

Министерство образования Республики Беларусь
Учреждение образования
«Брестский государственный технический университет»
Кафедра ИИТ

Лабораторная работа №7
По дисциплине «СПП» за 5 семестр

Выполнил:

Студент группы ПО-3

Ковалёва А. И.

Проверил:

Крощенко А. А.

Вариант 12

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

Задание 1:

Задать составление строки из символов, появляющихся из разных углов апплета и выстраивающихся друг за другом. Процесс должен циклически повторяться.

Текст программы:

```
package com.company;
import java.awt.*;
import javax.swing.*;

public class Main extends JPanel {
    static int iA, jA;
    static int iN, jN;
    static int iA2, jA2;
    static int iS, jS;
    static int iT, jT;
    static int iA3, jA3;
    static int iS2, jS2;
    static int iI, jI;
    static int iA4, jA4;
    static int i0, j0;
    static JFrame frame = new JFrame("Строка");

    public static void init() {
        iA = -400; jA = -50;
        iN = -350; jN = 850;
        iA2 = 600; jA2 = 850;
        iS = 650; jS = -50;
        iT = -200; jT = -50;
        iA3 = -150; jA3 = 850;
        iS2 = 810; jS2 = 850;
        iI = 860; jI = -50;
        iA4 = 0; jA4 = -50;
        i0 = 50; j0 = 850;
    }

    public static void main(String[] args) {
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        Main panel = new Main();
        frame.add(panel);
        frame.setMinimumSize(new Dimension(700, 700));
        frame.setVisible(true);
        while (true) {
            init();
            while (jA != 400) {
                iA++;
                jA++;
                frame.repaint();
                try {
                    Thread.sleep(1);
                } catch (Exception e) {
                }
            }
        }
    }
}
```

```

        }
    }
    while (jN != 400) {
        iN++;
        jN--;
        frame.repaint();
        try {
            Thread.sleep(1);
        } catch (Exception e) {
        }
    }
    while (jA2 != 400) {
        iA2--;
        jA2--;
        frame.repaint();
        try {
            Thread.sleep(1);
        } catch (Exception e) {
        }
    }
    while (jS != 400) {
        iS--;
        jS++;
        frame.repaint();
        try {
            Thread.sleep(1);
        } catch (Exception e) {
        }
    }
    while (jT != 400) {
        iT++;
        jT++;
        frame.repaint();
        try {
            Thread.sleep(1);
        } catch (Exception e) {
        }
    }
    while (jA3 != 400) {
        iA3++;
        jA3--;
        frame.repaint();
        try {
            Thread.sleep(1);
        } catch (Exception e) {
        }
    }
    while (jS2 != 400) {
        iS2--;
        jS2--;
        frame.repaint();
        try {
            Thread.sleep(1);
        } catch (Exception e) {
        }
    }
}

```

```

        while (jI != 400) {
            iI--;
            jI++;
            frame.repaint();
            try {
                Thread.sleep(1);
            } catch (Exception e) {
            }
        }
        while (jA4 != 400) {
            iA4++;
            jA4++;
            frame.repaint();
            try {
                Thread.sleep(1);
            } catch (Exception e) {
            }
        }
        while (j0 != 400) {
            i0++;
            j0--;
            frame.repaint();
            try {
                Thread.sleep(1);
            } catch (Exception e) {
            }
        }
        try {
            Thread.sleep(100);
        } catch (Exception e) {
        }
    }
}

```

@Override

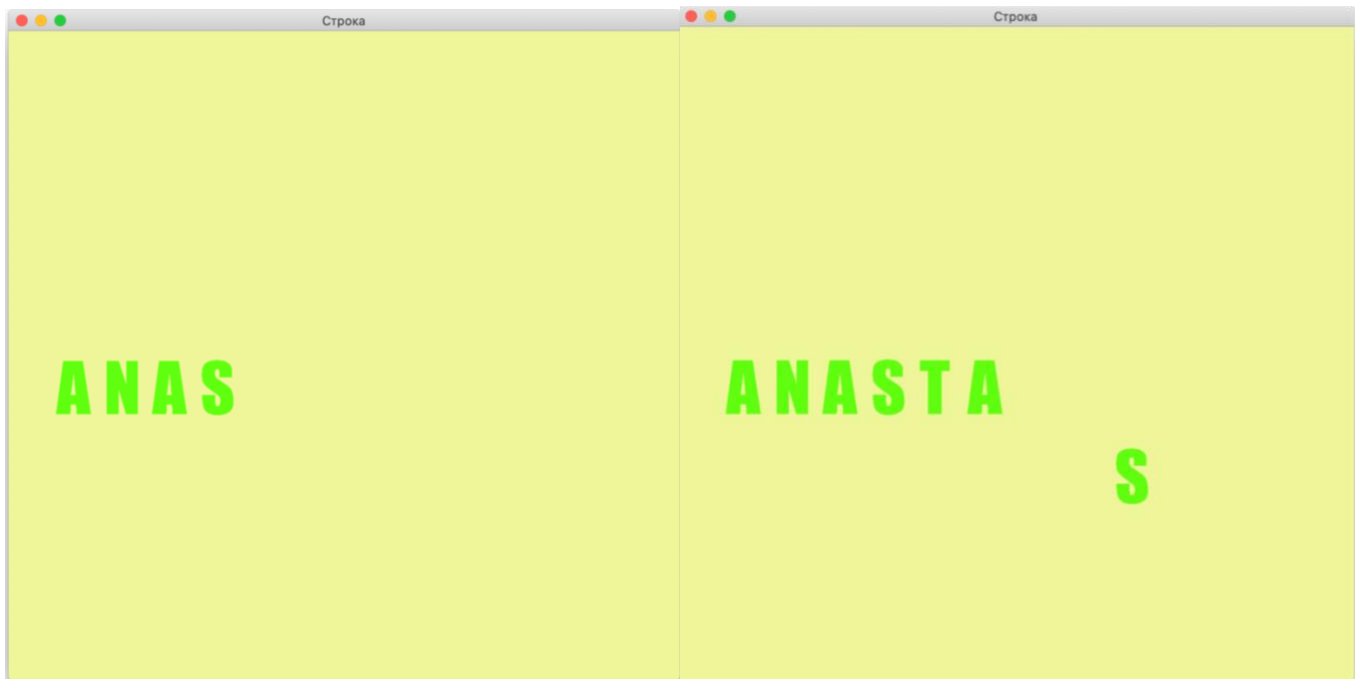
```

public void paintComponent(Graphics g) {
    g.setColor(new Color(241, 245, 164));
    g.fillRect(0, 0, 700, 700);
    g.setFont(new Font("Impact", Font.PLAIN, 70));
    g.setColor(new Color(142, 255, 81));
    g.drawString("A", iA, jA);
    g.drawString("N", iN, jN);
    g.drawString("A", iA2, jA2);
    g.drawString("S", iS, jS);
    g.drawString("T", iT, jT);
    g.drawString("A", iA3, jA3);
    g.drawString("S", iS2, jS2);
    g.drawString("I", iI, jI);
    g.drawString("A", iA4, jA4);
    g.drawString("!", i0, j0);
}

```

}

Результат выполнения:



Задание 2:

Кривая Гильберта

Текст программы:

```
package com.company;
import java.awt.*;
import java.util.ArrayList;
import java.util.Scanner;
import javax.swing.*;

public class Main extends JPanel {
    static int x = 10, y = 10, z = 600;
    static ArrayList<ArrayList<Integer>> list = new
    ArrayList<ArrayList<Integer>>();
    static JFrame frame;

    private static void addAndDrawLine(int param) {
        ArrayList sub = new ArrayList();
        if (list.size() == 0) {
            sub.add(x);
            sub.add(y);
        }
        x += (param - 1) % 2 * z;
        y += (param - 2) % 2 * z;
        if (list.size() != 0) {
            sub.add(list.get(list.size() - 1).get(2));
            sub.add(list.get(list.size() - 1).get(3));
        }
        sub.add(x);
        sub.add(y);
        list.add(sub);
        try {
            Thread.sleep(5);
        } catch (Exception e) {
```

```

        }
        frame.repaint();
    }

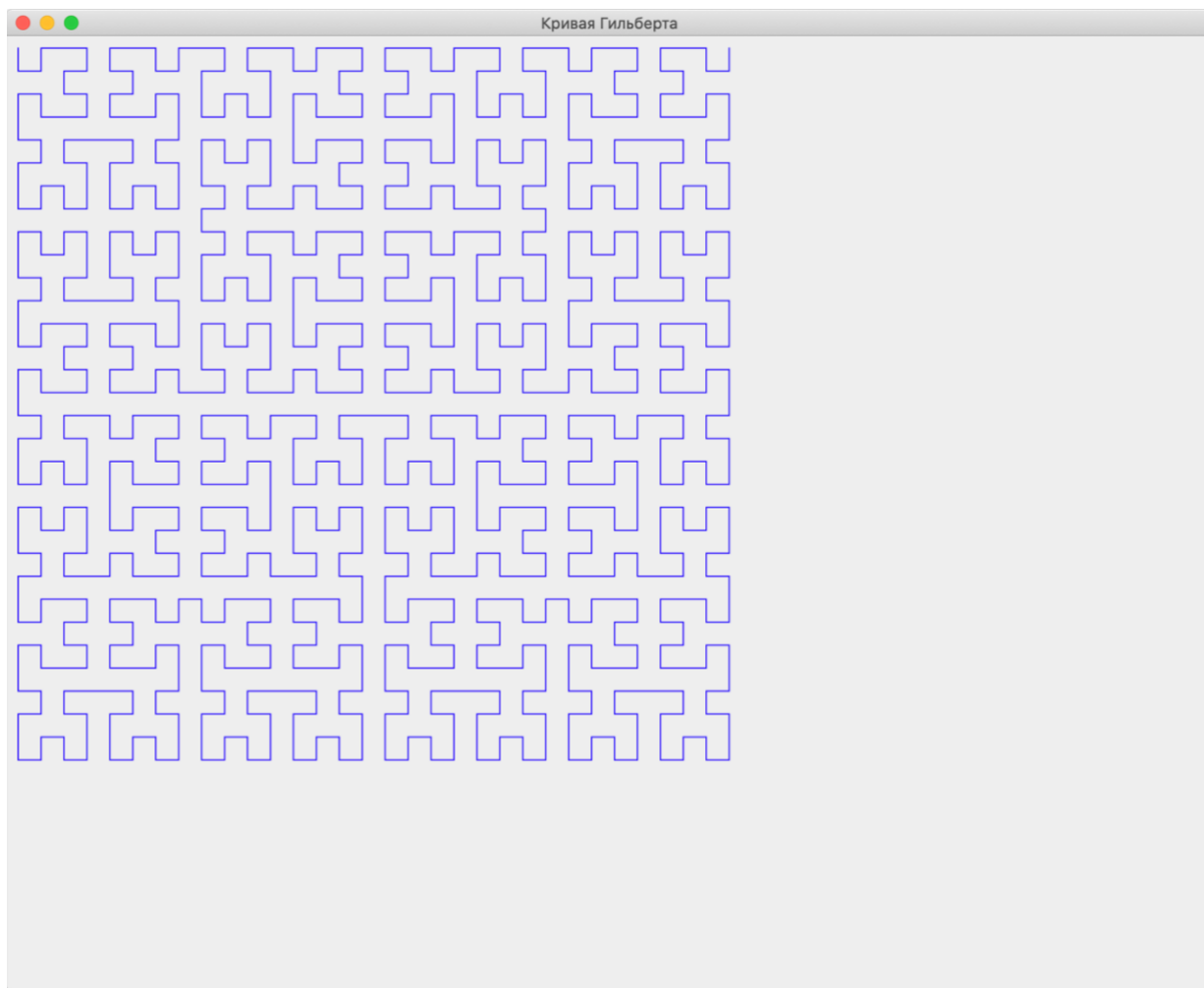
    private static void gilbert(int i, int j, int n) {
        if (n == 0) return;
        gilbert(j, i, n - 1);
        addAndDrawLine(j);
        gilbert(i, j, n - 1);
        addAndDrawLine(i);
        gilbert(i, j, n - 1);
        addAndDrawLine((j + 2) % 4);
        gilbert((j + 2) % 4, (i + 2) % 4, n - 1);
    }

    public static void main(String[] args) {
        int n;
        System.out.println("Введите порядок:");
        Scanner in = new Scanner(System.in);
        n = in.nextInt();
        frame = new JFrame("Кривая Гильберта");
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        Main panel = new Main();
        frame.add(panel);
        frame.setSize(1000, 1000);
        frame.setVisible(true);
        z /= ((1 << n) - 1);
        gilbert(2, 3, n);
    }

    @Override
    public void paintComponent(Graphics g) {
        g.setColor(new Color(0, 0, 255));
        for (int i = 0; i < list.size(); i++) {
            g.drawLine(list.get(i).get(0), list.get(i).get(1),
list.get(i).get(2), list.get(i).get(3));
        }
    }
}

```

Результат выполнения:



Вывод: В ходе выполненной работы освоила возможности языка программирования Java в построении графических приложений.