



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 № 2**  
з дисципліни ”Математичні та алгоритмічні основи комп'ютерної графіки”

Виконав(ла)  
студент(ка) III курсу  
групи КП - 81  
Івахненко Маргарита Василівна  
*(прізвище, ім'я, по батькові)*

Київ 2021

## **Постановка завдання**

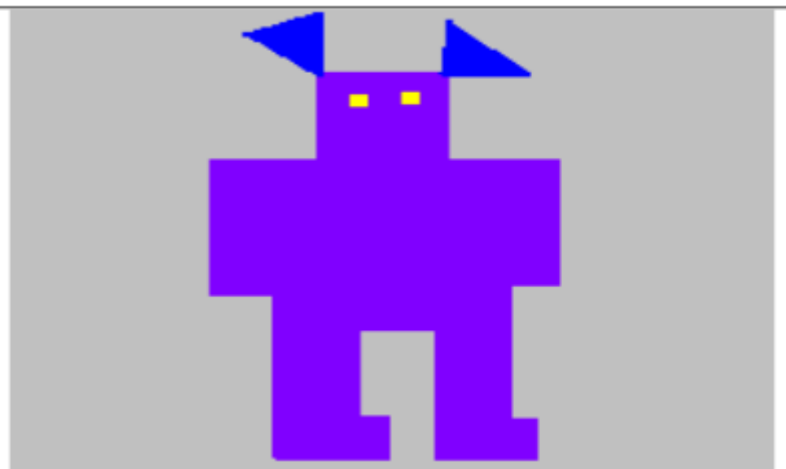
За допомогою Java 2D намалювати картинку з лабораторної роботи No1

Додатково виконати:

1. Хоча б 1 стандартний примітив, та хоча б 1 фігуру, побудовану по точкам (ламанною).
2. Хоча б 1 фігуру залити градієнтною фарбою за вибором (в цьому випадку колір може не співпадати з варіантом із лабораторної роботи No 1).
3. На достатній відстані від побудованого малюнку намалювати прямокутну рамку, всередині якої відбуватиметься анімація. Тип лінії рамки задано за варіантом.
4. Виконати анімацію малюнку, за варіантом. При цьому рамка повинна залишатися статичною. Взаємодія з рамкою не обов'язкова, якщо не передбачено варіантом.

Типи анімацій: Обертання навколо кута малюнка за годинниковою стрілкою,  
Зміна прозорості

6.



## Тексти коду програми

### Main.java

```
package com.company;

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.geom.GeneralPath;

public class Main extends JPanel implements ActionListener {
    private static int maxWidth;
    private static int maxHeight;

    private double angle = 0;
    private float alpha = 0;
    private boolean alphaGrow = true;

    Timer timer;

    Main() {
        timer = new Timer(10, this);
        timer.start();
    }

    public void paint(Graphics g) {
        Graphics2D g2d = (Graphics2D) g;

        RenderingHints rh = new
        RenderingHints(RenderingHints.KEY_ANTIALIASING,
        RenderingHints.VALUE_ANTIALIAS_ON);
        rh.put(RenderingHints.KEY_RENDERING,
        RenderingHints.VALUE_RENDER_QUALITY);
        g2d.setRenderingHints(rh);

        g2d.setBackground(Color.LIGHT_GRAY);
        g2d.clearRect(0, 0, maxWidth, maxHeight);

        Color purple = new Color(115, 38, 245);
        g2d.translate(60, 20);
        g2d.rotate(angle, 220, 270);

        AlphaComposite newComposite =
        AlphaComposite.getInstance(AlphaComposite.SRC_OVER, alpha);
        g2d.setComposite(newComposite);

        double[][] bodyPoints = {
            { 60, 0 }, { 60, 40 }, { 0, 40 }, { 0, 120 },
```

```

        { 30, 120 }, { 30, 220 }, { 80, 220 }, { 80, 190 },
        { 70, 190 }, { 70, 150 }, { 110, 150 }, { 110, 220
    },
        { 160, 220 }, { 160, 190 }, { 150, 190 }, { 150, 120
    },
        { 180, 120 }, { 180, 40 }, { 120, 40 }, { 120, 0 },
    };
    GeneralPath body = new GeneralPath();
    g2d.setColor(purple);
    g2d.translate(60, 50);

    body.moveTo(bodyPoints[0][0], bodyPoints[0][1]);
    for (int k = 1; k < bodyPoints.length; k++)
        body.lineTo(bodyPoints[k][0], bodyPoints[k][1]);
    body.closePath();
    g2d.fill(body);

    double[][] earlPoints = {
        {30, 28}, {30, 0}, {0, 8}
    };
    GeneralPath earl = new GeneralPath();
    g2d.setColor(Color.BLUE);
    g2d.translate(30, -28);

    earl.moveTo(earlPoints[0][0], earlPoints[0][1]);
    for (int k = 1; k < earlPoints.length; k++)
        earl.lineTo(earlPoints[k][0], earlPoints[k][1]);
    earl.closePath();
    g2d.fill(earl);

    double[][] ear2Points = {
        {0, 28}, {5, 0}, {30, 28}
    };
    GeneralPath ear2 = new GeneralPath();
    g2d.setColor(Color.BLUE);
    g2d.translate(90, 0);

    ear2.moveTo(ear2Points[0][0], ear2Points[0][1]);
    for (int k = 1; k < ear2Points.length; k++)
        ear2.lineTo(ear2Points[k][0], ear2Points[k][1]);
    ear2.closePath();
    g2d.fill(ear2);
    g2d.setColor(Color.YELLOW);
    g2d.translate(-180, -2);

    g2d.fillRect(135, 40, 8, 8);
    g2d.fillRect(160, 40, 8, 8);
}

public static void main( String[] args ) {
    JFrame frame = new JFrame("lab2");

```

```

        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setSize(600, 600);
        frame.setLocationRelativeTo(null);
        frame.setResizable(false);
        frame.add(new Main());

        frame.setVisible(true);
        Dimension size = frame.getSize();
        Insets insets = frame.getInsets();
        maxWidth = size.width - insets.left - insets.right - 1;
        maxHeight = size.height - insets.top - insets.bottom - 1;
    }

    public void actionPerformed(ActionEvent e) {
        angle += 0.01;
        float step = 0.01f;
        if (alphaGrow && alpha + step > 1)
            alphaGrow = false;
        if (!alphaGrow && alpha - step < 0)
            alphaGrow = true;
        if (alphaGrow)
            alpha += step;
        else
            alpha -= step;
        repaint();
    }
}

```

## Результат роботи програми

