end**

**else begin**

simetrYOZ(1);

f7:=true;

**end**;

Canvas.Brush.Color:=clWhite;

Canvas.FillRect(Canvas.ClipRect);

proections(p\*Pi/180,f\*Pi/180);

Canvas.Brush.Color:=clWhite;

Canvas.FillRect(Canvas.ClipRect);

draw;

**end**;

//растяжение

**procedure** TForm1.Button6Click(Sender: TObject);

**begin**

f1:=true;

f2:=true;

f3:=true;

**if** i1<10 **then begin**

Canvas.Brush.Color:=clWhite;

Canvas.FillRect(Canvas.ClipRect);

Canvas.Pen.Color:=clBlack;

Rast(1.1,1.1,1.1);

proections(p\*Pi/180,f\*Pi/180);

Canvas.Brush.Color:=clWhite;

Canvas.FillRect(Canvas.ClipRect);

draw;

i1:=i1+1;

i2:=i2-1;

**end**;

**end**;

**procedure** TForm1.Button7Click(Sender: TObject);

**begin**

f1:=true;

f2:=true;

f3:=true;

**if** i2<10 **then begin**

Canvas.Brush.Color:=clWhite;

Canvas.FillRect(Canvas.ClipRect);

Canvas.Pen.Color:=clBlack;

Rast(0.9,0.9,0.9);

proections(p\*Pi/180,f\*Pi/180);

Canvas.Brush.Color:=clWhite;

Canvas.FillRect(Canvas.ClipRect);

draw;

i1:=i1-1;

i2:=i2+1;

**end**;

**end**;

**procedure** TForm1.Button8Click(Sender: TObject); //поворот

**begin**

**if** f1=true **then begin**

form1.button1.Click;

f1:=false;

f2:=true;

f3:=true;

**end**;

inition;

q:=0.1;

Canvas.Brush.Color:=clWhite;

Canvas.FillRect(Canvas.ClipRect);

povorot('ox',q);

proections(p\*Pi/180,f\*Pi/180);

Canvas.Pen.Color:=clRed;

draw;

Canvas.Pen.Color:=clred;

draw;

**end**;

**procedure** TForm1.Button9Click(Sender:TObject);

**begin**

**if** f2=true **then begin**

form1.button1.Click;

f2:=false;

f1:=true;

f3:=true;

**end**;

inition;

q:=0.1;

Canvas.Brush.Color:=clWhite;

Canvas.FillRect(Canvas.ClipRect);

povorot('oy',q);

proections(p\*Pi/180,f\*Pi/180);

Canvas.Pen.Color:=clRed;

draw;

**end**;

**procedure** TForm1.Button10Click(Sender: TObject);

**begin**

**if** f3=true **then begin**

form1.button1.Click;

f3:=false;

f2:=true;

f1:=true;

**end**;

q:=0.1;

inition;

Canvas.Brush.Color:=clWhite;

Canvas.FillRect(Canvas.ClipRect); ;

povorot('oz',q);

proections(p\*Pi/180,f\*Pi/180);

Canvas.Pen.Color:=clRed;

draw;

**end**;

**procedure** TForm1.FormCreate(Sender: TObject);

**begin**

ll:=0;

flag1:=true;

Flag2:=true;

flag3:=true;

f1:=true;

f2:=true;

f3:=true;

f5:=true;

f6:=true;

f7:=true;

f10:=true;

f11:=true;

f12:=true;

**with** TShape.Create(Self) **do**

**begin**

Parent:=Self;

Left:=0;

Top:=0;

Width:=10000;

Height:=10000;

Visible:=True;

Shape:=stRectangle;

Brush.Color:=clWhite;

Pen.Color:=clWhite;

**end**;

inition;

Canvas.Brush.Color:=clWhite;

Canvas.FillRect(Canvas.ClipRect);

v[0].x:=-100; v[0].y:=-80; v[0].z:= 0;

v[1].x:=-100; v[1].y:= 0; v[1].z:= 0;

v[2].x:= 0; v[2].y:= 0; v[2].z:= 0;

v[3].x:= 0; v[3].y:=-80; v[3].z:=0;

v[4].x:= 0; v[4].y:=-80; v[4].z:=90;

v[5].x:= 0; v[5].y:= 0; v[5].z:=90;

v[6].x:=-100; v[6].y:= 0; v[6].z:=90;

v[7].x:=-100; v[7].y:=-80; v[7].z:=90;

xx:=320; xy:=310;

proections(p\*Pi/180,f\*Pi/180);

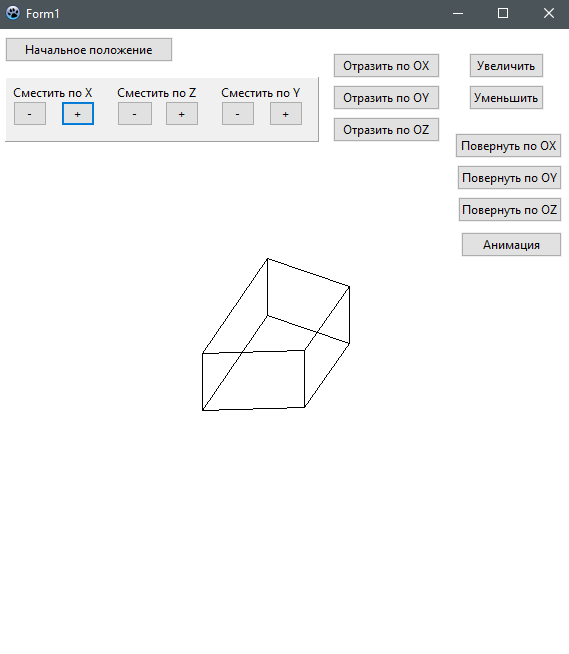
Canvas.Pen.Color:=clBlack;

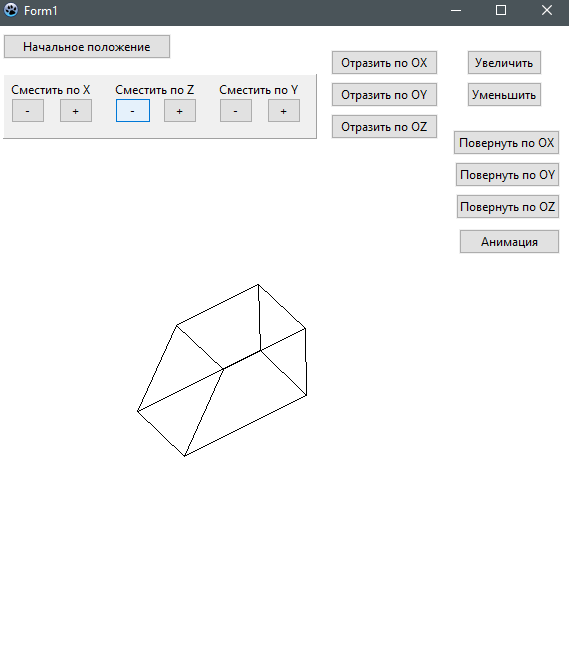
Canvas.Pen.Width:=1;

draw;

**end**;

**Экранные формы**





**Вывод:** были реализованы матрицы переноса, масштабирования, отражения и вращения применительно к координатам описанной в программе объемной фигуры (многогранника) с целью демонстрации движения и преобразования формы этой фигуры в пространстве.