Ministerul Educaţiei, Culturii și Cercetării al Republicii Moldova

Universitatea Tehnică a Moldovei

Departamentul Ingineria Software și Automatică

**RAPORT**

Lucrare de laborator Nr.2

Disciplina: GC

Tema: Transformări geometrice a figurilor grafice 2D

A efectuat: Белоскурник Дан st.gr. TI-186,

A verificat : Osovschi Mariana lect.univ.

Chișinău 2020

Цель работы: получение практических знаний по выполнению геометрических преобразований масштабирования, смещения и вращения двумерных фигур. Применение матриц преобразования геометрических фигур.

Рабочая задача:

1. Разработайте программу, которая рисует двухмерные геометрические фигуры и применяет вращение, масштабирование и перемещение преобразований к плоскости. Поворот вокруг оси и точки.

//---------------------------------------------------------------------------

#include <vcl.h>

#include <math.h>

#pragma hdrstop

#include "Unit1.h"

//---------------------------------------------------------------------------

#pragma package(smart\_init)

#pragma resource "\*.dfm"

TForm1 \*Form1;

//---------------------------------------------------------------------------

\_\_fastcall TForm1::TForm1(TComponent\* Owner)

: TForm(Owner)

{

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button1Click(TObject \*Sender)

{

Canvas->Brush->Color = clBlack;

Canvas->Pen->Color = clBlue;

Canvas->Pen->Width=2;

Canvas->MoveTo(40,135);

Canvas->LineTo(40,225);

Canvas->Arc(40,210,80,240,40,225,60,240);

Canvas->MoveTo(60,240);

Canvas->LineTo(120,240);

Canvas->Arc(100,240,140,270,140,255,120,240);

Canvas->MoveTo(140,255);

Canvas->LineTo(140,285);

Canvas->Arc(100,270,140,300,120,300, 140,285);

Canvas->MoveTo(120,300);

Canvas->LineTo(40,300);

Canvas->MoveTo(40,330);

Canvas->LineTo(120,330);

Canvas->Arc(100,330,140,360,140,345,120,330);

Canvas->MoveTo(140,345);

Canvas->LineTo(140,375);

Canvas->Arc(100,360,140,390,120,390,140,375);

Canvas->MoveTo(120,390);

Canvas->LineTo(60,390);

Canvas->Arc(40,390,80,420,60,390,40,405);

Canvas->MoveTo(40,405);

Canvas->LineTo(40,495);

Canvas->Arc(40,480,80,510,40,495,60,510);

Canvas->MoveTo(60,510);

Canvas->LineTo(740,510);

Canvas->Arc(720,480,760,510,740,510,760,495);

Canvas->MoveTo(760,495);

Canvas->LineTo(760,405);

Canvas->Arc(720,390,760,420,760,405,740,390);

Canvas->MoveTo(740,390);

Canvas->LineTo(680,390);

Canvas->Arc(660,360,700,390,660,375,680,390);

Canvas->MoveTo(660,375);

Canvas->LineTo(660,345);

Canvas->Arc(660,330,700,360,680,330,660,345);

Canvas->MoveTo(680,330);

Canvas->LineTo(760,330);

Canvas->MoveTo(760,300);

Canvas->LineTo(680,300);

Canvas->Arc(660,270,700,300,660,285,680,300);

Canvas->MoveTo(660,285);

Canvas->LineTo(660,255);

Canvas->Arc(660,240,700,270,680,240,660,255);

Canvas->MoveTo(680,240);

Canvas->LineTo(740,240);

Canvas->Arc(720,210,760,240,740,240,760,225);

Canvas->MoveTo(760,225);

Canvas->LineTo(760,135);

Canvas->Arc(720,120,760,150,760,135,740,120);

Canvas->MoveTo(740,120);

Canvas->LineTo(60,120);

Canvas->Arc(40,120,80,150,60,120,40,135);

Canvas->Rectangle(80,150,140,210);

Canvas->Rectangle(180,150,340,210);

Canvas->Rectangle(380,120,420,180);

Canvas->Rectangle(460,150,540,180);

TPoint tPoints[6];

tPoints[0].x = 580;

tPoints[0].y = 150;

tPoints[1].x = 720;

tPoints[1].y = 150;

tPoints[2].x = 720;

tPoints[2].y = 210;

tPoints[3].x = 660;

tPoints[3].y = 210;

tPoints[4].x = 660;

tPoints[4].y = 180;

tPoints[5].x = 580;

tPoints[5].y = 180;

Canvas->Polygon(tPoints,5);

Canvas->Rectangle(540,210,620,240);

TPoint tPoints1[9];

tPoints1[0].x = 500;

tPoints1[0].y = 210;

tPoints1[1].x = 500;

tPoints1[1].y = 270;

tPoints1[2].x = 380;

tPoints1[2].y = 270;

tPoints1[3].x = 380;

tPoints1[3].y = 210;

tPoints1[4].x = 420;

tPoints1[4].y = 210;

tPoints1[5].x = 420;

tPoints1[5].y = 240;

tPoints1[6].x = 460;

tPoints1[6].y = 240;

tPoints1[7].x = 460;

tPoints1[7].y = 210;

tPoints1[8].x = 500;

tPoints1[8].y = 210;

Canvas->Polyline(tPoints1,8);

Canvas->Rectangle(260,240,340,270);

TPoint tPoints2[8];

tPoints2[0].x = 220;

tPoints2[0].y = 240;

tPoints2[1].x = 220;

tPoints2[1].y = 300;

tPoints2[2].x = 300;

tPoints2[2].y = 300;

tPoints2[3].x = 300;

tPoints2[3].y = 330;

tPoints2[4].x = 220;

tPoints2[4].y = 330;

tPoints2[5].x = 220;

tPoints2[5].y = 390;

tPoints2[6].x = 180;

tPoints2[6].y = 390;

tPoints2[7].x = 180;

tPoints2[7].y = 240;

Canvas->Polygon(tPoints2,7);

TPoint tPoints3[6];

tPoints3[0].x = 620;

tPoints3[0].y = 270;

tPoints3[1].x = 620;

tPoints3[1].y = 360;

tPoints3[2].x = 500;

tPoints3[2].y = 360;

tPoints3[3].x = 500;

tPoints3[3].y = 300;

tPoints3[4].x = 540;

tPoints3[4].y = 300;

tPoints3[5].x = 540;

tPoints3[5].y = 270;

Canvas->Polygon(tPoints3,5);

TPoint tPoints4[8];

tPoints4[0].x = 250;

tPoints4[0].y = 360;

tPoints4[1].x = 250;

tPoints4[1].y = 480;

tPoints4[2].x = 340;

tPoints4[2].y = 480;

tPoints4[3].x = 340;

tPoints4[3].y = 420;

tPoints4[4].x = 460;

tPoints4[4].y = 420;

tPoints4[5].x = 460;

tPoints4[5].y = 390;

tPoints4[6].x = 300;

tPoints4[6].y = 390;

tPoints4[7].x = 300;

tPoints4[7].y = 360;

Canvas->Polygon(tPoints4,7);

TPoint tPoints5[6];

tPoints5[0].x = 500;

tPoints5[0].y = 390;

tPoints5[1].x = 500;

tPoints5[1].y = 450;

tPoints5[2].x = 460;

tPoints5[2].y = 450;

tPoints5[3].x = 460;

tPoints5[3].y = 480;

tPoints5[4].x = 620;

tPoints5[4].y = 480;

tPoints5[5].x = 620;

tPoints5[5].y = 390;

Canvas->Polygon(tPoints5,5);

Canvas->Rectangle(80,420,140,480);

Canvas->Rectangle(180,420,220,480);

Canvas->Rectangle(660,420,720,480);

Canvas->Rectangle(380,450,420,510);

// Canvas->Pen->Color = clBlue;

Canvas->Pen->Width=4;

Canvas->MoveTo(380,300);

Canvas->LineTo(340,300);

Canvas->LineTo(340,360);

Canvas->LineTo(460,360);

Canvas->LineTo(460,300);

Canvas->LineTo(420,300);

Canvas->Pen->Color = clFuchsia;

Canvas->Pen->Width=2;

Canvas->LineTo(380,300);

//---------------------------------------------------------

Canvas->Pen->Color = clBlack;

Canvas->Pen->Width=1;

Canvas->Brush->Color = clRed;

Canvas->Ellipse(51,131,71,143);

Canvas->Ellipse(151,219,169,231);

Canvas->Brush->Color = clYellow;

int i;

int x1=92;

int x2=108;

int d;

Canvas->Ellipse(x1,129,x2,141);

for(i=1;i<7;i++)

{

d=43;

x1=x1+d;

x2=x2+d;

Canvas->Ellipse(x1,129,x2,141);

}

Canvas->Ellipse(352,249,368,261);

Canvas->Ellipse(152,159,168,171);

Canvas->Ellipse(152,159+30,168,171+30);

int xx1=154;

int xx2=168;

int y1=219;

int y2=231;

int l;

for(i=1;i<4;i++)

{

d=33;

xx1=xx1-d;

xx2=xx2-d;

Canvas->Ellipse(xx1,219,xx2,231);

}

Canvas->Ellipse(54,189,69,201);

Canvas->Ellipse(54,189-27,69,201-27);

/\*

int xfix=160;

int yfix=405;

for(i=0;i<8;i++)

{

tPoints7[i].x=xfix+(tPoints6[i].x-xfix)\*cos(90)-(tPoints6[i].y-yfix)\*sin(90);

tPoints7[i].y=yfix+(tPoints6[i].x-xfix)\*sin(90)-(tPoints6[i].y-yfix)\*cos(90);

}

Canvas->Polygon(tPoints7,7);

\*/

TPoint tPoints6[4];

tPoints6[0].x = 20;

tPoints6[0].y = 20;

tPoints6[1].x = 50;

tPoints6[1].y = 20;

tPoints6[2].x = 50;

tPoints6[2].y = 50;

tPoints6[3].x = 20;

tPoints6[3].y = 50;

Canvas->Polygon(tPoints6,3);

Sleep(1000);

Canvas->Brush->Color = clBlack;

Canvas->Polygon(tPoints6,3);

Canvas->Brush->Color = clYellow;

TPoint tPoints7[4];

double xfix=35;

double yfix=35;

tPoints7[0].x=double(xfix+(tPoints6[0].x-xfix)\*cos(45)-(tPoints6[0].y-yfix)\*sin(45));

tPoints7[0].y=double(yfix+(tPoints6[0].x-xfix)\*sin(45)+(tPoints6[0].y-yfix)\*cos(45));

tPoints7[1].x=double(xfix+(tPoints6[1].x-xfix)\*cos(45)-(tPoints6[1].y-yfix)\*sin(45));

tPoints7[1].y=double(yfix+(tPoints6[1].x-xfix)\*sin(45)+(tPoints6[1].y-yfix)\*cos(45));

tPoints7[2].x=double(xfix+(tPoints6[2].x-xfix)\*cos(45)-(tPoints6[2].y-yfix)\*sin(45));

tPoints7[2].y=double(yfix+(tPoints6[2].x-xfix)\*sin(45)+(tPoints6[2].y-yfix)\*cos(45));

tPoints7[3].x=double(xfix+(tPoints6[3].x-xfix)\*cos(45)-(tPoints6[3].y-yfix)\*sin(45));

tPoints7[3].y=double(yfix+(tPoints6[3].x-xfix)\*sin(45)+(tPoints6[3].y-yfix)\*cos(45));

Canvas->Polygon(tPoints7,3);

TPoint tPoints12[8];

for(i=0;i<183;i++)

{

Canvas->Brush->Color = clYellow;

tPoints12[0].x = 146;

tPoints12[0].y = 404-i;

tPoints12[1].x = 150;

tPoints12[1].y = 411-i;

tPoints12[2].x = 160;

tPoints12[2].y = 416-i;

tPoints12[3].x = 170;

tPoints12[3].y = 411-i;

tPoints12[4].x = 174;

tPoints12[4].y = 404-i;

tPoints12[5].x = 167;

tPoints12[5].y = 396-i;

tPoints12[6].x = 160;

tPoints12[6].y = 404-i;

tPoints12[7].x = 153;

tPoints12[7].y = 396-i;

Canvas->Polygon(tPoints12,7);

Sleep(9);

}

Canvas->Brush->Color = clBlack;

Canvas->Polygon(tPoints7,7);

TPoint tPoints9[8];

Canvas->Brush->Color = clYellow;

tPoints9[0].x = 146/1.01;

tPoints9[0].y = 221;

tPoints9[1].x = 150/1.01;

tPoints9[1].y = 228;

tPoints9[2].x = 160;

tPoints9[2].y = 233\*1.01;

tPoints9[3].x = 170\*1.01;

tPoints9[3].y = 228;

tPoints9[4].x = 174\*1.01;

tPoints9[4].y = 221;

tPoints9[5].x = 167\*1.01;

tPoints9[5].y = 213;

tPoints9[6].x = 160;

tPoints9[6].y = 221\*1.01;

tPoints9[7].x = 153/1.01;

tPoints9[7].y = 213;

Canvas->Polygon(tPoints9,7);

TPoint tPoints8[8];

for(i=0;i<200;i++)

{

Canvas->Brush->Color = clYellow;

tPoints8[0].x = tPoints9[0].x+i;

tPoints8[0].y = tPoints9[0].y;

tPoints8[1].x = tPoints9[1].x+i;;

tPoints8[1].y = tPoints9[1].y;

tPoints8[2].x = tPoints9[2].x+i;;

tPoints8[2].y = tPoints9[2].y;

tPoints8[3].x = tPoints9[3].x+i;;

tPoints8[3].y = tPoints9[3].y;

tPoints8[4].x = tPoints9[4].x+i;;

tPoints8[4].y = tPoints9[4].y;

tPoints8[5].x = tPoints9[5].x+i;;

tPoints8[5].y = tPoints9[5].y;

tPoints8[6].x = tPoints9[6].x+i;;

tPoints8[6].y = tPoints9[6].y;

tPoints8[7].x = tPoints9[7].x+i;;

tPoints8[7].y = tPoints9[7].y;

Canvas->Polygon(tPoints8,7);

Sleep(9);

}

Canvas->Brush->Color = clBlack;

Canvas->Polygon(tPoints8,7);

Canvas->Brush->Color = clYellow;

tPoints9[0].x = tPoints8[0].x\*1.01;

tPoints9[0].y = tPoints8[0].y;

tPoints9[1].x = tPoints8[1].x\*1.01;

tPoints9[1].y = tPoints8[1].y;

tPoints9[2].x = tPoints8[2].x;

tPoints9[2].y = tPoints8[2].y/1.01;

tPoints9[3].x = tPoints8[3].x/1.01;

tPoints9[3].y = tPoints8[3].y;

tPoints9[4].x = tPoints8[4].x/1.01;

tPoints9[4].y = tPoints8[4].y;

tPoints9[5].x = tPoints8[5].x/1.01;

tPoints9[5].y = tPoints8[5].y;

tPoints9[6].x = tPoints8[6].x;

tPoints9[6].y = tPoints8[6].y/1.01;

tPoints9[7].x = tPoints8[7].x\*1.01;

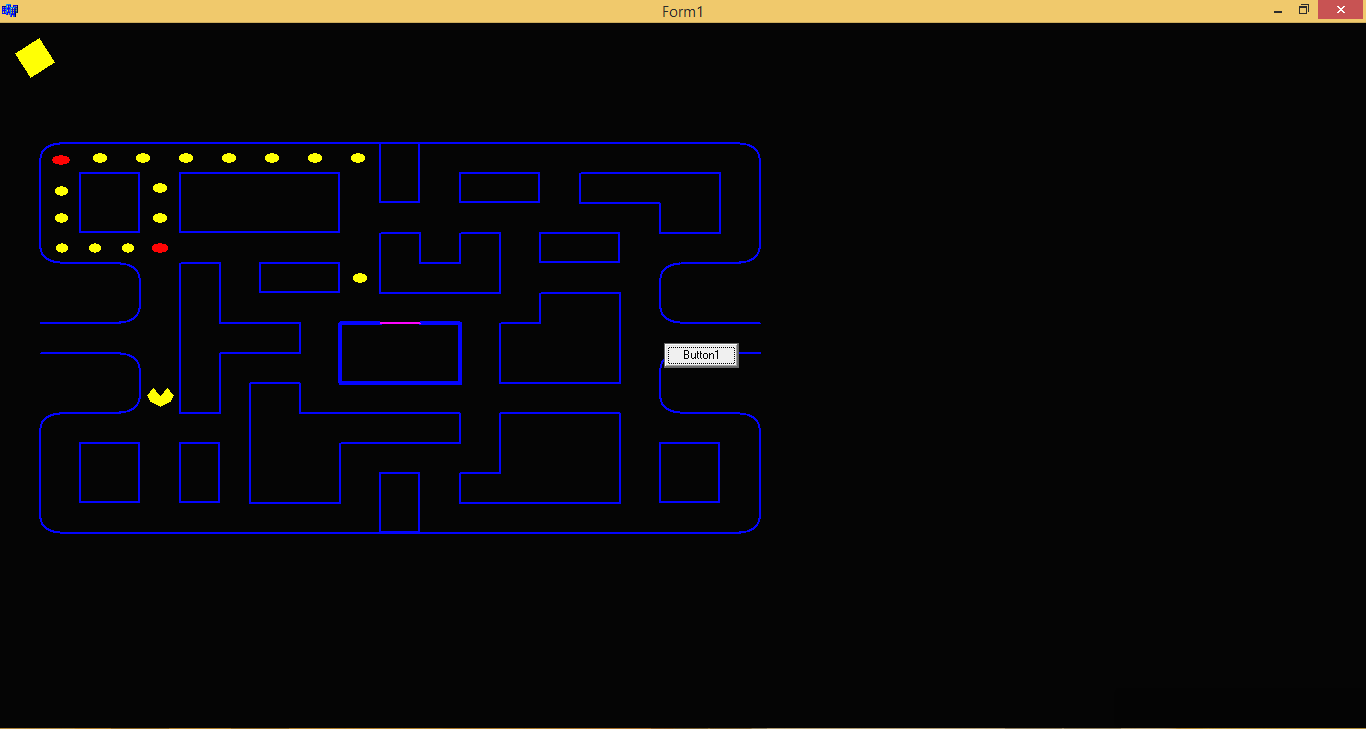
tPoints9[7].y = tPoints8[7].y;

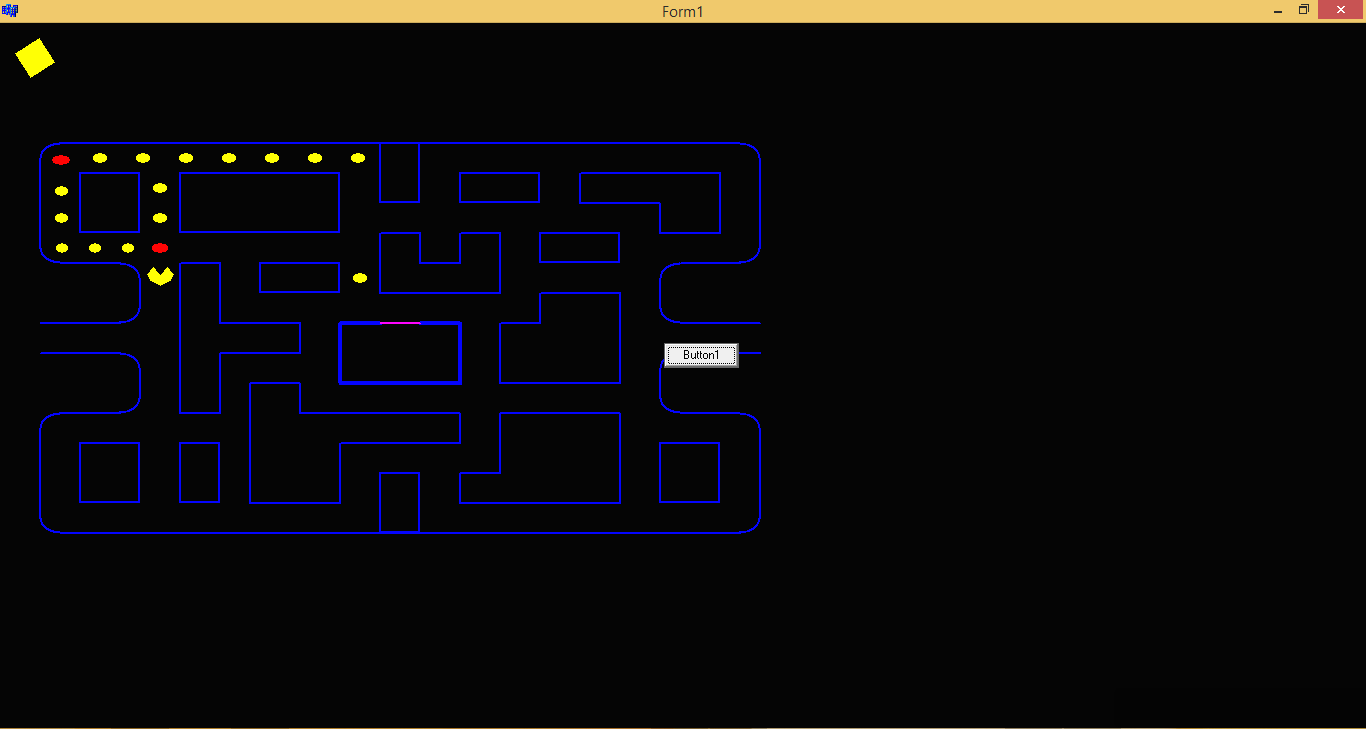
Canvas->Polygon(tPoints9,7);

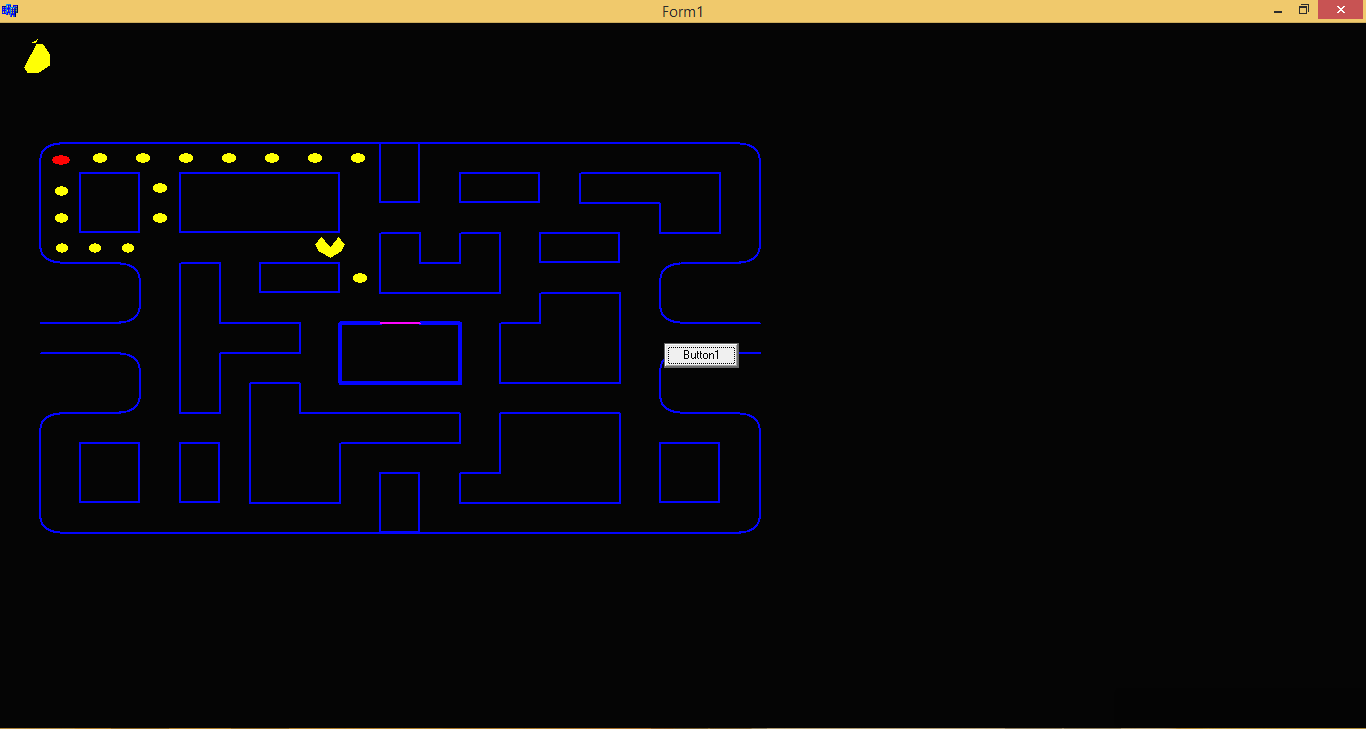
}

//---------------------------------------------------------------------------

Скриншоты анимации:







**Вывод**

В данной работе были применены на практике основные преобразования в 2D графике, а именно масштабирование перемещение и вращение. Для реализации анимации были использованы циклы вместе с командой Sleep().