

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

Лабораторная работа №7

По дисциплине: «Современные платформы программирования»

Выполнила:
Студентка 3 курса
Группы ПО-6
Юсковец М.А.
Проверил:
Монтик Н.С.

Брест, 2022

Цель работы: освоить возможности языка программирования Java в построении графических приложений.

Ход работы:

Вариант 25

Задание 1:

Построение графических примитивов и надписей

Требования к выполнению

- Реализовать соответствующие классы, указанные в задании;
- Организовать ввод параметров для создания объектов (можно использовать файлы);
- Осуществить визуализацию графических примитивов, решить поставленную задачу

Изобразить прямоугольник, вращающийся в плоскости фрейма вокруг одной из своих вершин.

Текст программы:

Main.java

```
import javax.swing.JFrame;
import java.awt.*;
import javax.swing.*;
import java.awt.geom.Rectangle2D;

public class Main extends JFrame {
    public Main() {
        add(new Rectangle());
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(500, 500);
        setLocationRelativeTo(null);
        getContentPane().setBackground(Color.WHITE);
        setTitle("Rectangle");
        setResizable(false);
        setVisible(true);
    }
    public static void main(String[] args) {
        new Main();
    }
}

class Rectangle extends JComponent implements Runnable {
    private static final int WIDTH = 100;
    private static final int HEIGHT = 120;
    private static final int X = 250;
    private static final int Y = 250;
    private long time = System.nanoTime();

    private Rectangle2D rectangle2D = new Rectangle2D.Double(X, Y, WIDTH,
```

```

HEIGHT);
private Double rotatePointX;
private Double rotatePointY;

public Rectangle() {
    super();
    rotatePointX = X * 1.0;
    rotatePointY = Y * 1.0;
    new Thread(this).start();
}
@Override
public void run() {
    while (true) {
        repaint();
        try {
            Thread.sleep(5);
        } catch (InterruptedException ignored) {}
    }
}
@Override
protected void paintComponent(Graphics g) {
    super.paintComponent(g);
    Graphics2D g2d = (Graphics2D) g;
    long tm = System.nanoTime() - time;
    double angle = tm / 3000000000.0;
    rectangle2D.setFrame(X, Y, WIDTH, HEIGHT);
    g2d.rotate(angle, rotatePointX, rotatePointY);
    g2d.setColor(Color.GREEN);
    g2d.fill(rectangle2D);
    g2d.draw(rectangle2D);
}
}
}

```

Rectangle.java

```

import javax.swing.JFrame;
import java.awt.*;
import javax.swing.*;
import java.awt.geom.Rectangle2D;

public class Main extends JFrame {
    public Main() {
        add(new Rectangle());
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(500, 500);
        setLocationRelativeTo(null);
        getContentPane().setBackground(Color.WHITE);
        setTitle("Rectangle");
        setResizable(false);
        setVisible(true);
    }
    public static void main(String[] args) {
        new Main();
    }
}

class Rectangle extends JComponent implements Runnable {
    private static final int WIDTH = 100;
    private static final int HEIGHT = 120;
    private static final int X = 250;
    private static final int Y = 250;
}

```

```

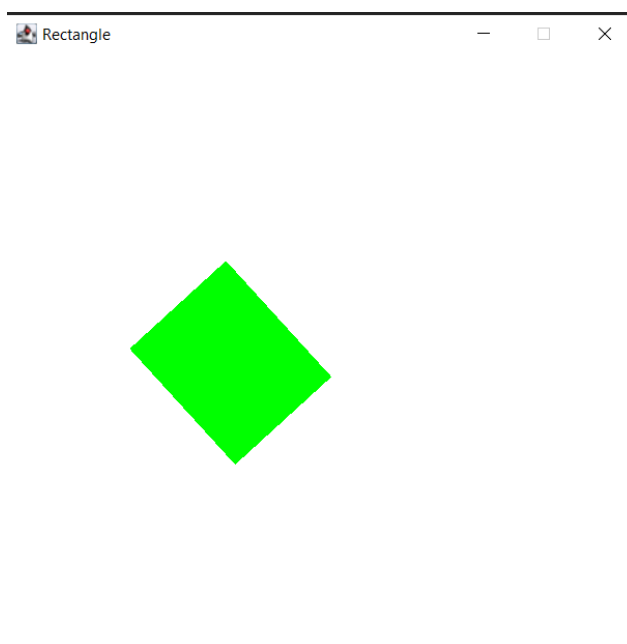
private long time = System.nanoTime();

private Rectangle2D rectangle2D = new Rectangle2D.Double(X, Y, WIDTH,
HEIGHT);
private Double rotatePointX;
private Double rotatePointY;

public Rectangle() {
    super();
    rotatePointX = X * 1.0;
    rotatePointY = Y * 1.0;
    new Thread(this).start();
}
@Override
public void run() {
    while (true) {
        repaint();
        try {
            Thread.sleep(5);
        } catch (InterruptedException ignored) {
        }
    }
}
@Override
protected void paintComponent(Graphics g) {
    super.paintComponent(g);
    Graphics2D g2d = (Graphics2D) g;
    long tm = System.nanoTime() - time;
    double angle = tm / 3000000000.0;
    rectangle2D setFrame(X, Y, WIDTH, HEIGHT);
    g2d.rotate(angle, rotatePointX, rotatePointY);
    g2d.setColor(Color.GREEN);
    g2d.fill(rectangle2D);
    g2d.draw(rectangle2D);
}
}

```

Результат программы:



Задание 2:

Реализовать построение заданного типа фрактала по варианту

Везде, где это необходимо, предусмотреть ввод параметров, влияющих на внешний вид фрактала

Множество Жюлиа.

Текст программы:

```
import java.awt.*;
import java.awt.image.BufferedImage;
import javax.swing.*;

public class Main extends JPanel {
    private final int maxIter = 1000;
    private final double zoom = 1;
    private double cY, cX;

    public Main() {
        setPreferredSize(new Dimension(800, 500));
        setBackground(Color.WHITE);
    }

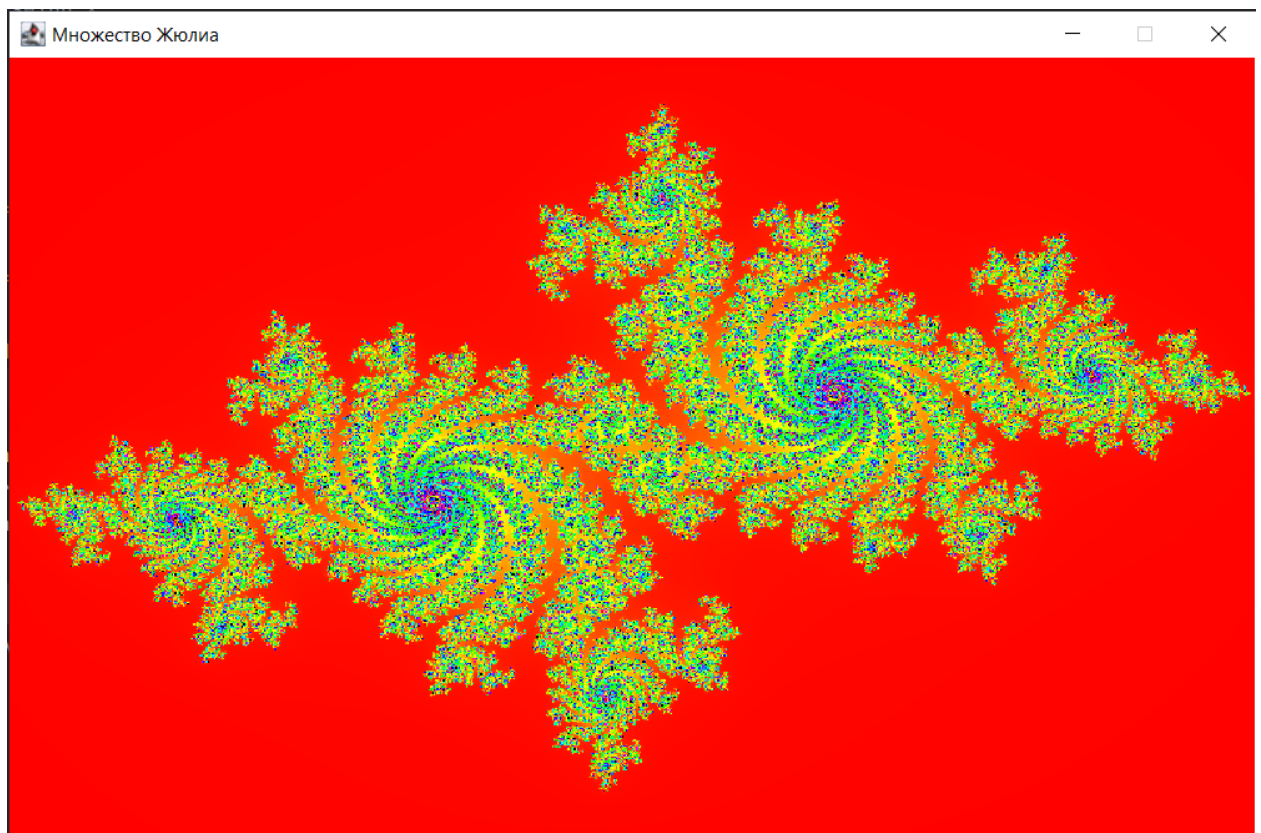
    void drawJuliaSet(Graphics2D g) {
        int w = getWidth();
        int h = getHeight();
        BufferedImage image = new BufferedImage(w, h,
            BufferedImage.TYPE_INT_RGB);

        cX = -0.7;
        cY = 0.27015;
        double moveX = 0, moveY = 0;
        double zx, zy;
        for (int x = 0; x < w; x++) {
            for (int y = 0; y < h; y++) {
                zx = 1.5 * (x - w / 2) / (0.5 * zoom * w) + moveX;
                zy = (y - h / 2) / (0.5 * zoom * h) + moveY;
                float i = maxIter;
                while (zx * zx + zy * zy < 4 && i > 0) {
                    double tmp = zx * zx - zy * zy + cX;
                    zy = 2.0 * zx * zy + cY;
                    zx = tmp;
                    i--;
                }
                int c = Color.HSBtoRGB((maxIter / i) % 1, 1, i > 0 ? 1 : 0);
                image.setRGB(x, y, c);
            }
        }
        g.drawImage(image, 0, 0, null);
    }

    @Override
    public void paintComponent(Graphics gg) {
        super.paintComponent(gg);
        Graphics2D g = (Graphics2D) gg;
        g.setRenderingHint(RenderingHints.KEY_ANTIALIASING,
            RenderingHints.VALUE_ANTIALIAS_ON);
        drawJuliaSet(g);
    }
}
```

```
public static void main(String[] args) {  
    SwingUtilities.invokeLater(() -> {  
        JFrame f = new JFrame();  
        f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
        f.setTitle("Множество Жюлиа");  
        f.setResizable(false);  
        f.add(new Main(), BorderLayout.CENTER);  
        f.pack();  
        f.setLocationRelativeTo(null);  
        f.setVisible(true);  
    });  
}
```

Результат программы:



Вывод: освоили возможности языка программирования Java в построении графических приложений.