```java
import java.awt.Dimension;
import java.awt.Graphics;
import javax.swing.JPanel;
import javax.swing.JFrame;

public class LineArt extends JPanel {

    // Unique version ID for this class to ensure saved objects can be loaded safely
    private static final long serialVersionUID = 1L;

    // Initial width of height of the starting rectangle
    private static int linesNum = 75;
    private static int startX = 10;
    private static int startY = 10;
    private static int width = 980;
    private static int height = 630;

    // main method to launch the program as a standalone application - no need to
    // modify
    public static void main(String[] args) {
        LineArt panel = new LineArt();
        panel.setPreferredSize(new Dimension(width + 20, height + 20)); // content size window dimensions

        JFrame frame = new JFrame("Line Art"); // Title of frame
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.add(panel);
        frame.pack();
        frame.setVisible(true);
    }

    /**
     * Draw the four corners of line art.  Line art displays straight lines inside a rectangle from one side to
     * a perpendicular side. The lines must be drawn in such a way that both the starting points of the lines on
     * one side and the ending points on the other side are equi-distant along the sides.  The size of the rectangle
     * is 980 pixels wide by 630 pixels high.
     *
     * @param g the Graphics object used for drawing shapes, text, and images
     */
    public void drawLineArt(Graphics g, int linesNum, int startX, int startY, int width, int height) {

        // Draw the initial rectangle
        g.drawRect(startX, startY, width, height);

        // Draw bottom-left corner
        for (int i = 0; i<=linesNum; i++) {
            g.drawLine(startX + i * width / linesNum, startY + height, startX + width, height + startY - i * height / linesNum);
            g.drawLine(startX + width - i * width / linesNum, startY + height, startX, height + startY - i * height / linesNum);
            g.drawLine(startX + i * width / linesNum, startY, startX + width, startY + i * height / linesNum);
            g.drawLine(startX + width - i * width / linesNum, startY, startX, startY + i *
```

```
  height / linesNum);
            }
        }
    /**
     * Overrides JPanel's paintComponent method to perform custom drawing.
     *
     * @param g the Graphics object used for drawing shapes, text, and images
     */
    @Override
    protected void paintComponent(Graphics g) {
        super.paintComponent(g); // Clears the panel before drawing
        drawLineArt(g, linesNum, startX, startY, width, height);
        drawLineArt(g, 67, 255, 168, width/2, height/2);
        drawLineArt(g, 50, 378, 248, width/4, height/4);
    }

}

```