## МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

## «Вятский государственный университет»

Факультет автоматики и вычислительной техники Кафедра электронных вычислительных машин

Лабораторная работа №3 по курсу «Разработка программных систем»

Выполнил студент группы ИВТ-32 \_\_\_\_\_\_\_/Рзаев А. Э./ Проверил доцент кафедры ЭВМ \_\_\_\_\_\_\_/Чистяков  $\Gamma$ . А./

#### 1 Задание

В ходе выполнения лабораторной работы необходимо решить следующие задачи:

- Выбрать и согласовать с преподавателем задачу, для решения которой может быть использована программа, разработанная в ходе предыдущей лабораторной работы.
- Разработать программу для решения выбранной задачи (взаимодействие с пользователем должно осуществляться с применением пользовательского интерфейса).

### 2 Листинг программы

Листинг программы приведен в приложении А.

### 3 Экранные формы

Экранные формы приведены в приложении Б.

#### 4 Вывод

В ходе выполнения лабораторной работы были изучены основы технологии Swing, ее основные компоненты. Написано приложение с графическим пользовательским интерфейсом, предназначенное для создания списков случайных чисел, представляемых в форматах JSON и XML (на выбор пользователя).

## Приложение А (обязательное) Листинг программы

```
Lab3.java
package rpslabs;
import javax.swing.*;
public class Lab3 {
    public static void main(String[] args) {
        SwingUtilities.invokeLater(new PrimeNumberGenerator());
}
    PrimeNumberGenerator.java
package rpslabs;
import java.awt.Dimension;
import javax.swing.*;
public class PrimeNumberGenerator implements Runnable {
    private static final String MAIN FRAME TITLE ="Prime random number generator";
    private static final String NUMBER PANEL TITLE = "Random number";
    private static final String ARRAY PANEL TITLE = "Array of random numbers";
    private JFrame mainFrame;
    @Override
    public void run() {
        mainFrame = new JFrame(MAIN FRAME TITLE);
        mainFrame.setPreferredSize(new Dimension(640, 480));
        JPanel primeRandomNumberPanel = new PrimeRandomNumberPanel();
        JPanel primeRandomArrayPanel = new PrimeRandomArrayPanel();
        JTabbedPane mainPane = new JTabbedPane();
        mainPane.addTab(NUMBER PANEL TITLE, primeRandomNumberPanel);
        mainPane.addTab(ARRAY PANEL TITLE, primeRandomArrayPanel);
        mainFrame.getContentPane().add(mainPane);
        mainFrame.pack();
        mainFrame.setVisible(true);
    }
}
    PrimeRandomArrayPanel.java
package rpslabs;
import java.awt.Dimension;
import java.awt.event.ActionEvent;
import java.util.Arrays;
import java.util.stream.Collectors;
import javax.swing.*;
import javax.swing.border.EmptyBorder;
public class PrimeRandomArrayPanel extends JPanel {
    private static final long serialVersionUID = 1L;
    private static final String ARRAY_LENGTH_LABEL = "Length of array: ";
```

```
private static final String PRIME ARRAY BUTTON LABEL = "Get array of ranodm
   numbers";
private static final String FORMAT LABEL = "Data format: ";
private static final String JSON FORMAT = "JSON";
private static final String XML FORMAT = "XML";
private static final int MAX ARRAY LENGTH = 10000000;
private static final String INVALID_ARRAY_LENGTH = String.format(
        "Length of array must be in (0; %s]",
        MAX ARRAY LENGTH
);
private JButton primeArrayButton;
private JTextArea primeArrayTextArea;
private JTextField arrayLengthField;
private JLabel arrayLengthLabel;
private String dumpFormat = JSON FORMAT;
private PrimeRandom random = new PrimeRandom();
public PrimeRandomArrayPanel() {
    super();
    this.setLayout(new BoxLayout(this, BoxLayout.Y AXIS));
    this.setBorder(new EmptyBorder(10, 10, 10, 10));
   primeArrayButton = new JButton(PRIME ARRAY BUTTON LABEL);
   primeArrayButton.addActionListener(this::onPrimeArrayButtonClicked);
   primeArrayTextArea = new JTextArea();
   primeArrayTextArea.setEditable(false);
   arrayLengthField = new JTextField();
    arrayLengthField.setMaximumSize(new Dimension(200, 30));
    arrayLengthLabel = new JLabel(ARRAY LENGTH LABEL);
    JPanel lengthPanel = new JPanel();
    lengthPanel.setLayout(new BoxLayout(lengthPanel, BoxLayout.X AXIS));
    lengthPanel.add(arrayLengthLabel);
    lengthPanel.add(arrayLengthField);
   lengthPanel.add(primeArrayButton);
   JRadioButton selectJSON = new JRadioButton(JSON FORMAT);
    selectJSON.setActionCommand(JSON FORMAT);
    selectJSON.setSelected(true);
    selectJSON.addActionListener(e -> dumpFormat = e.getActionCommand());
   JRadioButton selectXML = new JRadioButton(XML FORMAT);
    selectXML.setActionCommand(XML FORMAT);
    selectXML.addActionListener(e -> dumpFormat = e.getActionCommand());
   ButtonGroup formatGroup = new ButtonGroup();
    formatGroup.add(selectJSON);
    formatGroup.add(selectXML);
   JPanel formatPanel = new JPanel();
    formatPanel.setLayout(new BoxLayout(formatPanel, BoxLayout.X AXIS));
    formatPanel.add(new JLabel(FORMAT LABEL));
    formatPanel.add(selectJSON);
    formatPanel.add(selectXML);
    this.add(lengthPanel);
    this.add(Box.createVerticalStrut(10));
```

```
this.add(formatPanel);
        this.add(Box.createVerticalStrut(10));
        this.add(new JScrollPane(primeArrayTextArea));
    private String dumpArrayToJSON(int[] array) {
        StringBuilder serialized = new StringBuilder();
        serialized.append("[\n");
        serialized.append(
            Arrays.stream(array)
              .mapToObj(Integer::toString)
              .map(v \rightarrow " " + v + " \setminus n")
              .collect(Collectors.joining(""))
        );
        serialized.append("]");
        return serialized.toString();
    }
    private String dumpArrayToXML(int[] array) {
        StringBuilder serialized = new StringBuilder();
        serialized.append("<?xml version=\"1.0\" ?>\n");
        serialized.append("<array>\n");
        serialized.append(
            Arrays.stream(array)
              .mapToObj (Integer::toString)
              .map(v \rightarrow " < item>" + v + "</item>\n")
              .collect(Collectors.joining(""))
        );
        serialized.append("</array>");
        return serialized.toString();
    }
    private void onPrimeArrayButtonClicked(ActionEvent event) {
        try {
            int length = Integer.parseInt(arrayLengthField.getText());
            if (length <= 0 || length > MAX ARRAY LENGTH) {
                JOptionPane.showMessageDialog(this, INVALID ARRAY LENGTH);
            } else {
                int[] array = random.getRandomArray(length);
                if (dumpFormat.equals(JSON FORMAT)) {
                    primeArrayTextArea.setText(dumpArrayToJSON(array));
                 } else {
                    primeArrayTextArea.setText(dumpArrayToXML(array));
            }
        } catch (NumberFormatException ex) {
            JOptionPane.showMessageDialog(this, INVALID ARRAY LENGTH);
        }
    }
     PrimeRandomNumberPanel.java
package rpslabs;
import java.awt.Dimension;
import java.awt.event.ActionEvent;
```

}

```
import javax.swing.*;
import javax.swing.border.EmptyBorder;
public class PrimeRandomNumberPanel extends JPanel {
    private static final long serialVersionUID = 1L;
    private static final String PRIME_NUMBER_BUTTON_LABEL = "Get prime random
       number";
    private PrimeRandom random = new PrimeRandom();
    private JTextField primeNumberField;
    private JButton primeNumberButton;
    public PrimeRandomNumberPanel() {
        super();
        this.setLayout(new BoxLayout(this, BoxLayout.X AXIS));
        this.setBorder(new EmptyBorder(10, 10, 10, 10));
        primeNumberField = new JTextField();
        primeNumberField.setMaximumSize(new Dimension(200, 30));
        primeNumberField.setEditable(false);
        primeNumberButton = new JButton(PRIME NUMBER BUTTON LABEL);
        primeNumberButton.addActionListener(this::onPrimeNumberButtonClicked);
        this.add(primeNumberButton);
        this.add(Box.createHorizontalStrut(10));
        this.add(primeNumberField);
    }
    private void onPrimeNumberButtonClicked(ActionEvent event) {
        primeNumberField.setText(Integer.toString(random.getRandomNumber()));
}
```

# Приложение Б (обязательное) Экранные формы

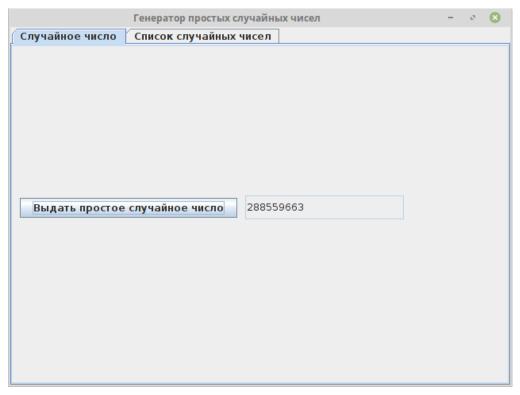


Рисунок 1 – Вкладка для генерации единственного случайного числа.

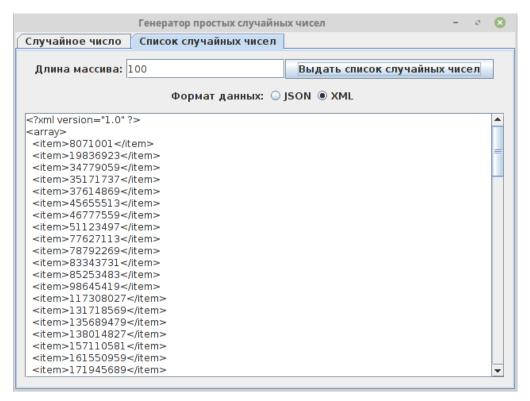


Рисунок 2 – Вкладка для генерации списка случайных чисел.