Министерство науки и высшего образования РФ

ФГАОУ ВО Пермский национальный исследовательский

политехнический университет

Кафедра «Вычислительная математика, механика и биомеханика»

Отчет по лабораторной работе № 9

тема «Визуализация данных»

по дисциплине «Информатика»

Выполнил: студент группы ИСТ-22-1б Золотарев И.О.

Пермь, 2023

**Содержание**

[Задание 1 3](file:///C:\Users\ivanx\Downloads\Telegram%20Desktop\лабагуи.docx#_Toc134195275)

[1.1. Постановка задачи 3](file:///C:\Users\ivanx\Downloads\Telegram%20Desktop\лабагуи.docx#_Toc134195276)

[1.2. Решение задачи, код программы 4](file:///C:\Users\ivanx\Downloads\Telegram%20Desktop\лабагуи.docx#_Toc134195277)

[1.3. Тестирование работы программы 5](file:///C:\Users\ivanx\Downloads\Telegram%20Desktop\лабагуи.docx#_Toc134195278)

[Задание 2 6](file:///C:\Users\ivanx\Downloads\Telegram%20Desktop\лабагуи.docx#_Toc134195279)

[2.1. Постановка задачи 6](file:///C:\Users\ivanx\Downloads\Telegram%20Desktop\лабагуи.docx#_Toc134195280)

[2.2. Решение задачи, код программы 7](file:///C:\Users\ivanx\Downloads\Telegram%20Desktop\лабагуи.docx#_Toc134195281)

[2.3. Тестирование работы программы 9](file:///C:\Users\ivanx\Downloads\Telegram%20Desktop\лабагуи.docx#_Toc134195282)

[Задание 3 10](file:///C:\Users\ivanx\Downloads\Telegram%20Desktop\лабагуи.docx#_Toc134195283)

[3.1. Постановка задача 10](file:///C:\Users\ivanx\Downloads\Telegram%20Desktop\лабагуи.docx#_Toc134195284)

[3.2. Решение задачи, код программы 11](file:///C:\Users\ivanx\Downloads\Telegram%20Desktop\лабагуи.docx#_Toc134195285)

[3.3. Тестирование работы программы 13](file:///C:\Users\ivanx\Downloads\Telegram%20Desktop\лабагуи.docx#_Toc134195286)

[Задание 4 14](file:///C:\Users\ivanx\Downloads\Telegram%20Desktop\лабагуи.docx#_Toc134195287)

[4.1. Постановка задачи 14](file:///C:\Users\ivanx\Downloads\Telegram%20Desktop\лабагуи.docx#_Toc134195288)

[4.2. Решение задачи, код программы 15](file:///C:\Users\ivanx\Downloads\Telegram%20Desktop\лабагуи.docx#_Toc134195289)

[4.3. Тестирование работы программы 17](file:///C:\Users\ivanx\Downloads\Telegram%20Desktop\лабагуи.docx#_Toc134195290)

[Задание 5 18](file:///C:\Users\ivanx\Downloads\Telegram%20Desktop\лабагуи.docx#_Toc134195291)

[5.1. Постановка задачи 18](file:///C:\Users\ivanx\Downloads\Telegram%20Desktop\лабагуи.docx#_Toc134195292)

[5.2. Решение задачи, код программы 19](file:///C:\Users\ivanx\Downloads\Telegram%20Desktop\лабагуи.docx#_Toc134195293)

[5.3. Тестирование работы программы 21](file:///C:\Users\ivanx\Downloads\Telegram%20Desktop\лабагуи.docx#_Toc134195294)

# Задание 1

## 

## Постановка задачи

Создать окно с двумя полями ввода и кнопкой. У первого поля установить ограничение по количеству вводимых символов 25 символов. Второе поле использовать для указания пути к файлу. При нажатии на кнопку должна происходить запись данных из поля ввода в файл путь, к которому указывается во втором поле ввода.

## 1.2. Решение задачи, код программы

import javax.swing.\*;  
import javax.swing.text.AttributeSet;  
import javax.swing.text.BadLocationException;  
import javax.swing.text.Document;  
import javax.swing.text.PlainDocument;  
import java.awt.\*;  
import java.awt.event.ActionEvent;  
import java.awt.event.ActionListener;  
import java.io.BufferedWriter;  
import java.io.FileWriter;  
import java.io.IOException;  
  
public class Main {  
 public static void main(String[] args) {  
 SwingUtilities.*invokeLater*(Main::*createAndShowGUI*);  
 }  
  
 private static void createAndShowGUI() {  
 JFrame frame = new JFrame("Input to File");  
 frame.setDefaultCloseOperation(JFrame.*EXIT\_ON\_CLOSE*);  
 frame.setSize(400, 150);  
 JPanel panel = new JPanel(new GridLayout(3, 2));  
 frame.add(panel);  
 JLabel inputLabel = new JLabel("Input (max 25 characters):");  
 panel.add(inputLabel);  
 JTextField inputField = new JTextField();  
 inputField.setDocument((Document) new JTextFieldLimit(25));  
 panel.add(inputField);  
 JLabel pathLabel = new JLabel("File path:");  
 panel.add(pathLabel);  
 JTextField pathField = new JTextField();  
 panel.add(pathField);  
 JButton saveButton = new JButton("Save to File");  
 saveButton.addActionListener(new ActionListener() {  
 @Override  
 public void actionPerformed(ActionEvent e) {  
 String input = inputField.getText();  
 String filePath = pathField.getText();  
 *saveToFile*(input, filePath);  
 }  
 });  
 panel.add(saveButton);  
  
 frame.setVisible(true);  
 }  
  
 private static void saveToFile(String input, String filePath) {  
 try (BufferedWriter writer = new BufferedWriter(new FileWriter(filePath))) {  
 writer.write(input);  
 JOptionPane.*showMessageDialog*(null, "Data saved to file successfully.");  
 } catch (IOException e) {  
 JOptionPane.*showMessageDialog*(null, "Error saving data to file: " + e.getMessage(), "Error", JOptionPane.*ERROR\_MESSAGE*);  
 }  
 }  
  
 static class JTextFieldLimit extends PlainDocument {  
 private int limit;  
  
 JTextFieldLimit(int limit) {  
 super();  
 this.limit = limit;  
 }  
  
 public void insertString(int offset, String str, AttributeSet attr) throws BadLocationException {  
 if (str == null) {  
 return;  
 }  
 if ((getLength() + str.length()) <= limit) {  
 super.insertString(offset, str, attr);  
 }  
 }  
 }  
}

## Тестирование работы программы

|  |
| --- |
| Количество символов меньше 25 (включительно) |
|  |
|  |
|  |

# Задание 2

## 

## 2.1. Постановка задачи

Создать программу с графическим интерфейсом, которая предлагает два варианта использования

1. Отображает поля ввода имени, фамилии, отчества, даты рождения, учебной группы. При нажатии на кнопку «Записать» сохраняет данные в файл
2. При нажатии кнопки «Загрузить» извлекает из файла данные и отображает в полях ввода.

## 2.2. Решение задачи, код программы

import javax.swing.\*;  
import java.awt.\*;  
import java.awt.event.ActionEvent;  
import java.awt.event.ActionListener;  
import java.io.BufferedReader;  
import java.io.BufferedWriter;  
import java.io.FileReader;  
import java.io.FileWriter;  
import java.io.IOException;  
  
public class Main {  
 public static void main(String[] args) {  
 SwingUtilities.*invokeLater*(() -> *createAndShowGUI*());  
 }  
  
 private static void createAndShowGUI() {  
 JFrame frame = new JFrame("Student Data");  
 frame.setDefaultCloseOperation(JFrame.*EXIT\_ON\_CLOSE*);  
 frame.setSize(500, 250);  
  
 JPanel panel = new JPanel(new GridLayout(6, 2));  
 frame.add(panel);  
  
 JLabel nameLabel = new JLabel("Name:");  
 panel.add(nameLabel);  
  
 JTextField nameField = new JTextField();  
 panel.add(nameField);  
  
 JLabel surnameLabel = new JLabel("Surname:");  
 panel.add(surnameLabel);  
  
 JTextField surnameField = new JTextField();  
 panel.add(surnameField);  
  
 JLabel patronymicLabel = new JLabel("Patronymic:");  
 panel.add(patronymicLabel);  
  
 JTextField patronymicField = new JTextField();  
 panel.add(patronymicField);  
  
 JLabel birthDateLabel = new JLabel("Birth Date:");  
 panel.add(birthDateLabel);  
  
 JTextField birthDateField = new JTextField();  
 panel.add(birthDateField);  
  
 JLabel groupLabel = new JLabel("Group:");  
 panel.add(groupLabel);  
  
 JTextField groupField = new JTextField();  
 panel.add(groupField);  
  
 JButton saveButton = new JButton("Save");  
 saveButton.addActionListener(new ActionListener() {  
 @Override  
 public void actionPerformed(ActionEvent e) {  
 String[] data = {  
 nameField.getText(),  
 surnameField.getText(),  
 patronymicField.getText(),  
 birthDateField.getText(),  
 groupField.getText()  
 };  
 *saveToFile*(data, "student\_data.txt");  
 }  
 });  
 panel.add(saveButton);  
  
 JButton loadButton = new JButton("Load");  
 loadButton.addActionListener(new ActionListener() {  
 @Override  
 public void actionPerformed(ActionEvent e) {  
 String[] data = *loadFromFile*("student\_data.txt");  
 if (data != null) {  
 nameField.setText(data[0]);  
 surnameField.setText(data[1]);  
 patronymicField.setText(data[2]);  
 birthDateField.setText(data[3]);  
 groupField.setText(data[4]);  
 }  
 }  
 });  
 panel.add(loadButton);  
  
 frame.setVisible(true);  
 }  
  
 private static void saveToFile(String[] data, String filePath) {  
 try (BufferedWriter writer = new BufferedWriter(new FileWriter(filePath))) {  
 for (String line : data) {  
 writer.write(line);  
 writer.newLine();  
 }  
 JOptionPane.*showMessageDialog*(null, "Data saved to file successfully.");  
 } catch (IOException e) {  
 JOptionPane.*showMessageDialog*(null, "Error saving data to file: " + e.getMessage(), "Error", JOptionPane.*ERROR\_MESSAGE*);  
 }  
 }  
  
 private static String[] loadFromFile(String filePath) {  
 String[] data = new String[5];  
 try (BufferedReader reader = new BufferedReader(new FileReader(filePath))) {  
 for (int i = 0; i < data.length; i++) {  
 data[i] = reader.readLine();  
 }  
 return data;  
 } catch (IOException e) {  
 JOptionPane.*showMessageDialog*(null, "Error loading data from file: " + e.getMessage(), "Error", JOptionPane.*ERROR\_MESSAGE*);  
 return null;  
 }  
 }  
}

## 2.3. Тестирование работы программы

|  |  |
| --- | --- |
| Записать | Загрузить |
|  |  |
|  |  |

# Задание 3

## 

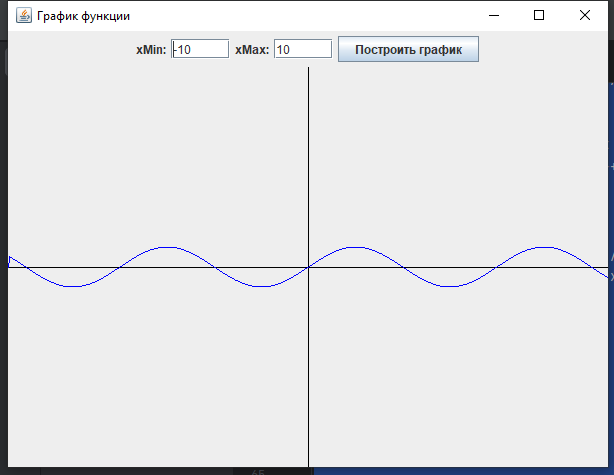
## 3.1. Постановка задача

Создать программу, которая запрашивает у пользователя интервал и строит на нем график функции из задания №10 лабораторной работы №5 вашего варианта.

## 3.2. Решение задачи, код программы

import javax.swing.\*;  
import java.awt.\*;  
import java.awt.event.ActionEvent;  
import java.awt.event.ActionListener;  
public class Main extends JFrame implements ActionListener {  
 private JTextField xMinField, xMaxField;  
 private JButton plotButton;  
 private JPanel graphPanel;  
 public Main() {  
 super("График функции");  
 xMinField = new JTextField("-10", 5);  
 xMaxField = new JTextField("10", 5);  
 plotButton = new JButton("Построить график");  
 plotButton.addActionListener(this);  
 JPanel inputPanel = new JPanel();  
 inputPanel.add(new JLabel("xMin:"));  
 inputPanel.add(xMinField);  
 inputPanel.add(new JLabel("xMax:"));  
 inputPanel.add(xMaxField);  
 inputPanel.add(plotButton);  
 graphPanel = new JPanel() {  
 public void paintComponent(Graphics g) {  
 super.paintComponent(g);  
 drawGraph(g);  
 }  
 };  
 graphPanel.setPreferredSize(new Dimension(600, 400));  
 getContentPane().add(inputPanel, BorderLayout.*NORTH*);  
 getContentPane().add(graphPanel, BorderLayout.*CENTER*);  
 setDefaultCloseOperation(JFrame.*EXIT\_ON\_CLOSE*);  
 pack();  
 setVisible(true);  
 }  
 public void actionPerformed(ActionEvent e) {  
 if (e.getSource() == plotButton) {  
 graphPanel.repaint();  
 }  
 }  
 private void drawGraph(Graphics g) {  
 double xMin = Double.*parseDouble*(xMinField.getText());  
 double xMax = Double.*parseDouble*(xMaxField.getText());  
 int width = graphPanel.getWidth();  
 int height = graphPanel.getHeight();  
 g.setColor(Color.*BLACK*);  
 g.drawLine(0, height / 2, width, height / 2);  
 g.drawLine(width / 2, 0, width / 2, height);  
 g.setColor(Color.*BLUE*);  
 double xStep = (xMax - xMin) / width;  
 double x = xMin;  
 double y = f(x);  
 int lastX = 0, lastY = height / 2;  
 for (int i = 1; i < width; i++) {  
 x += xStep;  
 y = f(x);  
 int xPos = i;  
 int yPos = (int) (height / 2 - y \* height / (xMax - xMin));  
 g.drawLine(lastX, lastY, xPos, yPos);  
 lastX = xPos;  
 lastY = yPos;  
 }  
 }  
 private double f(double x) {  
 return Math.*sin*(x);  
 }  
  
 public static void main(String[] args) {  
 new Main();  
 }  
}

## 3.3. Тестирование работы программы



# Задание 4

## 

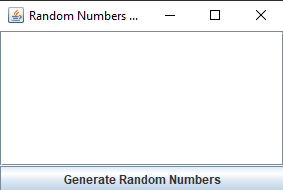
## 4.1. Постановка задачи

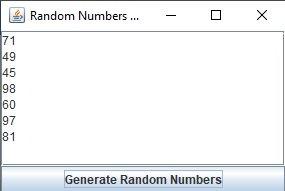
Создать программу, которая при нажатии кнопки вычисляет 7 случайных чисел и отображает их одно под другим в текстовом поле ввода. Поле ввода должно быть неизменяемо для пользователя!

## 4.2. Решение задачи, код программы

import javax.swing.\*;  
import java.awt.\*;  
import java.awt.event.ActionEvent;  
import java.awt.event.ActionListener;  
import java.util.Random;  
  
public class Main {  
 public static void main(String[] args) {  
 SwingUtilities.*invokeLater*(() -> {  
 JFrame frame = new JFrame("Random Numbers Generator");  
 frame.setDefaultCloseOperation(JFrame.*EXIT\_ON\_CLOSE*);  
 frame.setSize(300, 200);  
 JPanel panel = new JPanel();  
 panel.setLayout(new BorderLayout());  
 JTextArea textArea = new JTextArea();  
 textArea.setEditable(false);  
 JScrollPane scrollPane = new JScrollPane(textArea);  
 panel.add(scrollPane, BorderLayout.*CENTER*);  
 JButton button = new JButton("Generate Random Numbers");  
 button.addActionListener(new ActionListener() {  
 @Override  
 public void actionPerformed(ActionEvent e) {  
 Random random = new Random();  
 StringBuilder numbers = new StringBuilder();  
 for (int i = 0; i < 7; i++) {  
 numbers.append(random.nextInt(100)).append("\n");  
 }  
 textArea.setText(numbers.toString());  
 }  
 });  
 panel.add(button, BorderLayout.*SOUTH*);  
 frame.add(panel);  
 frame.setVisible(true);  
 });  
 }  
}

## 4.3. Тестирование работы программы



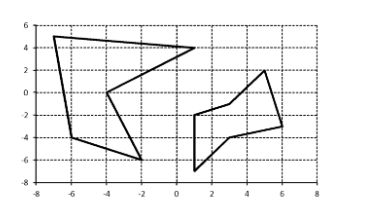


# Задание 5

## 

## 5.1. Постановка задачи

Построить график фигуры попадания точки в область из задания №5 лабораторной работы №4 Вашего варианта.



## 5.2. Решение задачи, код программы

import javax.swing.\*;  
import java.awt.\*;  
  
public class Main extends JFrame {  
 public Main(String s){  
 super(s);  
 setSize(900, 800);  
 setVisible(true);  
 setResizable(false);  
 setDefaultCloseOperation(JFrame.*EXIT\_ON\_CLOSE*);  
 }  
 public void paint(Graphics g) {  
  
 Graphics2D g2d = (Graphics2D) g;  
 g.setColor(Color.*BLACK*);  
  
 g2d.setStroke(new BasicStroke(2));  
 g.drawLine(50, 150, 50, 555);//ось y  
 g.drawLine(45, 550, 845, 550);//ось x  
 g2d.setStroke(new BasicStroke(4));  
  
 //phigure 1  
 g.drawLine(345,500,147,451);  
 g.drawLine(147,451,100,225);  
 g.drawLine(100,225,500,253);  
 g.drawLine(500,253,245,350);  
 g.drawLine(245,350,345,500);  
  
 //phigure2  
 g.drawLine(500,403,500,525);  
 g.drawLine(500,525,590,450);  
 g.drawLine(590,450,747,425);  
 g.drawLine(747,425,690,300);  
 g.drawLine(690,300,590,375);  
 g.drawLine(590,375,500,403);  
  
 g2d.setStroke(new BasicStroke(2, BasicStroke.*CAP\_BUTT*, BasicStroke.*JOIN\_MITER*, 1, new float[]{5f, 5f}, 0f));//Штрих  
  
 g.drawLine(150, 150, 150, 555);/////// x  
 g.drawLine(250, 150, 250, 555);  
 g.drawLine(350, 150, 350, 555);  
 g.drawLine(450, 150, 450, 555);  
 g.drawLine(550, 150, 550, 555);  
 g.drawLine(650, 150, 650, 555);  
 g.drawLine(750, 150, 750, 555);  
 g.drawLine(845, 150, 845, 555);  
  
  
 g.setFont(new Font("Serif", Font.*ROMAN\_BASELINE*, 14));  
 g.drawString("-8", 40, 573); // Подписи X  
 g.drawString("-6", 140, 573);  
 g.drawString("-4", 240, 573);  
 g.drawString("-2", 340, 573);  
 g.drawString("0", 440, 573);  
 g.drawString("2", 540, 573);  
 g.drawString("4", 640, 573);  
 g.drawString("6", 740, 573);  
 g.drawString("8", 840, 573);  
  
  
 g.drawLine(45, 150, 845, 150);///y  
 g.drawLine(45, 200, 845, 200);  
 g.drawLine(45, 250, 845, 250);  
 g.drawLine(45, 300, 845, 300);  
 g.drawLine(45, 350, 845, 350);  
 g.drawLine(45, 400, 845, 400);  
 g.drawLine(45, 450, 845, 450);  
 g.drawLine(45, 500, 845, 500);  
 g.drawLine(45, 550, 845, 550);  
  
  
 g.drawString("-8", 27, 553); //Подписи Y  
 g.drawString("-6", 27, 503);  
 g.drawString("-4", 27, 453);  
 g.drawString("-2", 27, 403);  
 g.drawString("0", 27, 353);  
 g.drawString("2", 27, 303);  
 g.drawString("4", 27, 253);  
 g.drawString("6", 27, 203);  
 g.drawString("8", 27, 153);  
  
 }  
 public static void main(String args[]){  
 new Main("Фигуры");  
 }  
}

## 5.3. Тестирование работы программы

