

```

// LineArt.java
// Student version of the Lab06 Line Art Graphics Program assignment.
import java.awt.*;
import java.applet.*;
import java.awt.Color;
import java.util.Random;
import javax.swing.Timer;
import java.awt.event.*;

public class LineArt extends Applet implements ActionListener
{

    public final Timer timer = new Timer(1000/12, this);

    public void paint(Graphics g)
    {

        if (!timer.isRunning()) {
            timer.start();
        }
        int width = 980;
        int height = 630;
        g.drawRect(10,10,width,height);
        //draw(g, 10, 10, width, height);
        //draw(g, 10+width/4,10+height/4, width/2, height/2);
        int x = 10;
        int y = 10;

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

            draw(g, x, y, width, height, 50);
            x = x+width/4;
            y = y+height/4;
            width = width/2;
            height = height/2;

        }

    }
    public static void draw(Graphics g, int x,int y, int width, int height, int q) {

        //x and y are the topleft corner of the rectangle, width and height are of the rect you are filling
        for (int i = 0; i<=q; i++) {
            Random randall = new Random();
            float r = randall.nextFloat();
            float gc = randall.nextFloat();
            float b = randall.nextFloat();

            g.setColor(new Color(r, gc, b));
            g.drawLine(x, (y+height)-i*((y+height)-y)/q, (x+width)+i*(x-(x+width))/q, (y+height));
            g.drawLine((x+width), (y+height)-i*((y+height)-y)/q, x+i*((x+width)-x)/q, (y+height));
            g.drawLine((x+width), y-i*(y-(y+height))/q, x+i*((x+width)-x)/q, y);
            g.drawLine(x, y-i*(y-(y+height))/q, (x+width)+i*(x-(x+width))/q, y);
        }

    }

    @Override
    public void actionPerformed(ActionEvent e) {
        // TODO Auto-generated method stub
        this.repaint();
    }

}

```