**A**

**LAB REPORT**

**ON**

**Computer Graphics**

**By**

**Rupesh Mahat**

**Exam Roll No: 8659/18**



**Submitted to:**

**Rakesh Shrestha**

**Department of Computer Science**

**Kantipur College of Management and Information Technology**

**In partial fulfillment of the requirements for the Course**

**Computer Graphics**

**Mid Baneshwor, Kathmandu**

**December 2022**

# Write a program to use DLD Algorithm.

## Source Code

import java.awt.\*;

import java.awt.event.\*;

//import java.awt.geom.Ellipse2D;

import javax.swing.\*;

public class DLDinput extends JFrame implements ActionListener

{

JLabel msgx1, msgx2, msgy1, msgy2;

JTextField tx1, tx2, ty1, ty2;

JButton bt1;

public DLDinput()

{

setLayout(new FlowLayout());

msgx1=new JLabel("Enter x1");

msgy1=new JLabel("Enter y1");

msgx2=new JLabel("Enter x2");

msgy2=new JLabel("Enter y2");

tx1 = new JTextField(20);

ty1 = new JTextField(20);

tx2 = new JTextField(20);

ty2 = new JTextField(20);

bt1=new JButton("Draw");

bt1.addActionListener(this);

add(msgx1);

add(tx1);

add(msgy1);

add(ty1);

add(msgx2);

add(tx2);

add(msgy2);

add(ty2);

add(bt1);

setTitle("RUPESH MAHAT");

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setSize(1024,760);

setVisible(true);

}

public static void main(String[] args)

{

new DLDinput();

}

@Override

public void actionPerformed(ActionEvent ae)

{

new Drawing(tx1.getText(),ty1.getText(),tx2.getText(),ty2.getText());

}

}

Drawing

import java.awt.\*;

//import java.awt.event.\*;

import java.awt.geom.Ellipse2D;

import javax.swing.\*;

public class Drawing extends JFrame

{

int x0,y0,r,p,x1,x2,y1,y2;

float dy, dx, x,y,m;

public Drawing(String x1,String y1,String x2, String y2)

{

JPanel p=new JPanel();

getContentPane().add(p);

setTitle("RUPESH MAHAT");

setSize(1024,768);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setVisible(true);

this.x1=Integer.parseInt(x1);

this.y1=Integer.parseInt(y1);

this.x2=Integer.parseInt(x2);

this.y2=Integer.parseInt(y2);

}

@Override

public void paint(Graphics g)

{

super.paint(g);

Graphics2D g2=(Graphics2D) g;

dx = x2-x1;

dy = y2-y1;

if(Math.abs(dx)>Math.abs(dy))

{

m=dy/dx;

if(dx>0)

{

for(inti=x1; i<=x2; i++)

{

y = y1+(m\*(i-y1));

g2.draw(new Ellipse2D.Float(i,y,1,1));

}

}

else

{

for(inti=y2; i<=y1; i++)

{

y=y1+(m\*(i-y1));

g2.draw(new Ellipse2D.Float(i,y,1,1));

}

}

}

else{

if(dy>0)

{

m=dx/dy;

for(inti=y1;i<=y2;i++)

{

x=x1+(m \*(i-y1));

g2.draw(new Ellipse2D.Float(x,i,1,1));

}

}

else

{

for(inti=y1;i>=y2;i--)

{

x=x1+(m\*(i-y1));

g2.draw(new Ellipse2D.Float(x,i,1,1));

}

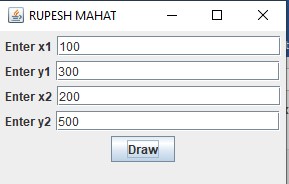
}

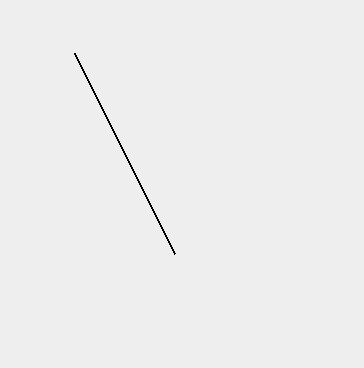
}

}

}

## Output Window





# Write a program to use DDA algorithm.

## Source code

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

class DDA extends JFrame implements ActionListener

{

JLabel msgx1, msgx2, msgy1, msgy2;

JTextField tx1, tx2, ty1, ty2;

JButton bt1;

public DDA()

{

setLayout(new FlowLayout());

msgx1=new JLabel("Enter x1");

msgy1=new JLabel("Enter y1");

msgx2=new JLabel("Enter x2");

msgy2=new JLabel("Enter y2");

tx1 = new JTextField(20);

ty1 = new JTextField(20);

tx2 = new JTextField(20);

ty2 = new JTextField(20);

bt1=new JButton("Draw");

bt1.addActionListener(this);

add(msgx1);

add(tx1);

add(msgy1);

add(ty1);

add(msgx2);

add(tx2);

add(msgy2);

add(ty2);

add(bt1);

set Title("RUPESH MAHAT");

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setSize(1024,760);

setVisible(true);

}

public static void main(String[] args)

{

new DDA();

}

public void actionPerformed(ActionEvent ae)

{

new ddadrawing(tx1.getText(),ty1.getText(),tx2.getText(),ty2.getText());

}

}

import java.awt.\*;

import java.awt.geom.Ellipse2D;

import javax.swing.\*;

public class ddadrawing extends JFrame

{

inti,m,x1,x2,y1,y2,step,dy, dx;

double x, y, xinc, yinc;

public ddadrawing(String x1,String y1,String x2, String y2)

{

JPanel p=new JPanel();

getContentPane().add(p);

setTitle("Example");

setSize(1024,768);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setVisible(true);

this.x1=Integer.parseInt(x1);

this.y1=Integer.parseInt(y1);

this.x2=Integer.parseInt(x2);

this.y2=Integer.parseInt(y2);

}

public void paint(Graphics g)

{

super.paint(g);

Graphics2D g2=(Graphics2D) g;

dx = x2-x1;

dy = y2-y1;

if(Math.abs(dx)>Math.abs(dy))

{

step = Math.abs(dx);

}

else

{

step = Math.abs(dy);

}

xinc=dx/step;

yinc=dy/step;

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

{

x=x+xinc;

y=y+yinc;

g2.draw(new Ellipse2D.Double(x,y,1,1));

}

for(m=0;m<step;m++)

{

x=x+xinc;

y=y+yinc;

g2.draw(new Ellipse2D.Double(x,y,1,1));

}

if(Math.abs(dx)<Math.abs(dy))

{

step = Math.abs(dy);

}

else

{

step = Math.abs(dx);

}

xinc=dx/step;

yinc=dy/step;

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

{

x=x+xinc;

y=y+yinc;

g2.draw(new Ellipse2D.Double(x,y,1,1));

}

for(m=0;m<step;m++)

{

x=x+xinc;

y=y+yinc;

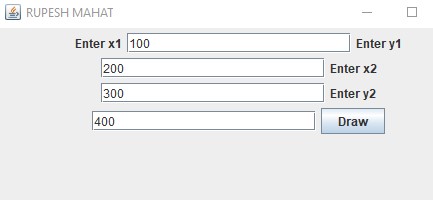
g2.draw(new Ellipse2D.Double(x,y,1,1));

}

}

}

## Output window





# Write a program to draw a line using bresanhams Line A lgorithm .

## Source code

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

public class bresenhams extends JFrame implements ActionListener

{

    JLabel m1,m2,m3,m4;

    JTextField x1,x2,y1,y2;

    JButtonbtn;

    public bresenhams()

    {

        m1=new JLabel("enter x1");

        x1=new JTextField(10);

        m2=new JLabel("enter x2");

        x2=new JTextField(10);

        m3=new JLabel("enter y1");

        y1=new JTextField(10);

        m4=new JLabel("enter y2");

        y2= new JTextField(10);

        btn=new JButton("draw line");

        btn.addActionListener(this);

        add(m1);add(x1);add(m2);add(x2);

        add(m3);add(y1);add(m4);add(y2);

        add(btn);

        setTitle("bresenhams algorithm");

        setSize(700,500);

        setLayout(new FlowLayout());

        setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

        setVisible(true);

    }

    public void actionPerformed(ActionEvent e)

    {

        new bline(x1.getText(), x2.getText(), y1.getText(), y2.getText());

    }

    public static void main(String[] args) {

        new bresenhams();

    }

}

import javax.swing.\*;

import java.awt.\*;

//import java.awt.event.\*;

import java.awt.geom.Ellipse2D;

public class line extends JFrame

{

    int x1,x2,y1,y2;

    intdx,dy,p0=0,i;

    public line(String x1,String x2,String y1, String y2)

    {

        JPanel p=new JPanel();

        getContentPane().add(p);

        setTitle("bresenhams algorithm");

        setSize(900,900);

        setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

        setVisible(true);

        this.x1=Integer.parseInt(x1);

        this.y1=Integer.parseInt(y1);

        this.x2=Integer.parseInt(x2);

        this.y2=Integer.parseInt(y2);

    }

    public void paint(Graphics g)

    {

        super.paint(g);

        Graphics2D g2=(Graphics2D) g;

        dx=x2-x1;

        dy=y2-y1;

        g2.draw(new Ellipse2D.Float(x1,y1,1,1));

        p0=2\*dy-dx;

        for(i=0; i<dx; i++)

        {

            if(p0<0)

            {

                x1=x1+1;

                //g2.draw(new Ellipse2D.Float(x1,y1,2,2));

                g2.drawLine(x1, y1, x1, y1);

                p0=p0+2\*dy;

               // System.out.println(p0);

            }

            else

            {

                x1=x1+1;

                y1=y1+1;

                //g2.draw(new Ellipse2D.Float(x1,y1,2,2));

                g2.drawLine(x1, y1, x1, y1);

                p0=p0+2\*dy-2\*dx;

              //  System.out.println(p0);

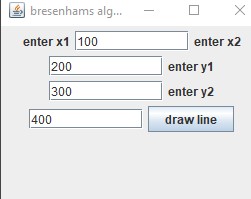
            }

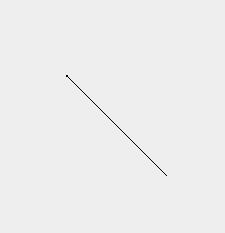
        }

    }

}

## Output Window





# Write a program to draw traingle .

## Source code

import javax.swing.\*;

import java.awt.\*;

public class Triangle extends JComponent {

@Override

protected void paintComponent(Graphics g) {

super.paintComponent(g);

g.drawLine(120, 130, 280, 130);

g.drawLine(120, 130, 200, 65);

g.drawLine(200, 65, 280, 130);

}

public Dimension getPreferredSize() {

return new Dimension(500, 300);

}

public static void main(String[] args) {

JFramejFrame = new JFrame();

jFrame.add(new Triangle());

jFrame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

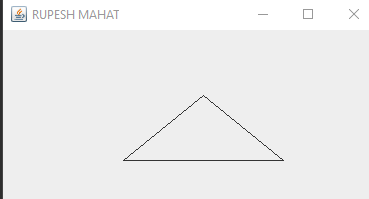
jFrame.setTitle(“RUPESH MAHAT”);

jFrame.setVisible(true);

}

}

## Output window



# Write a program to draw a circle .

## Source code

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

import java.awt.geom.Ellipse2D;

public class circleprogram extends JFrame

{

    int r=60,xc=50,yc=50,x=0,y;

    int p;

    public circleprogram()

    {

        JPanel p=new JPanel();

        getContentPane().add(p);

        setTitle("RUPESH MAHAT");

        setSize(900,900);

        setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

        setVisible(true);

    }

    public void paint(Graphics g)

    {

        intcPointx = 350+xc;

        intcPointy = 350+yc;

        y=r;

        super.paint(g);

        Graphics2D g2=(Graphics2D) g;

        g2.draw(new Ellipse2D.Double(cPointx,cPointy+r,10,10));

        g2.draw(new Ellipse2D.Double(cPointx,cPointy-r,10,10));

        g2.draw(new Ellipse2D.Double(cPointy+r,cPointx,10,10));

        g2.draw(new Ellipse2D.Double(cPointy-r,cPointx,10,10));

        p=1-r;

        do

        {

            if(p<0)

            {

                x=x+1;

                p=p+(2\*x)+1;

                    x=x+1;

                y=y-1;

                p=p+(2\*x+1)+1-(2\*y+1);

            }

            g2.draw(new Ellipse2D.Double(cPointx+x,cPointy+y,10,10));

            g2.draw(new Ellipse2D.Double(cPointx+x,cPointy-y,10,10));

            g2.draw(new Ellipse2D.Double(cPointx-x,cPointy+y,10,10));

            g2.draw(new Ellipse2D.Double(cPointx-x,cPointy-y,10,10));

            g2.draw(new Ellipse2D.Double(cPointx+y,cPointy+x,10,10));

            g2.draw(new Ellipse2D.Double(cPointx+y,cPointy-x,10,10));

            g2.draw(new Ellipse2D.Double(cPointx-y,cPointy+x,10,10));

            g2.draw(new Ellipse2D.Double(cPointx-y,cPointy-x,10,10));

}

            else

            {

                               }

        while(x<=y);

    }

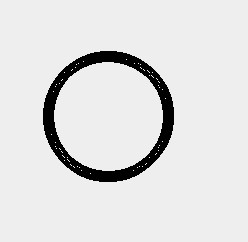
    public static void main(String[] args) {

        new circleprogram();

    }

}

## Output Window



# Write a program to draw a house.

## Source code

import javax.swing.\*;

import java.awt.\*;

public class housedraw extends JFrame

{

public housedraw()

{

JPanel p=new JPanel(); getContentPane().add(p);

setTitle("circle program"); setSize(500,500);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE); setVisible(true);

}

public void paint(Graphics g)

{ super.paint(g);

Graphics2D g2=(Graphics2D) g;

g2.drawRect(280, 130,200, 250);

//recatngle

g2.drawRect( 120, 130, 160, 250); //rect to square DOOR g2.drawRect(150, 280,100, 100); //window

// g2.setColor(Color.green); g2.drawRect(170, 150,50, 50);

// g2.setPaint(color.pink);

//circle

g2.drawOval(180, 80, 40, 40);

//traingle

g2.drawLine(120, 130, 280, 130);

g2.drawLine(120, 130, 200, 65);

g2.drawLine(200, 65, 280, 130);

//next door

g2.drawRect( 340, 280,100, 100);

//next window //window

g2.drawRect( 340, 150,50, 50);

g2.drawRect( 380, 150,50, 50);

}

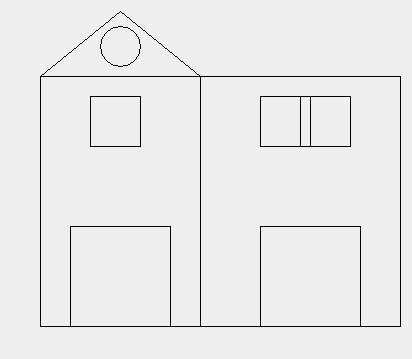
public static void main(String[] args) {

new housedraw();

}

}

## Output window



# Write a program to translate a line .

## Source code

import java.awt.Color;

import java.awt.Graphics;

import javax.swing.JFrame;

class translation extends JFrame{

int x1 = 60;

int y1 = 60;

int x2 = 150;

int y2 = 150;

inttx = 200;

int ty = 200;

int px1,px2,py1,py2;

public translation()

{

setTitle(“RUPESH MAHAT");

setSize(500,500);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

px1 = x1+tx;

px2 = x2+tx;

py1 = y1+ty;

py2 = y2+ty;

setVisible(true);

}

public void paint(Graphics g)

{

g.drawLine(x1, y1, x2, y2);

g.setColor(Color.black);

g.drawLine(px1, py1, px2, py2);

}

public static void main(String args[])

{

new translation();

}

}

## Output Window



# Write a program to scaling a line .

## Source code

import java.awt.Color;

import java.awt.Graphics;

import javax.swing.\*;

class Scaling extends JFrame{

int x1 = 75;

int y1 = 75;

int x2 = 200;

int y2 = 200;

intsx = 4;

intsy = 4;

int px1,px2,py1,py2;

public Scaling()

{

setTitle("RUPESH MAHAT");

setSize(500,500);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

px1 = x1\*sx;

px2 = x2\*sx;

py1 = y1\*sy;

py2 = y2\*sy;

setVisible(true);

}

public void paint(Graphics g)

{

g.drawLine(x1, y1, x2, y2);

g.setColor(Color.black);

g.drawLine(px1, py1, px2, py2);

}

public static void main(String args[])

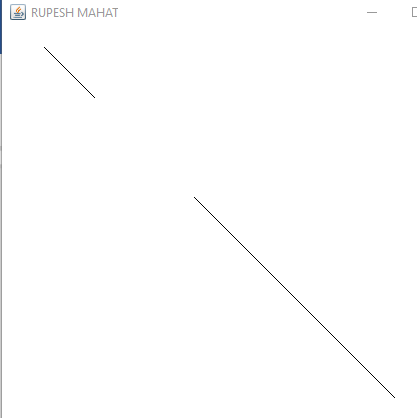
{

new Scaling();

}

}

## Output window



# Write a program to shear a polygon in x-axis.

## Source code

import java.awt.Graphics;

import javax.swing.JFrame;

class Shear\_x extends JFrame{

int x1 = 100;

int y1 = 0;

int x2 = 200;

int y2 = 0;

int x3 = 200;

int y3 = 200;

int x4 = 100;

int y4 = 200;

intshx = 2;

int px1,px2,px3,px4;

public Shear\_x()

{

setTitle("RUPESH MAHAT");

setSize(1000,1000);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

px1 = x1 + y1\*shx;

px2 = x2 + y2\*shx;

px3 = x3+ y3 \*shx;

px4 = x4+ y4 \* shx;

setVisible(true);

}

public void paint(Graphics g)

{

//before shear

g.drawLine(x1, y1, x2, y2);

g.drawLine(x2, y2, x3, y3);

g.drawLine(x3, y3, x4, y4);

g.drawLine(x4, y4, x1, y1);

//after shear

g.drawLine(px1, y1, px2, y2);

g.drawLine(px2, y2, px3, y3);

g.drawLine(px3, y3, px4, y4);

g.drawLine(px4, y4, px1, y1);

}

public static void main(String args[])

{

new Shear\_x();

}

}

## Output window



# Write a program to reflect the line on the origin.

## Source code

import java.awt.Graphics;

import javax.swing.JFrame;

class originreflection extends JFrame{

int x1 = 150;

int y1 = 50;

int x2 = 150;

int y2 = 150;

int px1,px2,py1,py2;

public originreflection()

{

setTitle("RUPESH MAHAT");

setSize(500,500);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

px1 = -x1+550;

px2 = -x2+550;

py1 = -y1+550;

py2 = -y2+550;

setVisible(true);

}

public void paint(Graphics g)

{

g.drawLine(x1, y1, x2, y2);

g.drawLine(px1, py1, px2, py2);

}

public static void main(String args[])

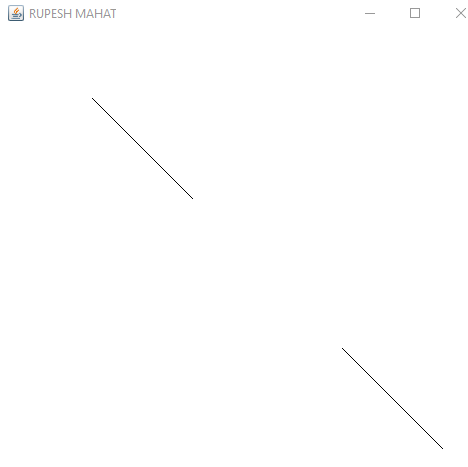
{

new originreflection();

}

}

## Output window



# Write a program to Rotation the line.

## Source code

import java.awt.Graphics;

import javax.swing.JFrame;

class RotationCounterClockwise extends JFrame{

int x1 = 100;

int y1 = 100;

int x2 = 50;

int y2 = 50;

int angle = 15;

double t = angle \* Math.PI / 180;

int px1,px2,py1,py2;

public RotationCounterClockwise()

{

setTitle("RUPESH MAHAT");

setSize(500,500);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

px1 = (int)(x1 \* Math.cos(t) - y1 \* Math.sin(t));

px2 = (int)(x2\*Math.cos(t) - y2 \* Math.sin(t));

py1 = (int)(y1\*Math.cos(t) + x1\*Math.sin(t));

py2 = (int)(y2\*Math.cos(t) + x2\*Math.sin(t));

setVisible(true);

}

public void paint(Graphics g)

{

g.drawLine(x1, y1, x2, y2);

g.drawLine(px1, py1, px2, py2);

}

public static void main(String args[])

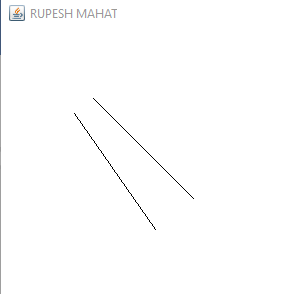
{

new RotationCounterClockwise();

}

}

## Output window



# Write a program to Reflection a line on x=y.

## Source code

import java.awt.Graphics;

import javax.swing.JFrame;

class Reflection\_Y extends JFrame{

int x1 = 50;

int y1 = 50;

int x2 = 50;

int y2 = 150;

int px1,px2;

public Reflection\_Y()

{

setTitle("Reflection of line on Y axis");

setSize(500,500);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

px1 = -x1+150;

px2 = -x2+150;

setVisible(true);

}

public void paint(Graphics g)

{

g.drawLine(x1, y1, x2, y2);

g.drawLine(px1, y1, px2, y2);

}

public static void main(String args[])

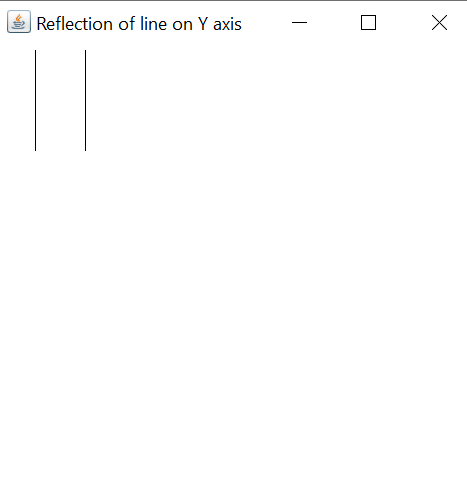
{

new Reflection\_Y();

}

}

## Output window



# Write a program to reflection a line on Y= - X

## Source code

import java.awt.Graphics;

import javax.swing.JFrame;

class Reflection\_Y\_equals\_minus\_X extends JFrame{

int x1 = 50;

int y1 = 50;

int x2 = 70;

int y2 = 250;

int px1,px2,py1,py2;

public Reflection\_Y\_equals\_minus\_X()

{

setTitle("Reflection of line on Y = -X");

setSize(500,500);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

px1 = -y1+350;

px2 = -y2+350;

py1 = -x1+350;

py2 = -x2+350;

setVisible(true);

}

public void paint(Graphics g)

{

g.drawLine(x1, y1, x2, y2);

g.drawLine(px1, py1, px2, py2);

}

public static void main(String args[])

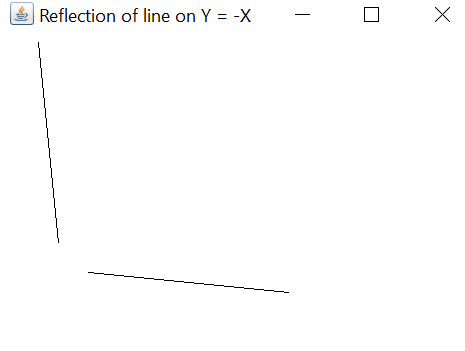
{

new Reflection\_Y\_equals\_minus\_X();

}

}

## Output window



Contents

[1 DLD Algorithm. 1](#_Toc98173170)

[1.1 Source Code 1](#_Toc98173171)

[1.2 Output Window 5](#_Toc98173172)

[2 DDA algorithm. 5](#_Toc98173173)

[2.1 Source code 5](#_Toc98173174)

[2.2 Output window 10](#_Toc98173175)

[3 Write a program to draw a line using bresanhams . 10](#_Toc98173176)

[3.1 Source code 10](#_Toc98173177)

[3.2 Output Window 13](#_Toc98173178)

[4 Write a program to draw traingle . 14](#_Toc98173179)

[4.1 Source code 14](#_Toc98173180)

[4.2 Output window 15](#_Toc98173181)

[5 Write a program to draw a circle . 15](#_Toc98173182)

[5.1 Source code 15](#_Toc98173183)

[5.2 Output Window 18](#_Toc98173184)

[6 Write a program to draw a house. 18](#_Toc98173185)

[6.1 Source code 18](#_Toc98173186)

[6.2 Output window 20](#_Toc98173187)

[7 Write a program to translate a line . 20](#_Toc98173188)

[7.1 Source code 20](#_Toc98173189)

[7.2 Output Window 21](#_Toc98173190)

[8 Write a program to scaling a line . 23](#_Toc98173191)

[8.1 Source code 23](#_Toc98173192)

[8.2 Output window 24](#_Toc98173193)

[9 Write a program to shear a polygon in x-axis. 25](#_Toc98173194)

[9.1 Source code 25](#_Toc98173195)

[9.2 Output window 26](#_Toc98173196)

[10 Write a program to reflect the line on the origin. 27](#_Toc98173197)

[10.1 Source code 27](#_Toc98173198)

[10.2 Output window 28](#_Toc98173199)

[11 Rotation 29](#_Toc98173200)

[11.1 Source code 29](#_Toc98173201)

[11.2 Output window 30](#_Toc98173202)