

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

ОТЧЁТ

по лабораторной работе №7

Выполнил:
студент группы ПО-9
Дарашкевич Д.И.

Проверил:
Крощенко А.А.

Брест 2024

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

Вариант 4

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
- 4) Изобразить разносторонний треугольник, вращающийся в плоскости апплета вокруг своего центра тяжести.

Код программы:

```
import javafx.animation.Animation;
import javafx.animation.RotateTransition;
import javafx.application.Application;
import javafx.scene.Group;
import javafx.scene.Scene;
import javafx.scene.paint.Color;
import javafx.scene.shape.Polygon;
import javafx.stage.Stage;
import javafx.util.Duration;

public class RotatingTriangle extends Application {

    @Override
    public void start(Stage primaryStage) {
        Group root = new Group();

        double[] points = { 100, 50, 200, 200, 0, 200 };
        Polygon triangle = new Polygon(points);
        triangle.setFill(Color.BLUE);

        root.getChildren().add(triangle);

        RotateTransition rotateTransition = new RotateTransition(Duration.seconds(3),
triangle);
        rotateTransition.setByAngle(360);
        rotateTransition.setCycleCount(Animation.INDEFINITE);
        rotateTransition.setAutoReverse(false);

        rotateTransition.play();

        Scene scene = new Scene(root, 300, 250);

        primaryStage.setScene(scene);

        primaryStage.setTitle("Rotating Triangle");

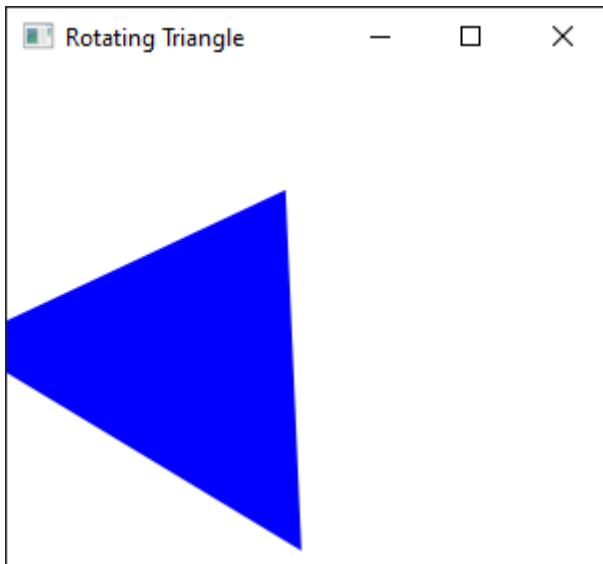
        primaryStage.show();
    }

    public static void main(String[] args) {
        launch(args);
    }
}
```

Входные данные:

```
double[] points = { 100, 50, 200, 200, 0, 200 };
    Polygon triangle = new Polygon(points);
    triangle.setFill(Color.BLUE);
```

Результат работы программы:



Задание 2:

Реализовать построение заданного типа фрактала по варианту.

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

4) Ковер Серпинского

Код программы:

```
package com.example.main;
```

```
import javafx.application.Application;
import javafx.scene.Scene;
import javafx.scene.canvas.Canvas;
import javafx.scene.canvas.GraphicsContext;
import javafx.scene.layout.StackPane;
import javafx.scene.paint.Color;
import javafx.stage.Stage;
```

```
public class SierpinskiCarpet extends Application {
```

```
    private final int width = 800;
    private final int height = 800;
    private final int depth = 5;
```

```
    @Override
```

```
    public void start(Stage primaryStage) {
        primaryStage.setTitle("Sierpinski Carpet");
        Canvas canvas = new Canvas(width, height);
        drawSierpinskiCarpet(canvas.getGraphicsContext2D(), 0, 0, width, height, depth);
        StackPane root = new StackPane();
        root.getChildren().add(canvas);
        primaryStage.setScene(new Scene(root, width, height));
        primaryStage.show();
    }
```

```
    private void drawSierpinskiCarpet(GraphicsContext gc, double x, double y, double width,
double height, int depth) {
        if (depth == 0) {
            gc.setFill(Color.BLACK);
            gc.fillRect(x, y, width, height);
        } else {
            double newWidth = width / 3;
            double newHeight = height / 3;
            for (int i = 0; i < 3; i++) {
                for (int j = 0; j < 3; j++) {
                    if (i != 1 || j != 1) {
```

```

        drawSierpinskiCarpet(gc, x + i * newWidth, y + j * newHeight,
newWidth, newHeight, depth - 1);
    }
}
}
}

public static void main(String[] args) {
    launch(args);
}
}

```

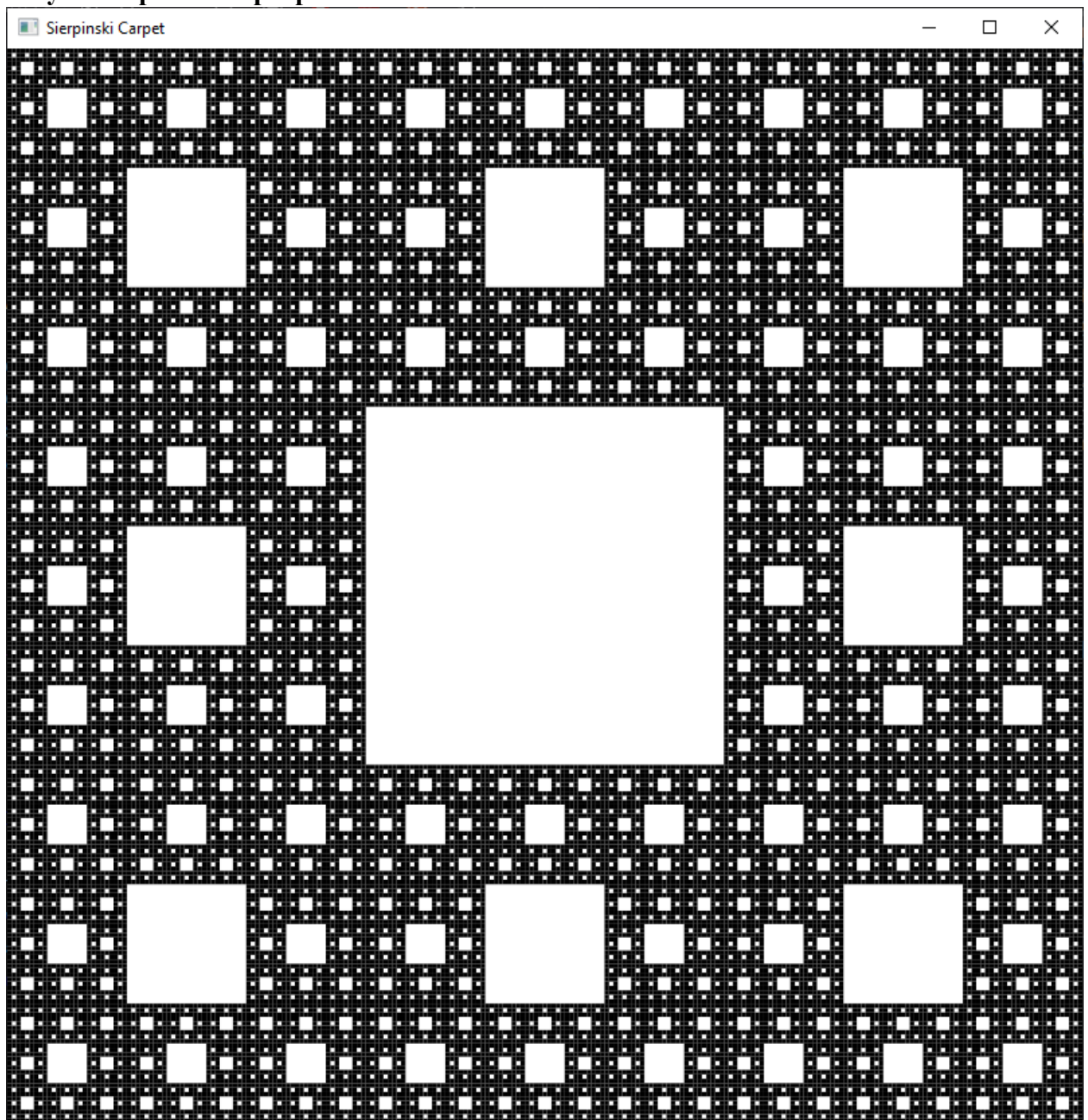
Входные данные:

```

private final int width = 800;
private final int height = 800;
private final int depth = 5;

```

Результат работы программы:



Вывод: в ходе выполнения данной лабораторной работы я освоил возможности языка программирования Java в построении графических приложений.