

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
“БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ”
КАФЕДРА ИИТ

ОТЧЁТ
по лабораторной работе №7

Выполнил:

Студент 3 курса
группы ПО-9
Кучко Ярослав Валерьевич

Проверил:

Крощенко А. А.

Брест 2024

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

Вариант 10

Задание 1. Изобразить в апплете приближающийся издали и удаляющийся шар. Шар должен двигаться с постоянной скоростью.

```
public class MovingBallApplet extends JApplet implements ActionListener {
```

```
    private static final int WIDTH = 800;
    private static final int HEIGHT = 600;
    private static final int BALL_SIZE = 50;

    private Timer timer;

    private int ballRadius;
    private int direction;

    @Override
    public void init() {
        ballRadius = BALL_SIZE;
        direction = 1;

        timer = new Timer(30, this);
        timer.start();
    }

    @Override
    public void actionPerformed(ActionEvent e) {
        ballRadius += direction;

        if (ballRadius == 200 || ballRadius == BALL_SIZE) {
            direction *= -1;
        }

        repaint();
    }

    @Override
    public void paint(Graphics g) {

        Graphics2D g2d = (Graphics2D) g;

        int centerX = getWidth() / 2;
        int centerY = getHeight() / 2;

        int x = centerX - ballRadius / 2;
        int y = centerY - ballRadius / 2;

        g2d.setColor(Color.WHITE);
        g2d.fillOval(x - 1, y - 1, ballRadius + 2, ballRadius + 2);
        g2d.setColor(Color.RED);
        g2d.fillOval(x, y, ballRadius, ballRadius);
    }

    public static void main(String[] args) {
        JFrame frame = new JFrame("Moving Ball Applet");
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        JApplet applet = new MovingBallApplet();
        applet.init();
    }
}
```

```

        frame.add(applet);
        frame.setSize(WIDTH, HEIGHT);
        frame.setVisible(true);
    }
}

```

Задание 2. Кривая Пеано.

```

public class PeanoCurve {
    public static void main(final String[] args) {
        Locale.setDefault(Locale.US);
        try (Writer writer = new BufferedWriter(new
FileWriter("peano_curve.svg"))) {
            PeanoCurve s = new PeanoCurve(writer);
            final int length = 8;
            s.currentAngle = 90;
            s.currentX = length;
            s.currentY = length;
            s.lineLength = length;
            s.begin(656);
            s.execute(rewrite(4)); // порядок
            s.end();
        } catch (final Exception ex) {
            ex.printStackTrace();
        }
    }

    private PeanoCurve(final Writer writer) {
        this.writer = writer;
    }

    private void begin(final int size) throws IOException {
        write("<svg      xmlns='http://www.w3.org/2000/svg'      width='%d'
height='%d'>\n", size, size);
        write("<rect width='100%%' height='100%%' fill='white'/>\n");
        write("<path stroke-width='1' stroke='black' fill='none' d=''");
    }

    private void end() throws IOException {
        write("'>\n</svg>\n");
    }

    private void execute(final String s) throws IOException {
        write("M%g,%g\n", currentX, currentY);
        for (int i = 0, n = s.length(); i < n; ++i) {
            switch (s.charAt(i)) {
                case 'F':
                    line(lineLength);
                    break;
                case '+':
                    turn(ANGLE);
                    break;
                case '-':
                    turn(-ANGLE);
                    break;
            }
        }
    }

    private void line(final double length) throws IOException {

```

```

        final double theta = (Math.PI * currentAngle) / 180.0;
        currentX += length * Math.cos(theta);
        currentY += length * Math.sin(theta);
        write("L%g,%g\n", currentX, currentY);
    }

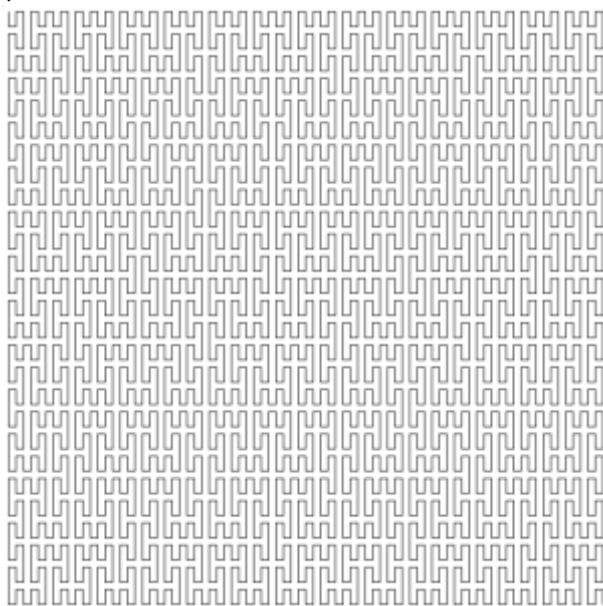
    private void turn(final int angle) {
        currentAngle = (currentAngle + angle) % 360;
    }

    private void write(final String format, final Object... args) throws
IOException {
        writer.write(String.format(format, args));
    }

    private static String rewrite(final int order) {
        String s = "L";
        for (int i = 0; i < order; ++i) {
            final StringBuilder sb = new StringBuilder();
            for (int j = 0, n = s.length(); j < n; ++j) {
                final char ch = s.charAt(j);
                if (ch == 'L')
                    sb.append("LFRFL-F-RFLFR+F+LFRFL");
                else if (ch == 'R')
                    sb.append("RFLFR+F+LFRFL-F-RFLFR");
                else
                    sb.append(ch);
            }
            s = sb.toString();
        }
        return s;
    }

    private final Writer writer;
    private double lineLength;
    private double currentX;
    private double currentY;
    private int currentAngle;
    private static final int ANGLE = 90;
}

```



Вывод: в ходе выполнения лабораторной работы освоены возможности языка программирования Java в построении графических приложений.