///\*FLEVIA FARGOSE

SE A -30

Program to implement 2D Transformation .\*///

import java.io.\*;

import java.util.\*;

import java.applet.\*;

import java.awt.\*;

/\*<applet code="Transformation.class" width=800 height=800>

</applet>

\*/

public class Transformation extends Applet

{

int n;

int x[]=new int[10];

int y[]=new int[10];

float xo[]=new float[10];

float yo[]=new float[10];

double pi=3.14;

String p="TRANSLATION";

String q="SCALING";

String r="ROTATION";

String s="REFLECTION";

int ch,tx,ty,sx,sy,ang,i,vch,gl;

String c;

Scanner sc = new Scanner(System.in)

public void init()

{ }

public void paint(Graphics g)

{

g.setColor(Color.red);

g.drawLine(400,0,400,800);

g.drawLine(0,400,800,400);

g.drawString(p,30,30);

g.drawString(q,430,30);

g.drawString(r,30,430);

g.drawString(s,430,430);

g.setColor(Color.black);

do

{

System.out.print("Enter the no. of vertices : " );

n = sc.nextInt();

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

{

System.out.println("Enter co-ordinates for "+(i+1)+": ");

x[i] = sc.nextInt();

y[i] = sc.nextInt();

}

for(i=0;i<n-1;i++)

g.drawLine(x[i],y[i],x[i+1],y[i+1]);

g.drawLine(x[i],y[i],x[0],y[0]);

System.out.print("\n 1: Translation \n 2:Scaling \n 3: Rotation \n4:reflexion \n Enter your choice : ") ;

ch=sc.nextInt();

switch(ch)

{

case 1 :

System.out.print("Enter <Tx> & <Ty> value : ");

tx=sc.nextInt();

ty=sc.nextInt();

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

{

xo[i]=x[i]+tx;

yo[i]=y[i]+ty;

}

break;

case 2 :

System.out.print("Enter <Sx> & <Sy>value : ");

sx=sc.nextInt();

sy=sc.nextInt();

System.out.print("Enter Reference Vertex : ");

vch=sc.nextInt();

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

{

xo[i]=(x[i]-x[vch-1])\*sx+x[vch-1];

yo[i]=(y[i]-y[vch-1])\*sy+y[vch-1];

}

break;

case 3 :

double t;

System.out.print("Enter Angle : ");

t=sc.nextInt();

System.out.print("Enter Reference Vertex : ");

vch=sc.nextInt();

t=t\*(pi/180);

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

{

xo[i]=((x[i]-x[vch-1])\*(float)Math.cos(t))-

((y[i]-y[vch-1])\*(float)Math.sin(t))+x[vch-1];

yo[i]=((x[i]-x[vch-1])\*(float)Math.sin(t))-

((y[i]-y[vch-1])\*(float)Math.cos(t))+y[vch-1];

}

break;

case 4:

System.out.println("if abt x axis enter 1 and for y axis enter2");

gl=sc.nextInt();

if(gl==1)

{

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

{

xo[i]=x[i];

yo[i]=y[i]+100;

}

}

if(gl==2)

{

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

{

xo[i]=x[i]+100;

yo[i]=y[i];

}

}

break;

default:

System.out.println("wrong entry");

break;

}

g.setColor(Color.blue);

for(i=0;i<n-1;i++)

g.drawLine((int)xo[i],(int)yo[i],(int)xo[i+1],(int)yo[i+1]);

g.drawLine((int)xo[i],(int)yo[i],(int)xo[0],(int)yo[0]); System.out.print("Do you wanna coninue? y/n : ") ;

c = sc.next();

repaint();

g.setColor(Color.black);

}while(c.equals("y"));

}

}

/\*Output:

D:\Flevia 30>javac Transformation.java

D:\Flevia 30>appletviewer Transformation.java

Enter the no. of vertices : 4

Enter co-ordinates for 1:

70

50

Enter co-ordinates for 2:

140

50

Enter co-ordinates for 3:

140

120

Enter co-ordinates for 4:

70

120

1: Translation

2:Scaling

3: Rotation

4:reflexion

Enter your choice : 1

Enter <Tx> & <Ty> value :

100

100

Do you wanna coninue? y/n : y

Enter the no. of vertices : 4

Enter co-ordinates for 1:

470

50

Enter co-ordinates for 2:

540

50

Enter co-ordinates for 3:

540

120

Enter co-ordinates for 4:

470

120

1: Translation

2:Scaling

3: Rotation

4:reflexion

Enter your choice : 2

Enter <Sx> & <Sy>value : 2

2

Enter Reference Vertex : 1

Do you wanna coninue? y/n : y

Enter the no. of vertices : 4

Enter co-ordinates for 1:

70

450

Enter co-ordinates for 2:

140

450

Enter co-ordinates for 3:

140

520

Enter co-ordinates for 4:

70

520

1: Translation

2:Scaling

3: Rotation

4:reflexion

Enter your choice : 3

Enter Angle : 150

Enter Reference Vertex : 3

Do you wanna coninue? y/n : y

Enter the no. of vertices : 4

Enter co-ordinates for 1:

470

450

Enter co-ordinates for 2:

540

450

Enter co-ordinates for 3:

540

520

Enter co-ordinates for 4:

470

520

1: Translation

2:Scaling

3: Rotation

4:reflexion

Enter your choice : 4

if abt x axis enter 1 and for y axis enter2

1

Do you wanna coninue? y/n : n

