

Министерство образования Республики Беларусь
Учреждение образования
«Брестский государственный технический университет»
Кафедра ИИТ

Лабораторная работа №7
по дисциплине: **СПП**
Тема: Графика в Java

Выполнил
студент 3 курса
Корнаसेвич И. Д.

Проверил
Крощенко А. А.

Цель работы освоить возможности языка программирования Java в построении графических приложений

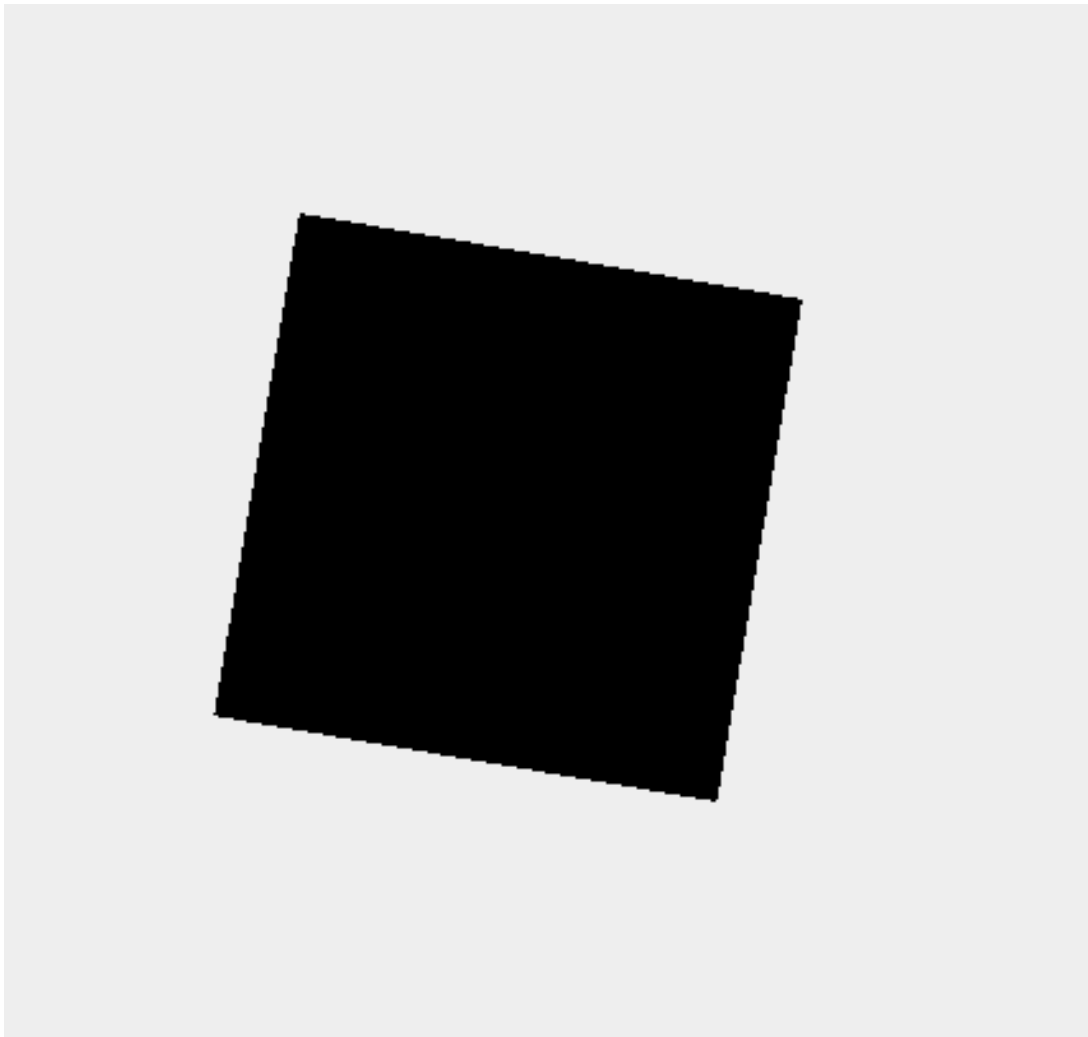
Задание 1 Изобразить четырехугольник, вращающийся в плоскости апплета вокруг своего центра тяжести.

Main.java

```
1 package main7.task1;
2
3 import javax.swing.*;
4
5
6 public final class Main {
7
8     public static void main(String[] args) {
9         var frame = new JFrame("Test");
10        var canvas = new TestCanvas();
11        canvas.setSize(450, 450);
12        frame.add(canvas);
13        frame.setSize(450, 450);
14        frame.setVisible(true);
15    }
16 }
```

TestCanvas.java

```
1 package main7.task1;
2
3 import java.awt.*;
4 import java.awt.geom.AffineTransform;
5
6 import static java.lang.Math.*;
7
8
9 public final class TestCanvas extends Canvas {
10
11     private final double angle = 0.001;
12
13     private final AffineTransform affineTransform = new AffineTransform(
14         cos(angle),
15         sin(angle),
16         -sin(angle),
17         cos(angle),
18         200 - 200 * cos(angle) + 200 * sin(angle),
19         200 - 200 * cos(angle) - 200 * sin(angle));
20
21     private Shape rectangle = new Rectangle(100, 100, 200, 200);
22
23     @Override
24     public void paint(Graphics g) {
25         while (true) {
26             var g2d = (Graphics2D) g;
27             g2d.setColor(Color.WHITE);
28             g2d.clearRect(0, 0, getWidth(), getHeight());
29             rectangle = affineTransform.createTransformedShape(rectangle);
30             g2d.setColor(Color.black);
31             g2d.fill(rectangle);
32             try {
33                 Thread.sleep(5);
34             } catch (InterruptedException e) {
35                 e.printStackTrace();
36             }
37         }
38     }
39 }
```



Задание 2 Реализовать построение заданного типа фрактала по варианту Везде, где это необходимо, предусмотреть ввод параметров, влияющих на внешний вид фрактала

Треугольная салфетка Серпинского

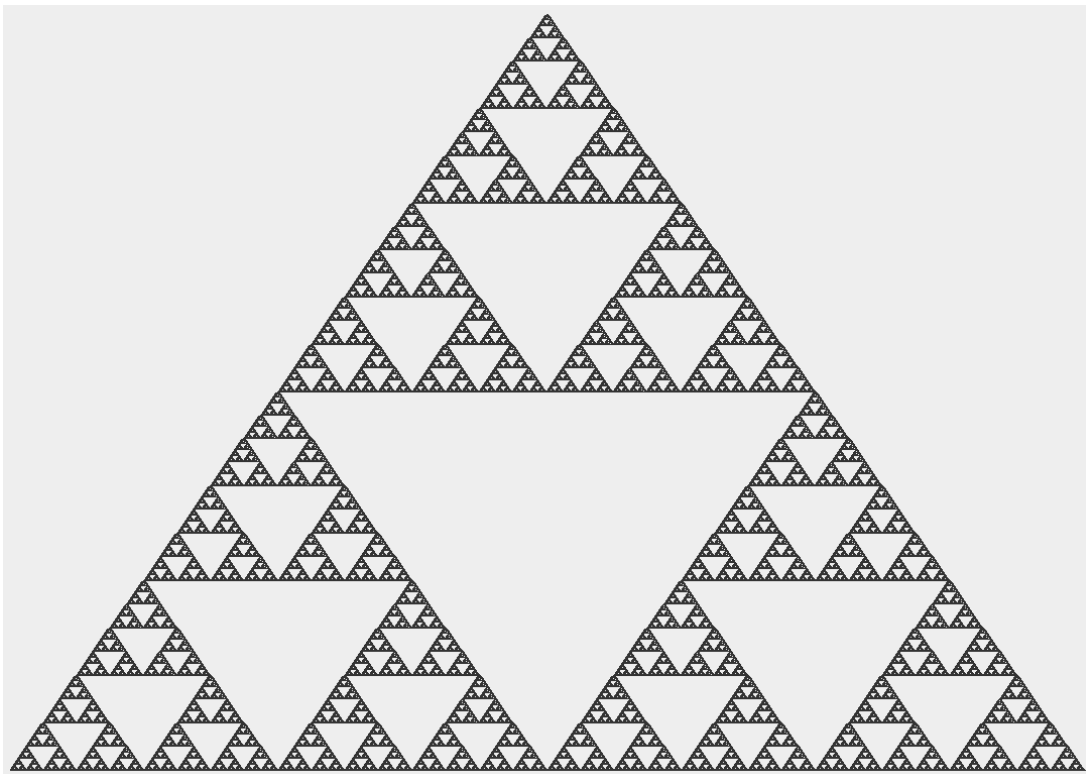
Test.java

```
1 package main7.task2;
2
3 import javax.swing.*;
4 import java.awt.*;
5
6
7 public class Test {
8
9     public static void main(String[] args) {
10         JFrame f = new SierpinskiTriangle();
11         f.setSize(300, 300);
12         f.setLocationRelativeTo(null);
13         f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
14         f.setVisible(true);
15     }
16 }
17
18 class SierpinskiTriangle extends JFrame {
19
20     public SierpinskiTriangle() {
21         JPanel panel = new JPanel();
22         add(panel, BorderLayout.SOUTH);
23         SierpinskiPanel sie = new SierpinskiPanel();
24         add(sie);
25         sie.setOrder(10);
26     }
27
28     private static class SierpinskiPanel extends JPanel {
29
```

```

30     private int order = 0;
31
32     private static void displayTriangles(Graphics g, int order,
33                                         Point p1, Point p2, Point p3) {
34         if (order == 0) {
35             g.drawLine(p1.x, p1.y, p2.x, p2.y);
36             g.drawLine(p1.x, p1.y, p3.x, p3.y);
37             g.drawLine(p2.x, p2.y, p3.x, p3.y);
38         } else {
39             Point p12 = midPoint(p1, p2);
40             Point p23 = midPoint(p2, p3);
41             Point p31 = midPoint(p3, p1);
42             displayTriangles(g, order - 1, p1, p12, p31);
43             displayTriangles(g, order - 1, p12, p2, p23);
44             displayTriangles(g, order - 1, p31, p23, p3);
45         }
46     }
47
48     private static Point midPoint(Point p1, Point p2) {
49         return new Point((p1.x + p2.x) / 2, (p1.y + p2.y) / 2);
50     }
51
52     public void setOrder(int order) {
53         this.order = order;
54         repaint();
55     }
56
57     @Override
58     protected void paintComponent(Graphics g) {
59         super.paintComponent(g);
60         Point p1 = new Point(getWidth() / 2, 10);
61         Point p2 = new Point(getWidth() - 10, getHeight() - 10);
62         Point p3 = new Point(10, getHeight() - 10);
63         displayTriangles(g, order, p1, p2, p3);
64     }
65 }
66 }

```



Вывод Я создал графическое приложение на Java.