



МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ  
імені ІГОРЯ СІКОРСЬКОГО”

Факультет прикладної математики  
Кафедра програмного забезпечення комп’ютерних систем

**Лабораторна робота № 1**  
з дисципліни “Математичні та алгоритмічні основи комп’ютерної графіки”

Виконав  
студент III курсу  
групи КП-83  
Ландо Максим Юрійович  
варіант № 9

Зарахована  
“ \_\_\_\_ ” “ \_\_\_\_ ” 20\_\_ р.  
викладачем  
Шкурат Оксаною Сергіївною

### **Задання на лабораторну роботу**

**Завдання:** Створити малюнок за варіантом користуючись графічними примітивами бібліотеки JavaFX.

**Варіант:**



## Лістинг коду програми

### Main.java

```
package sample;

import javafx.application.Application;
import javafx.scene.paint.Color;
import javafx.scene.Group;
import javafx.scene.Scene;
import javafx.stage.Stage;

import javafx.scene.shape.*;

public class Main extends Application {
    @Override
    public void start(Stage primaryStage) {
        Group root = new Group();
        Scene scene = new Scene(root, 800, 600);
        scene.setFill(Color.rgb(255,0,0));

        Double[] big_star_points = new Double[10];
        Double[] small_star_points = new Double[10];

        double big_star_r = 200;
        double small_star_r = 75;

        double center_x = 325;
        double center_y = 300;

        for (int i=0; i<5; i++) {
            big_star_points[i*2] = center_x + big_star_r*Math.cos(Math.PI*(1.0/2.0
+ 2.0*i/5.0));
            big_star_points[i*2+1] = center_y - big_star_r*Math.sin(Math.PI*(1.0/2.
0 + 2.0*i/5.0));
            small_star_points[i*2] = center_x + small_star_r*Math.cos(Math.PI*(-
1.0/2.0 + 2.0*(i+2.0)/5.0));
            small_star_points[i*2+1] = center_y - small_star_r*Math.sin(Math.PI*(-
1.0/2.0 + 2.0*(i+2.0)/5.0));
        }

        Line line1 = new Line(small_star_points[6], small_star_points[7], small_star_points[6], small_star_points[7]+200);
        root.getChildren().add(line1);
        line1.setStroke(Color.YELLOW);
        line1.setStrokeWidth(12.0);

        Polygon pentaider = new Polygon();
        pentaider.getPoints().addAll(small_star_points);
        pentaider.setFill(Color.rgb(0,128,255));
        root.getChildren().add(pentaider);

        for (int i=0; i<5; i++) {
            Polygon trio = new Polygon(
                small_star_points[i*2], small_star_points[i*2+1],
                big_star_points[i*2], big_star_points[i*2+1],
                small_star_points[((i+1)*2) % 10], small_star_points[((i+1)*2+1
) % 10]);
            trio.setFill(Color.rgb(255,255,0));
            root.getChildren().add(trio);
        }
    }
}
```

```
        primaryStage.setScene(scene);
        primaryStage.show();
    }

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

## Результат

