

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-81  Ладуда Данило Володимирович  (*прізвище, ім’я, по батькові*)  варіант № 9 |  |  | Зарахована  “\_\_\_\_” “\_\_\_\_\_\_\_\_\_\_\_\_” 20\_\_\_ р.  викладачем  Шкурат Оксаною Сергіївною (*прізвище, ім’я, по батькові*) |

Київ 2020

**Варіант завдання**

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

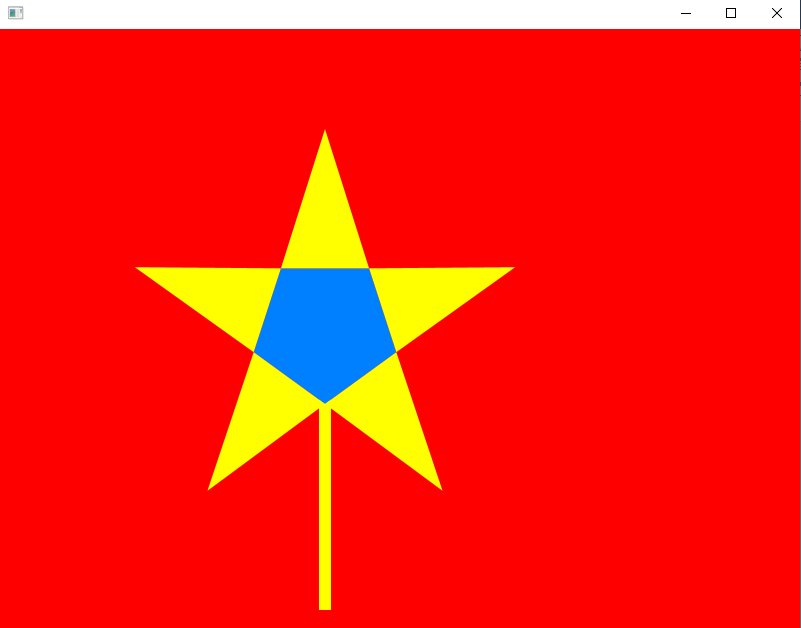
**Варіант:**



**Лістинг коду програми**

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**

****