

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
“БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ”
КАФЕДРА ИНТЕЛЛЕКТУАЛЬНЫХ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

ОТЧЁТ
по лабораторной работе №7

Выполнила:
студентка группы ПО-9
Кот А. А.

Проверил:
Крощенко А. А.

Брест 2024

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

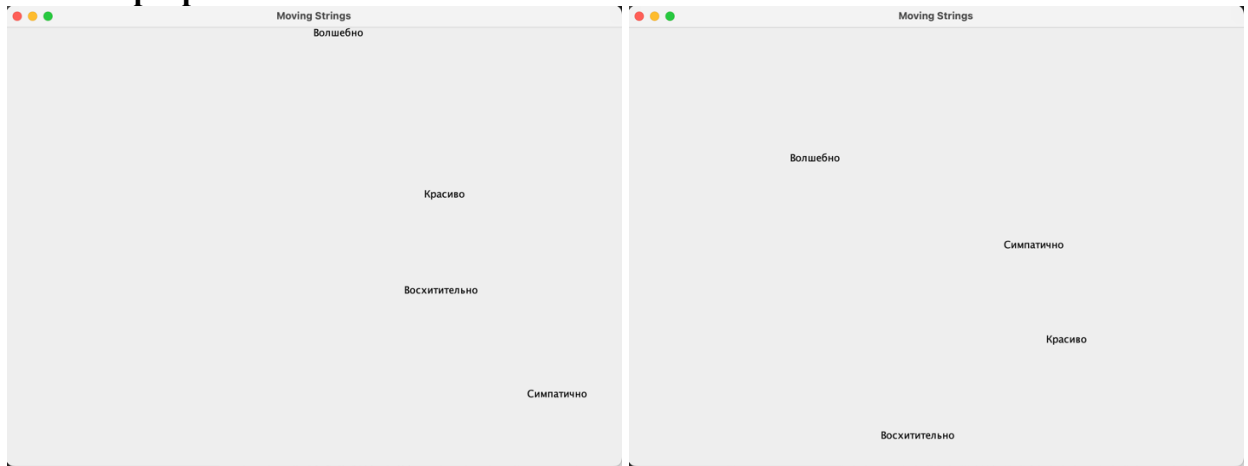
Вариант 8.

Ход работы

Задание 1.

Задать движение по экрану строк (одна за другой) из массива строк. Направление движения по апплету и значение каждой строки выбираются случайным образом.

Работа программы:



Код программы:

Task1.java

```
import javax.swing.*;
import java.awt.*;
import java.util.ArrayList;
import java.util.Random;
import java.util.Scanner;

public class Task1 extends JPanel implements Runnable {
    private String[] strings;
    private int[] xPositions;
    private int[] yPositions;
    private int[] xSpeeds;
    private int[] ySpeeds;
    private int numStrings;
    private int speed;

    public Task1(String[] strings, int speed) {
        this.strings = strings;
        this.speed = speed;
        numStrings = strings.length;
        xPositions = new int[numStrings];
        yPositions = new int[numStrings];
        xSpeeds = new int[numStrings];
        ySpeeds = new int[numStrings];
    }

    @Override
    protected void paintComponent(Graphics g) {
        super.paintComponent(g);
```

```

        for (int i = 0; i < numStrings; i++) {
            g.drawString(strings[i], xPositions[i], yPositions[i]);
        }
    }

    @Override
    public void run() {
        Random random = new Random();
        for (int i = 0; i < numStrings; i++) {
            xPositions[i] = random.nextInt(getWidth());
            yPositions[i] = random.nextInt(getHeight());
            xSpeeds[i] = random.nextInt(numStrings) - (numStrings / 2);
            ySpeeds[i] = random.nextInt(numStrings) - (numStrings / 2);
        }

        while (true) {
            moveStrings();
            repaint();
            try {
                Thread.sleep(this.speed); // Adjust the speed of movement
            } catch (InterruptedException e) {
                e.printStackTrace();
            }
        }
    }

    private void moveStrings() {
        for (int i = 0; i < numStrings; i++) {
            xPositions[i] += xSpeeds[i];
            yPositions[i] += ySpeeds[i];

            if (xPositions[i] < 0 || xPositions[i] > getWidth()) {
                xSpeeds[i] = -xSpeeds[i];
            }

            if (yPositions[i] < 0 || yPositions[i] > getHeight()) {
                ySpeeds[i] = -ySpeeds[i];
            }
        }
    }

    public static void main(String[] args) {
        ArrayList<String> strings = new ArrayList<>();

        Scanner scanner = new Scanner(System.in);

        System.out.println("Введите строки (для завершения введите пустую строку):");

        String input;
        while (!(input = scanner.nextLine()).isEmpty()) {
            strings.add(input);
        }

        System.out.println("Введите скорость движения строк: ");
        int speed = scanner.nextInt();

        JFrame frame = new JFrame("Moving Strings");
        Task1 movingStrings = new Task1(strings.toArray(new String[0]),
speed);
        frame.add(movingStrings);
        frame.setSize(800, 600); // Adjust the frame size as needed
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setVisible(true);
    }

```

```

        Thread thread = new Thread(movingStrings);
        thread.start();
    }
}

```

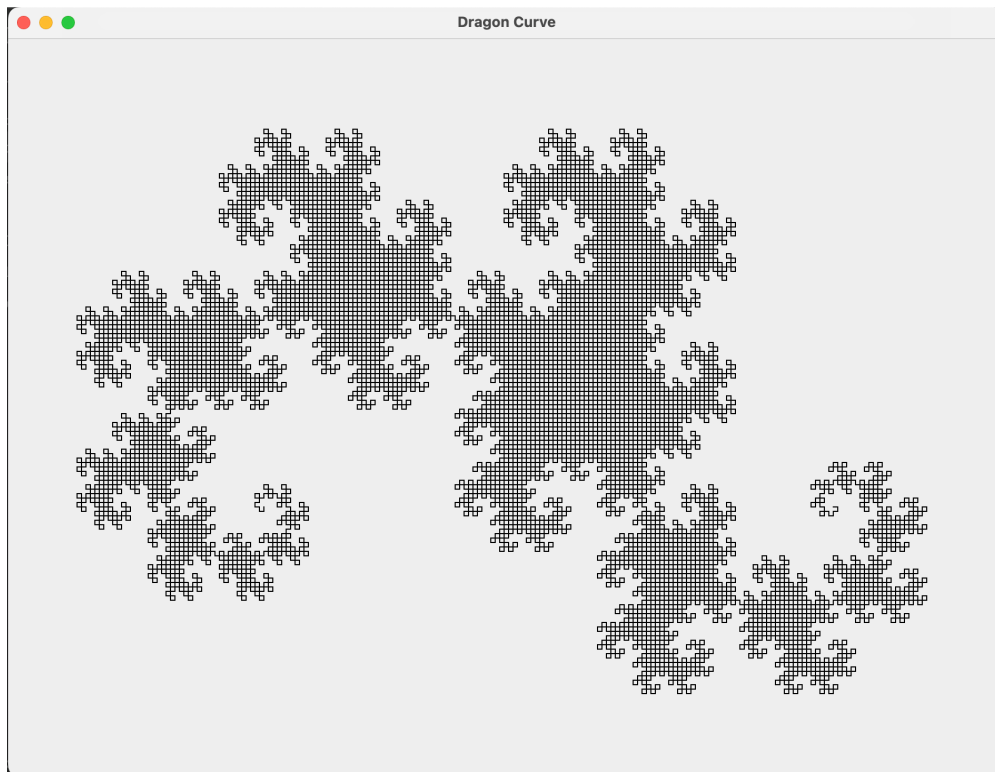
Задание 2.

Реализовать построение заданного типа фрактала по варианту: Кривая дракона.

Работа программы:

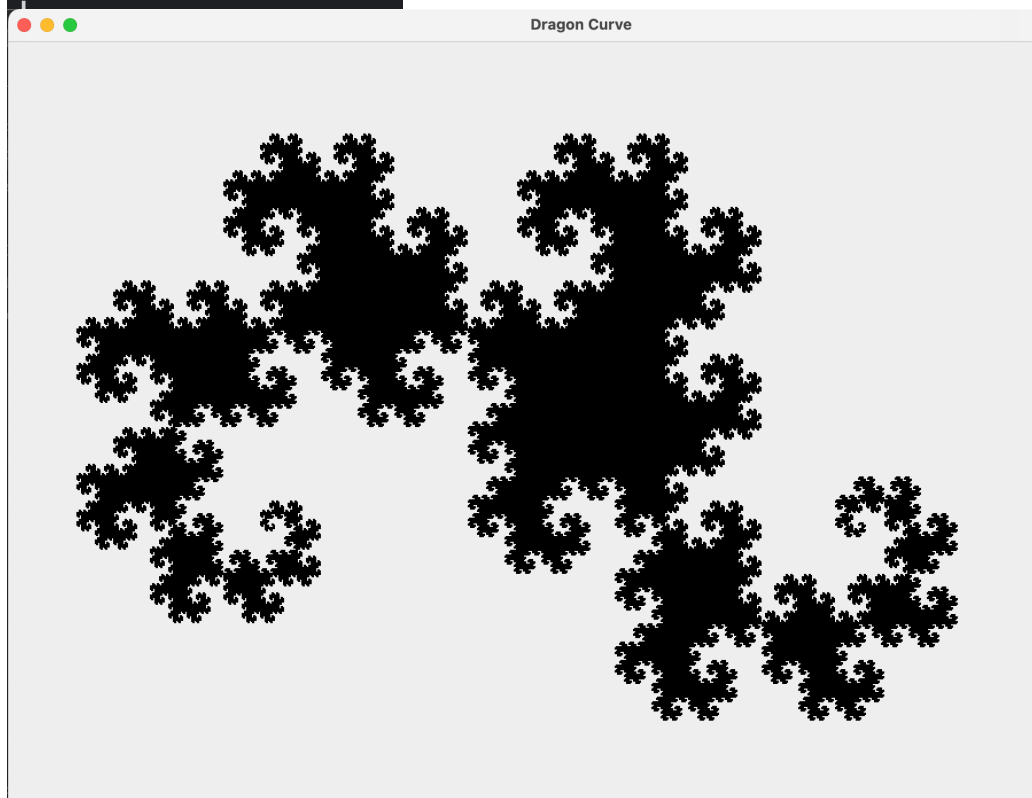
Введите количество шагов:

14



Введите количество шагов:

18



Код программы:

Task2.java

```
import javax.swing.*;
import java.awt.*;
import java.util.ArrayList;
import java.util.Collections;
import java.util.List;
import java.util.Scanner;

public class Task2 extends JFrame {
    private List<Integer> turns;
    private double startingAngle, sideLength;
    private static final double ANGLE_45DEGREES = Math.PI / 4;

    public Task2(int amountOfIterations) {
        super("Dragon Curve");
        setBounds(100, 100, 1000, 800);
        setDefaultCloseOperation(EXIT_ON_CLOSE);
        turns = getSequence(amountOfIterations);
        startingAngle = -amountOfIterations * ANGLE_45DEGREES;
        sideLength = 550 / Math.pow(2, amountOfIterations / 2.);
    }

    public List<Integer> getSequence(int iterations) {
        List<Integer> turnSequence = new ArrayList<>();
        for (int i = 0; i < iterations; i++) {
            List<Integer> copy = new ArrayList<>(turnSequence);
            Collections.reverse(copy);
            turnSequence.add(1);
            for (Integer turn : copy) {
                turnSequence.add(-turn);
            }
        }
    }
}
```

```

    }
    return turnSequence;
}

@Override
public void paint(Graphics g) {
    g.setColor(Color.BLACK);
    double angle = startingAngle;
    int x1 = 230, y1 = 450;
    int x2 = x1 + (int) (Math.cos(angle) * sideLength);
    int y2 = y1 + (int) (Math.sin(angle) * sideLength);
    g.drawLine(x1, y1, x2, y2);
    x1 = x2;
    y1 = y2;
    for (Integer turn : turns) {
        angle += turn * (Math.PI / 2);
        x2 = x1 + (int) (Math.cos(angle) * sideLength);
        y2 = y1 + (int) (Math.sin(angle) * sideLength);
        g.drawLine(x1, y1, x2, y2);
        x1 = x2;
        y1 = y2;
    }
}

public static void main(String[] args) {
    Scanner scanner = new Scanner(System.in);
    System.out.println("Введите количество шагов: ");
    int amountOfIterations = scanner.nextInt();

    new Task2(amountOfIterations).setVisible(true);
}
}

```

Вывод: в ходе лабораторной работы мы освоили возможности языка программирования Java в построении графических приложений.