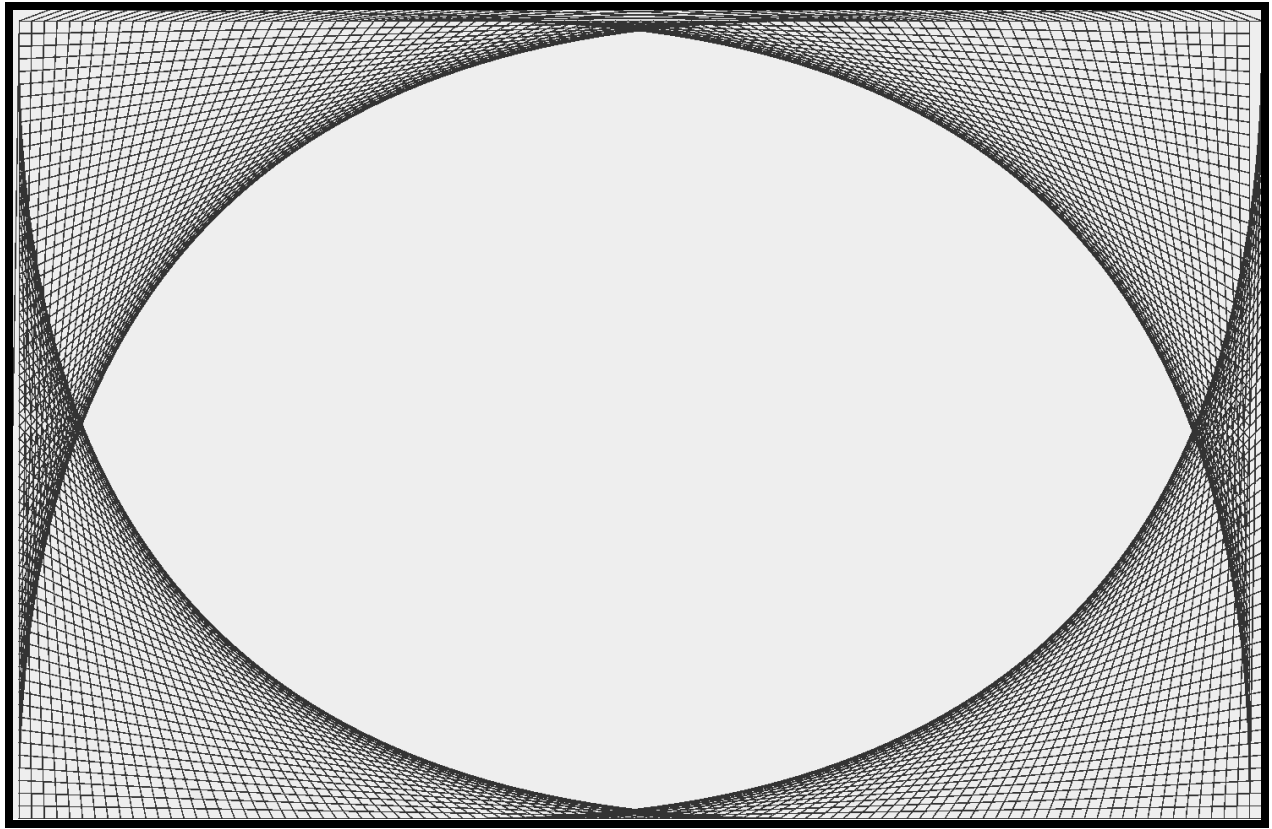




```
import java.awt.Dimension;
import java.awt.Graphics;
import javax.swing.JPanel;
import javax.swing.JFrame;

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 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) {

    // Draw the initial rectangle
    g.drawRect(10, 10, width, height);

    // Draw bottom-left corner
    int x = 10;
    int y = 10;

    for (x = 0; x <= 980; x += 10) {
        y = y + (10 * (980/630)) ;
        g.drawLine(10 , y, x , 640);
    }

    y = 10;

    for (x = 990; x >= 10; x -= 10) {
        y = y + (10 * (980/630));
        g.drawLine(990, y, x, 640);
    }
}

```

```
y = 640;
```

```
for (x = 990; x >= 10; x -= 10) {
```

```
    y = y - (10 * (980/630));
```

```
    g.drawLine(x, 10, 980, y);
```

```
}
```

```
y = 640;
```

```
for (x = 10; x <= 990; x += 10) {
```

```
    y = y - (10*(980/630));
```

```
    g.drawLine(x, 10, 10, y);
```

```
}
```

```
}
```

```
// Draw bottom-right corner
```

```
// Draw top-right corner
```

```
        // Draw top-left corner

/**
 * 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) {
    drawing    super.paintComponent(g); // Clears the panel before
                drawLineArt(g);

}

}
```