```
import java.awt.Graphics;
import java.applet.Applet;
import java.awt.Color;
import java.util.*;
import java.util.Random;

public class Star extends Applet {
	public void paint(Graphics g) {
		draw(g);
		draw(g);
		draw(g);
		draw(g);
		draw(g);
		draw(g);
		draw(g);
		draw(g);
		draw(g);
		draw(g);
	}

	public void draw(Graphics g) {
		// Draw maze
		Random peter = new Random();
		int[] nvalues = { 4, 5, 6, 9, 10, 12, 15, 18, 20 };
		int nI = peter.nextInt(nvalues.length);
		int n = nvalues[nI];
		double size = peter.nextInt(50) + 50;
		double xcenter = peter.nextInt(500)+100;
		double ycenter = peter.nextInt(500)+100;
		int[] xvalues = new int[2 * n];
		int[] yvalues = new int[2 * n];
		for (int i = 0; i < 2 * n; i++) {
			if (i % 2 == 0) {
				xvalues[i] = Math.round((int) (size *
Math.cos(Math.toRadians((double) 180 * i / (n))) + xcenter));
				yvalues[i] = Math.round((int) ((size *
Math.sin(Math.toRadians((double) 180 * i / (n)))) + ycenter));
			} else {
				xvalues[i] = Math.round((int) (size *
Math.cos(Math.toRadians((double) 180 * i / (n))) / 2 + xcenter));
				yvalues[i] = Math.round((int) (size *
Math.sin(Math.toRadians((double) 180 * i / (n))) / 2 + ycenter));
```

```
                }
            }
            switch (n) {
            case 4:
                g.setColor(Color.BLACK);
                break;
            case 5:
                g.setColor(Color.YELLOW);
                break;
            case 6:
                g.setColor(Color.RED);
                break;
            case 9:
                g.setColor(Color.BLUE);
                break;
            case 10:
                g.setColor(Color.GREEN);
                break;
            case 12:
                g.setColor(Color.ORANGE);
                break;
            case 15:
                g.setColor(Color.MAGENTA);
                break;
            case 18:
                g.setColor(Color.CYAN);
                break;
            case 20:
                g.setColor(Color.PINK);
                break;
            }
            g.fillPolygon(xvalues, yvalues, 2 * n);

        }
}
```