**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”

Виконала

студентка 3 курсу

групи КП-83:

Тиченко Анна

Варіант: 20

Київ 2021

**Завдання:**

За допомогою Java 2D намалювати картинку з лабораторної роботи №1 (за варіантом). Додатково виконати:

1. Хоча б 1 стандартний примітив, та хоча б 1 фігуру, побудовану по точкам (ламаною).

2. Хоча б 1 фігуру залити градієнтною фарбою за вибором (в цьому випадку колір може не співпадати з варіантом із лабораторної роботи № 1). 11

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.

**Код програми**

|  |
| --- |
| FirTree.java |
| import java.awt.\*;  import java.awt.event.ActionEvent;  import java.awt.event.ActionListener;  import java.awt.geom.GeneralPath;  import javax.swing.JFrame;  import javax.swing.JPanel;  import javax.swing.Timer;  @SuppressWarnings("serial")  public class FirTree extends JPanel implements ActionListener {  Timer timer;  private static int maxWidth = 1600;  private static int maxHeight = 900;  private static int paddingX = 500;  private static int paddingY = 250;  private double angle = 0;  private double scale = 0.1;  private double delta = 0.01;  private final double center\_x = 1;  private final double center\_y = 1;  public FirTree() {  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(new Color(0, 127, 255));  g2d.clearRect(0, 0, maxWidth, maxHeight);  g2d.setColor(Color.RED);  BasicStroke bs1 = new BasicStroke(16, BasicStroke.CAP\_ROUND,  BasicStroke.JOIN\_MITER);  g2d.setStroke(bs1);  g2d.drawRect(10, 10, 1580, 850);  g2d.translate(maxWidth / 2, maxHeight / 2);  g2d.rotate(angle, center\_x, center\_y);  g2d.scale(scale, scale);  GradientPaint gp = new GradientPaint(5, 25, new Color(10, 115, 71), 20, 2, new Color(24, 204, 96), true);  g2d.setPaint(gp);  double points[][] = {  { -10, -250 },  { -80, -150},  { -40, -150 },  { -100, -50 },  { -60, -50 },  { -120, 50 },  { 120, 50 },  { 60, -50 },  { 100, -50 },  { 40, -150 },  { 80, -150 },  { 0, -250 }  };  GeneralPath firtree = new GeneralPath();  firtree.moveTo(points[0][0], points[0][1]);  for (int k = 1; k < points.length; k++)  firtree.lineTo(points[k][0], points[k][1]);  firtree.closePath();  g2d.fill(firtree);  g2d.setColor(new Color(128, 64, 0));  g2d.fillRect(-30, 50, 50, 120);  g2d.setColor(new Color(255, 251, 1));  g2d.fillRect(25, -120, 30, 30);  g2d.fillRect(-40, -105, 30, 30);  g2d.fillRect(20, -40, 30, 30);  }  public void actionPerformed(ActionEvent e) {  if (scale < 0.01) {  delta = -delta;  } else if (scale > 0.99) {  delta = -delta;  }  angle -= 0.1;  scale += delta;  repaint();  }  public static void main(String[] args) {  JFrame frame = new JFrame("FirTree Lab2");  frame.add(new FirTree());  frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);  frame.setSize(1600, 900);  frame.setResizable(false);  frame.setLocationRelativeTo(null);  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;  }  } |

**Результати роботи програми**

****