Министерство науки и высшего образования РФ

ФГАОУ ВО Пермский национальный исследовательский

политехнический университет

Кафедра «Вычислительная математика, механика и биомеханика»

Отчет по лабораторной работе № 11

тема «GUI»

по дисциплине «Информатика»

Выполнил: студент группы ИСТ-22-1б Васин М.А.

Проверил: ассистент каф. ВММБ Нетбай Г.В.

Пермь, 2021

**Содержание**

[Задание 1 3](#_Toc136812927)

[1.1. Постановка задачи 3](#_Toc136812928)

[1.2. Решение задачи, код программы 3](#_Toc136812929)

[1.3. Тестирование работы программы 3](#_Toc136812930)

[Задание 2 3](#_Toc136812931)

[2.1. Постановка задачи 3](#_Toc136812932)

[2.2. Решение задачи, код программы 4](#_Toc136812933)

[2.3. Тестирование работы программы 4](#_Toc136812934)

[Задание 3 4](#_Toc136812935)

[3.1. Постановка задачи 4](#_Toc136812936)

[3.2 Решение задачи, код программы 4](#_Toc136812937)

[3.3. Тестирование работы программы 5](#_Toc136812938)

[Задание 4 5](#_Toc136812939)

[4.1. Постановка задачи 5](#_Toc136812940)

[4.2. Решение задачи, код программы 5](#_Toc136812941)

[4.3. Тестирование работы программы 5](#_Toc136812942)

[Задание 5 6](#_Toc136812943)

[5.1. Постановка задачи 6](#_Toc136812944)

[5.2. Решение задачи, код программы 6](#_Toc136812945)

[5.3. Тестирование работы программы 7](#_Toc136812946)

# Задание 1

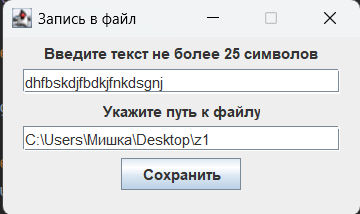
## 1.1. Постановка задачи

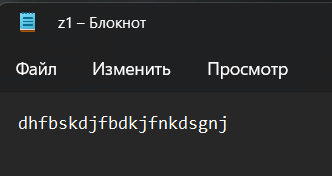
Создать окно с двумя полями ввода и кнопкой. У первого поля установить ограничение по количеству вводимых символов 25 символов. Второе поле использовать для указания пути к файлу. При нажатии на кнопку должна происходить запись данных из поля ввода в файл путь к которому указывается во втором поле ввода.

## 1.2. Решение задачи, код программы

import javax.swing.\*;  
import java.awt.\*;  
import java.awt.event.\*;  
import java.io.\*;  
public class z1 extends JFrame {  
 JButton b1;  
 JLabel l1, l2;  
 JTextField t1, t2;  
 eWork knopa = new eWork();  
 public z1(String s){  
 super(s);  
 setLayout(new FlowLayout());  
 b1 = new JButton("Сохранить");  
 l1 = new JLabel("Введите текст не более 25 символов");  
 l2 = new JLabel("Укажите путь к файлу");  
 t1 = new JTextField("", 25);  
 t2 = new JTextField("", 25);  
 add(l1);  
 add(t1);  
 add(l2);  
 add(t2);  
 add(b1);  
 b1.addActionListener(knopa);  
 }  
 public static void main(String[] args) {  
 z1 g = new z1("Запись в файл");  
 g.setVisible(true);  
 g.setDefaultCloseOperation(JFrame.*EXIT\_ON\_CLOSE*);  
 g.setSize(300, 180);  
 g.setLocationRelativeTo(null);  
 g.setResizable(false);  
 }  
  
 public class eWork implements ActionListener{  
 public void actionPerformed(ActionEvent e) {  
 if (t1.getText().length() > 25) {  
 JOptionPane.*showMessageDialog*(null, "Введите текст не более 25 символов!");  
 return;  
 }  
 String s1 = t1.getText();  
 String s2 = t2.getText();  
 try {  
 if (e.getSource() == b1) {  
 File file = new File(s2);  
 FileWriter fwr = new FileWriter(file);  
 fwr.write(s1);  
 fwr.flush();  
 fwr.close();  
 JOptionPane.*showMessageDialog*(null, "Данные успешно записаны в файл!");  
 }  
 } catch (IOException ex) {  
 JOptionPane.*showMessageDialog*(null, "Ошибка записи в файл: " + ex.getMessage());  
 }  
 }  
 }  
}

## 1.3. Тестирование работы программы





# Задание 2

## 2.1. Постановка задачи

Создать программу с графическим интерфейсом, которая предлагает два варианта использования

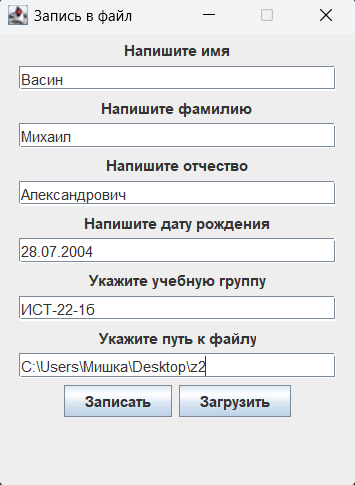
a. Отображает поля ввода имени, фамилии, отчества, даты рождения, учебной группы. При нажатии на кнопку «Записать» сохраняет данные в файл

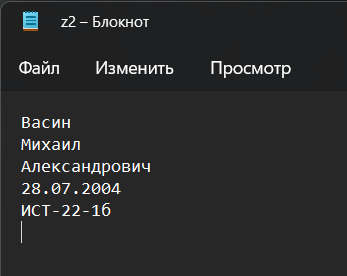
b. При нажатии кнопки «Загрузить» извлекает из файла данные и отображает в полях ввода.

## 2.2. Решение задачи, код программы

import javax.swing.\*;  
import java.awt.\*;  
import java.awt.event.\*;  
import java.io.\*;  
public class z2 extends JFrame {  
 JButton b1, b2;  
 JLabel l1, l2, l3, l4, l5, l6;  
 JTextField t1, t2, t3, t4, t5, t6;  
 eWork knopa = new eWork();  
 public z2(String s){  
 super(s);  
 setLayout(new FlowLayout());  
 b1 = new JButton("Записать");  
 b2 = new JButton("Загрузить");  
 l1 = new JLabel("Напишите имя");  
 l2 = new JLabel("Напишите фамилию");  
 l3 = new JLabel("Напишите отчество");  
 l4 = new JLabel("Напишите дату рождения");  
 l5 = new JLabel("Укажите учебную группу");  
 l6 = new JLabel("Укажите путь к файлу");  
 t1 = new JTextField("", 25);  
 t2 = new JTextField("", 25);  
 t3 = new JTextField("", 25);  
 t4 = new JTextField("", 25);  
 t5 = new JTextField("", 25);  
 t6 = new JTextField("", 25);  
 add(l1);  
 add(t1);  
 add(l2);  
 add(t2);  
 add(l3);  
 add(t3);  
 add(l4);  
 add(t4);  
 add(l5);  
 add(t5);  
 add(l6);  
 add(t6);  
 add(b1);  
 add(b2);  
 b1.addActionListener(knopa);  
 b2.addActionListener(knopa);  
 }  
 public static void main(String[] args) {  
 z2 g = new z2("Запись в файл");  
 g.setVisible(true);  
 g.setDefaultCloseOperation(JFrame.*EXIT\_ON\_CLOSE*);  
 g.setSize(300, 400);  
 g.setLocationRelativeTo(null);  
 g.setResizable(false);  
 }  
 public class eWork implements ActionListener{  
 public void actionPerformed(ActionEvent e) {  
 String s1 = t1.getText();  
 String s2 = t2.getText();  
 String s3 = t3.getText();  
 String s4 = t4.getText();  
 String s5 = t5.getText();  
 String s6 = t6.getText();  
 try {  
 if (e.getSource() == b1) {  
 File file = new File(s6);  
 FileWriter fwr = new FileWriter(file);  
 fwr.write(s1 + "\n");  
 fwr.write(s2 + "\n");  
 fwr.write(s3 + "\n");  
 fwr.write(s4 + "\n");  
 fwr.write(s5 + "\n");  
 fwr.flush();  
 fwr.close();  
 JOptionPane.*showMessageDialog*(null, "Данные успешно записаны в файл!");  
 }  
 } catch (Exception ex) {  
 JOptionPane.*showMessageDialog*(null, "Файл не найден: " + ex.getMessage());  
 }  
 try{  
 if (e.getSource() == b2) {  
 File file = new File(s6);  
 BufferedReader read = new BufferedReader(new FileReader(file));  
 t1.setText(read.readLine());  
 t2.setText(read.readLine());  
 t3.setText(read.readLine());  
 t4.setText(read.readLine());  
 t5.setText(read.readLine());  
 read.close();  
 JOptionPane.*showMessageDialog*(null, "Данные успешно загружены из файла!");  
 }  
 }catch (Exception ex){  
 JOptionPane.*showMessageDialog*(null, "Файл не найден!" + ex.getMessage());  
 }  
 }  
 }  
}

## 2.3. Тестирование работы программы





# Задание 3

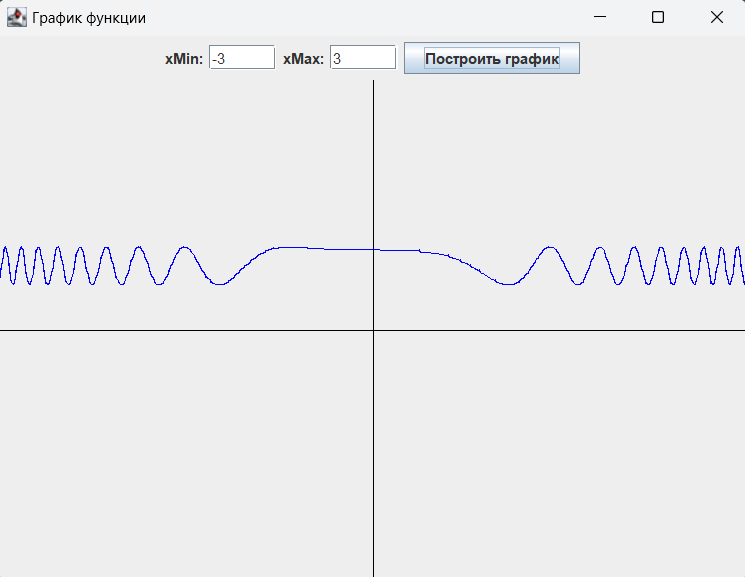
## 3.1. Постановка задачи

Создать программу, которая запрашивает у пользователя интервал и строит на нем график функции из задания №10 лабораторной работы «Циклы» 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 z3 extends JFrame implements ActionListener {  
 private JTextField xMinField, xMaxField;  
 private JButton plotButton;  
 private JPanel graphPanel;  
 public z3() {  
 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.*cos*(Math.*cos*(Math.*pow*(x,3)+5));  
 }  
 public static void main(String[] args) {  
 new z3();  
 }  
}

## 3.3. Тестирование работы программы



# Задание 4

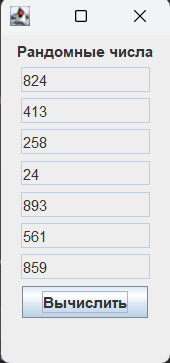
## 4.1. Постановка задачи

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 z4 extends JFrame {  
 JButton b1;  
 JLabel l1;  
 JTextField t1, t2, t3, t4, t5, t6, t7;  
 eWork knopa = new eWork();  
 public z4(String s){  
 super(s);  
 setLayout(new FlowLayout());  
 b1 = new JButton("Вычислить");  
 l1 = new JLabel("Рандомные числа");  
 t1 = new JTextField("",10);  
 t2 = new JTextField("",10);  
 t3 = new JTextField("",10);  
 t4 = new JTextField("",10);  
 t5 = new JTextField("",10);  
 t6 = new JTextField("",10);  
 t7 = new JTextField("",10);  
 t1.setEditable(false);  
 t2.setEditable(false);  
 t3.setEditable(false);  
 t4.setEditable(false);  
 t5.setEditable(false);  
 t6.setEditable(false);  
 t7.setEditable(false);  
 add(l1);  
 add(t1);  
 add(t2);  
 add(t3);  
 add(t4);  
 add(t5);  
 add(t6);  
 add(t7);  
 add(b1);  
 b1.addActionListener(knopa);  
 }  
 public static void main(String[] args) {  
 z4 g = new z4("Вычисление случ. чисел");  
 g.setVisible(true);  
 g.setDefaultCloseOperation(JFrame.*EXIT\_ON\_CLOSE*);  
 g.setSize(150, 300);  
 g.setLocationRelativeTo(null);  
 g.setResizable(true);  
 }  
 public class eWork implements ActionListener{  
 public void actionPerformed(ActionEvent e) {  
 try {  
 if (e.getSource() == b1) {  
 Random rand = new Random();  
 t1.setText(String.*valueOf*(rand.nextInt(1000)));  
 t2.setText(String.*valueOf*(rand.nextInt(1000)));  
 t3.setText(String.*valueOf*(rand.nextInt(1000)));  
 t4.setText(String.*valueOf*(rand.nextInt(1000)));  
 t5.setText(String.*valueOf*(rand.nextInt(1000)));  
 t6.setText(String.*valueOf*(rand.nextInt(1000)));  
 t7.setText(String.*valueOf*(rand.nextInt(1000)));  
 JOptionPane.*showMessageDialog*(null, "Данные успешно записаны!");  
 }  
 } catch (Exception ex) {  
 JOptionPane.*showMessageDialog*(null, "Ошибка! " + ex.getMessage());  
 }  
 }  
 }  
}

## 4.3. Тестирование работы программы



# Задание 5

## 5.1. Постановка задачи

Построить график фигуры попадания точки в область из задания №5 лабораторной работы «Ветвящиеся алгоритмы» Вашего варианта.

## 5.2. Решение задачи, код программы

import javax.swing.\*;  
import java.awt.\*;  
public class z5 extends JPanel {  
 @Override  
 protected void paintComponent(Graphics g) {  
 super.paintComponent(g);  
 Graphics2D g2d = (Graphics2D) g;  
 g2d.setColor(Color.*LIGHT\_GRAY*);  
 for (int i = 1; i <= 8; i++) {  
 int x = 100 \* i;  
 g2d.drawLine(x, 50, x, 500);  
 }  
 for (int i = 1; i <= 10; i++) {  
 int y = 500 - 50 \* i;  
 g2d.drawLine(50, y, 750, y);  
 }  
 g2d.setColor(Color.*RED*);  
 g2d.setStroke(new BasicStroke(2));  
 g2d.drawLine(150, 450, 100, 250);  
 g2d.drawLine(100, 250, 200, 200);  
 g2d.drawArc(200, 100, 200, 200, 0, 180);  
 g2d.drawLine(400, 200, 550, 150);  
 g2d.drawLine(550, 150, 350, 400);  
 g2d.drawLine(350, 400, 150, 450);  
 g2d.setColor(Color.*BLUE*);  
 g2d.setStroke(new BasicStroke(2));  
 g2d.drawLine(550, 250, 650, 150);  
 g2d.drawLine(650, 150, 700, 350);  
 g2d.drawArc(500, 250, 200, 200, 0, -180);  
 g2d.drawLine(500, 350, 600, 400);  
 g2d.drawLine(600, 400, 550, 250);  
 g2d.setColor(Color.*BLACK*);  
 g2d.setFont(new Font("Serif", Font.*PLAIN*, 12));  
 g2d.drawString("-6", 40, 525);  
 g2d.drawString("-4", 143, 525);  
 g2d.drawString("-2", 243, 525);  
 g2d.drawString("0", 346, 525);  
 g2d.drawString("2", 446, 525);  
 g2d.drawString("4", 546, 525);  
 g2d.drawString("6", 646, 525);  
 g2d.drawString("-3", 27, 503);  
 g2d.drawString("-2", 27, 453);  
 g2d.drawString("-1", 27, 403);  
 g2d.drawString("0", 27, 353);  
 g2d.drawString("1", 27, 303);  
 g2d.drawString("2", 27, 253);  
 g2d.drawString("3", 27, 203);  
 g2d.drawString("4", 27, 153);  
 g2d.drawString("5", 27, 103);  
 g2d.drawString("6", 27, 53);  
 }  
 public static void main(String[] args) {  
 SwingUtilities.*invokeLater*(new Runnable() {  
 public void run() {  
 *createAndShowGUI*();  
 }  
 });  
 }  
 private static void createAndShowGUI() {  
 JFrame frame = new JFrame("График");  
 frame.setDefaultCloseOperation(JFrame.*EXIT\_ON\_CLOSE*);  
 frame.setSize(800, 600);  
 frame.setLocationRelativeTo(null);  
 z5 panel = new z5();  
 frame.add(panel);  
 frame.setVisible(true);  
 }  
}

## 5.3. Тестирование работы программы

