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 в Java»

по дисциплине «Информатика»

Выполнил: студент группы ИСТ-22-1б Петраков Максим

Проверил: ассистент каф. ВММБ Нетбай Г.В.

Пермь, 2023

[Задание 1 3](#_Toc136764782)

[Задание 2 5](#_Toc136764783)

[Задание 3 8](#_Toc136764784)

[Задание 4 10](#_Toc136764785)

[Задание 5 12](#_Toc136764786)

Задание 1

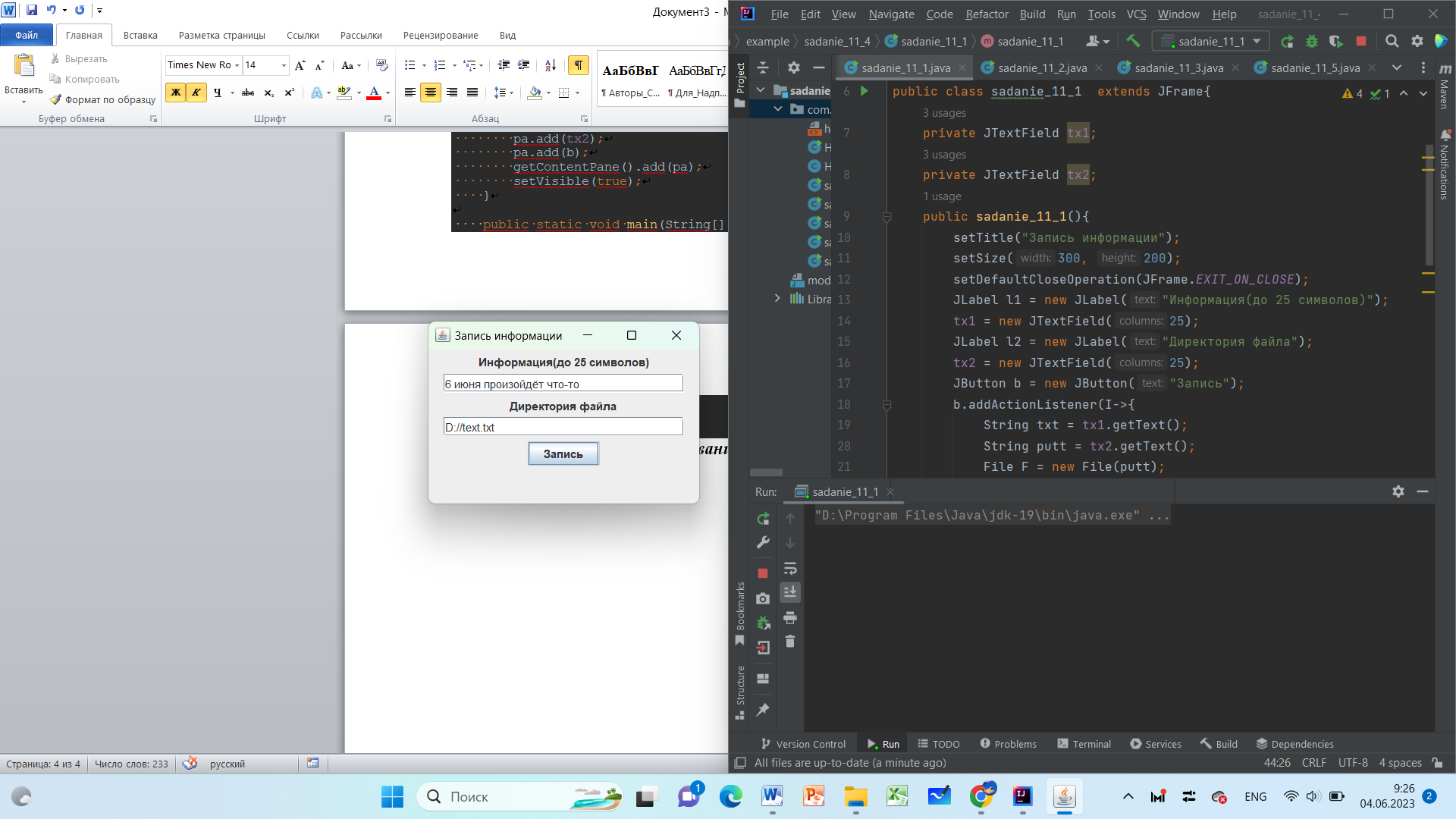
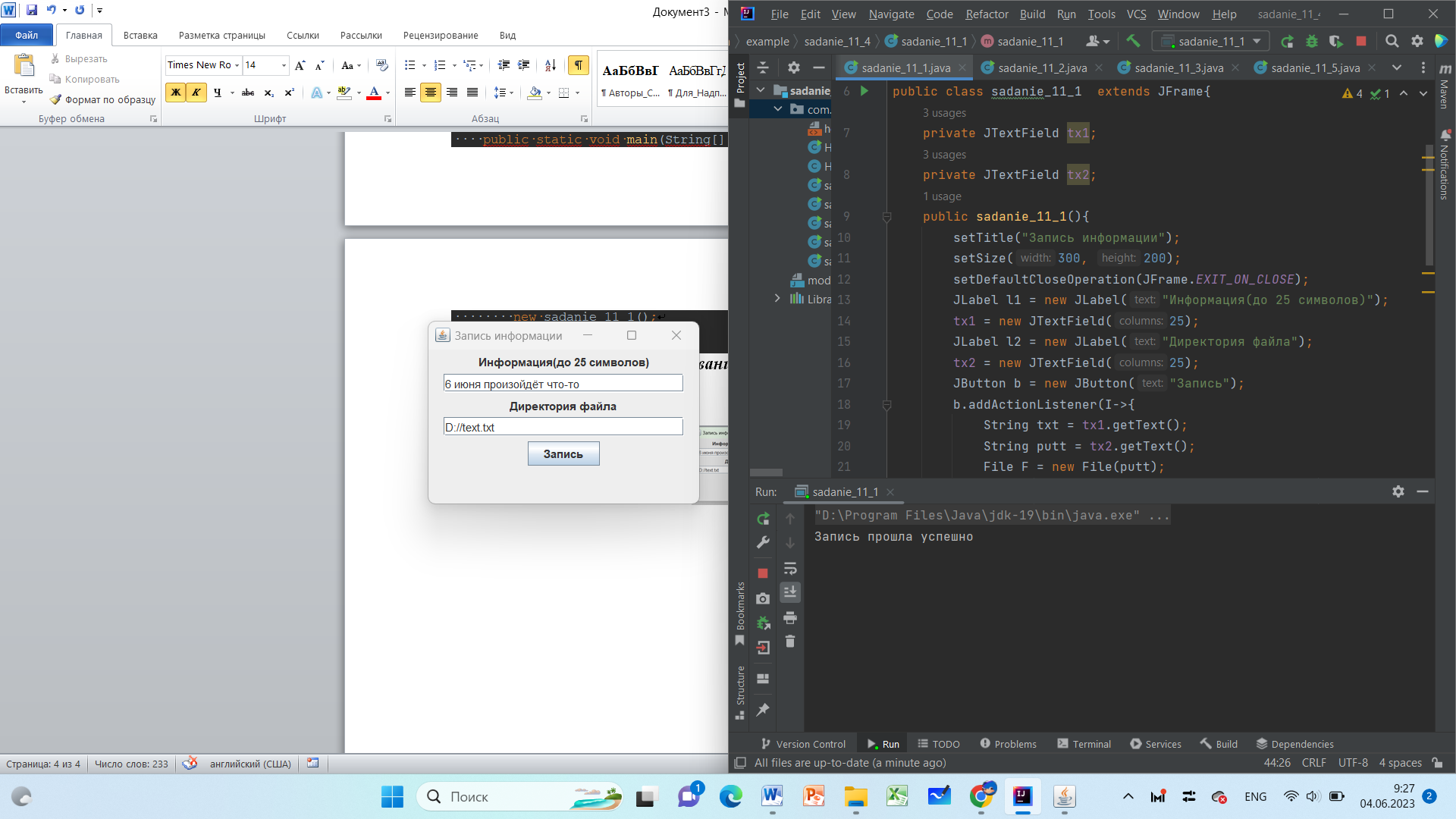
1.1. Постановка задания

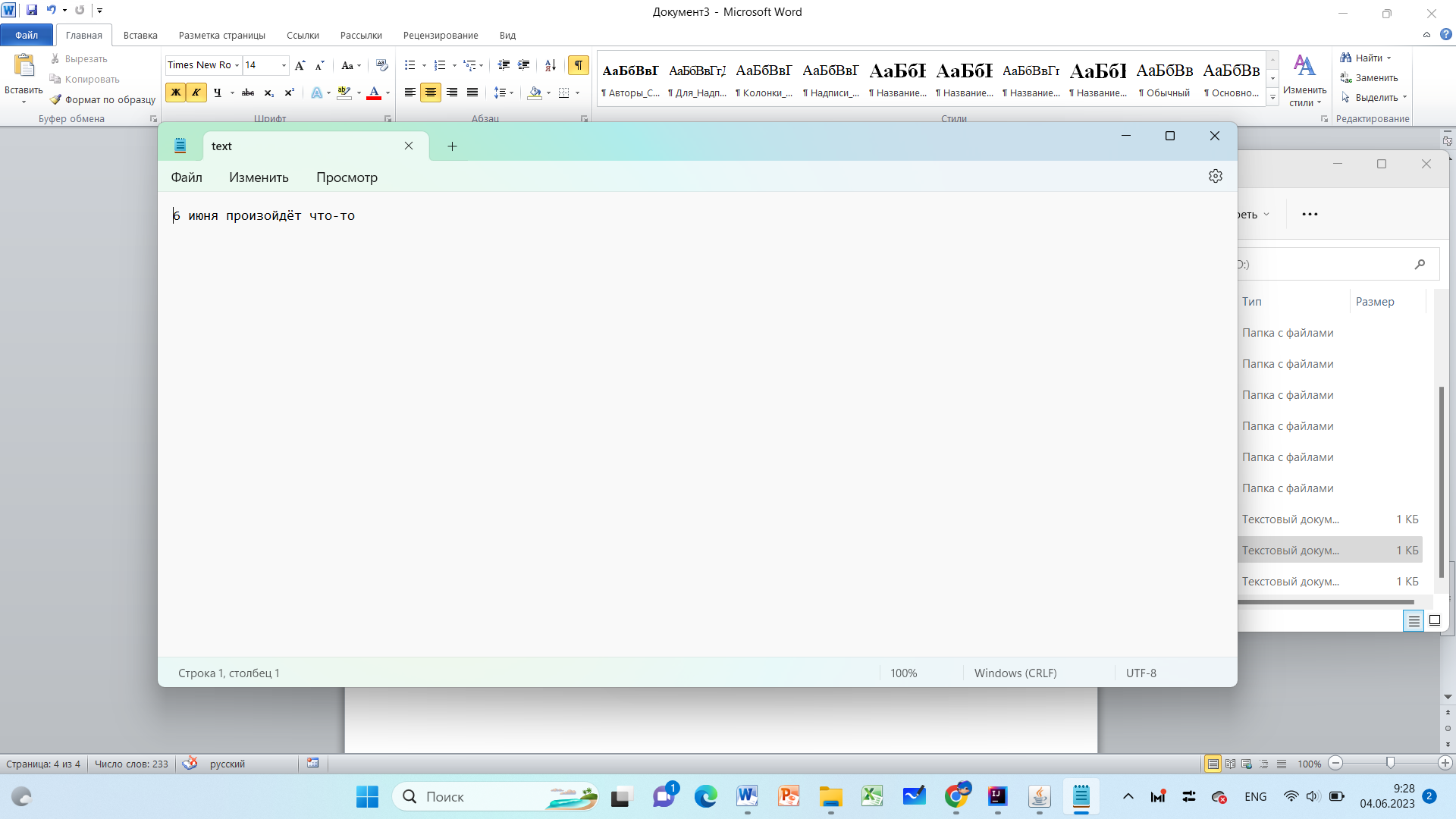
Создать окно с двумя полями ввода и кнопкой. У первого поля установить ограничение по количеству вводимых символов 25 символов. Второе поле использовать для указания пути к файлу. При нажатии на кнопку должна происходить запись данных из поля ввода в файл путь к которому указывается во втором поле ввода.

1.2. Решение задачи, код программы

package com.example.sadanie\_11\_4;  
  
import javax.swing.\*;  
import java.io.\*;  
import static java.lang.System.\*;  
public class sadanie\_11\_1 extends JFrame{  
 private JTextField tx1;  
 private JTextField tx2;  
 public sadanie\_11\_1(){  
 setTitle("Запись информации");  
 setSize(300, 200);  
 setDefaultCloseOperation(JFrame.*EXIT\_ON\_CLOSE*);  
 JLabel l1 = new JLabel("Информация(до 25 символов)");  
 tx1 = new JTextField(25);  
 JLabel l2 = new JLabel("Директория файла");  
 tx2 = new JTextField(25);  
 JButton b = new JButton("Запись");  
 b.addActionListener(I->{  
 String txt = tx1.getText();  
 String putt = tx2.getText();  
 File F = new File(putt);  
 try {  
 FileOutputStream FF = new FileOutputStream(F);  
 if(txt.length()<=25){  
 FF.flush();  
 byte c[] = txt.getBytes();  
 FF.write(c);  
 *out*.println("Запись прошла успешно");  
 }  
 } catch (FileNotFoundException e) {  
 *out*.println("Возникла ошибка");  
 } catch (IOException e) {  
 throw new RuntimeException(e);  
 }  
  
 });  
 JPanel pa = new JPanel();  
 pa.add(l1);  
 pa.add(tx1);  
 pa.add(l2);  
 pa.add(tx2);  
 pa.add(b);  
 getContentPane().add(pa);  
 setVisible(true);  
 }  
  
 public static void main(String[] args) {  
 new sadanie\_11\_1();  
 }  
}

1.3. Тестирование работы программы



Задание 2

2.1. Постановка задания

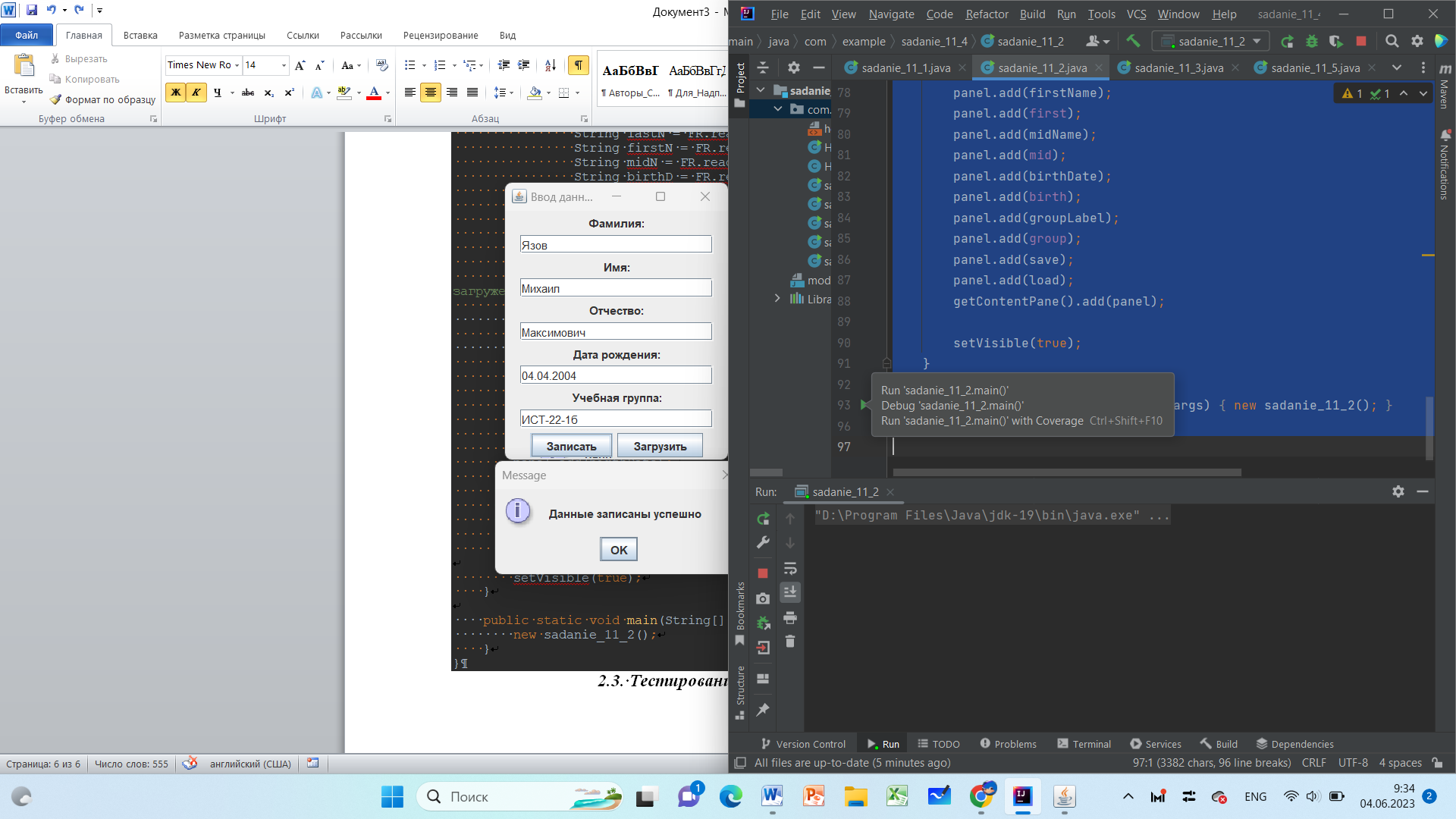
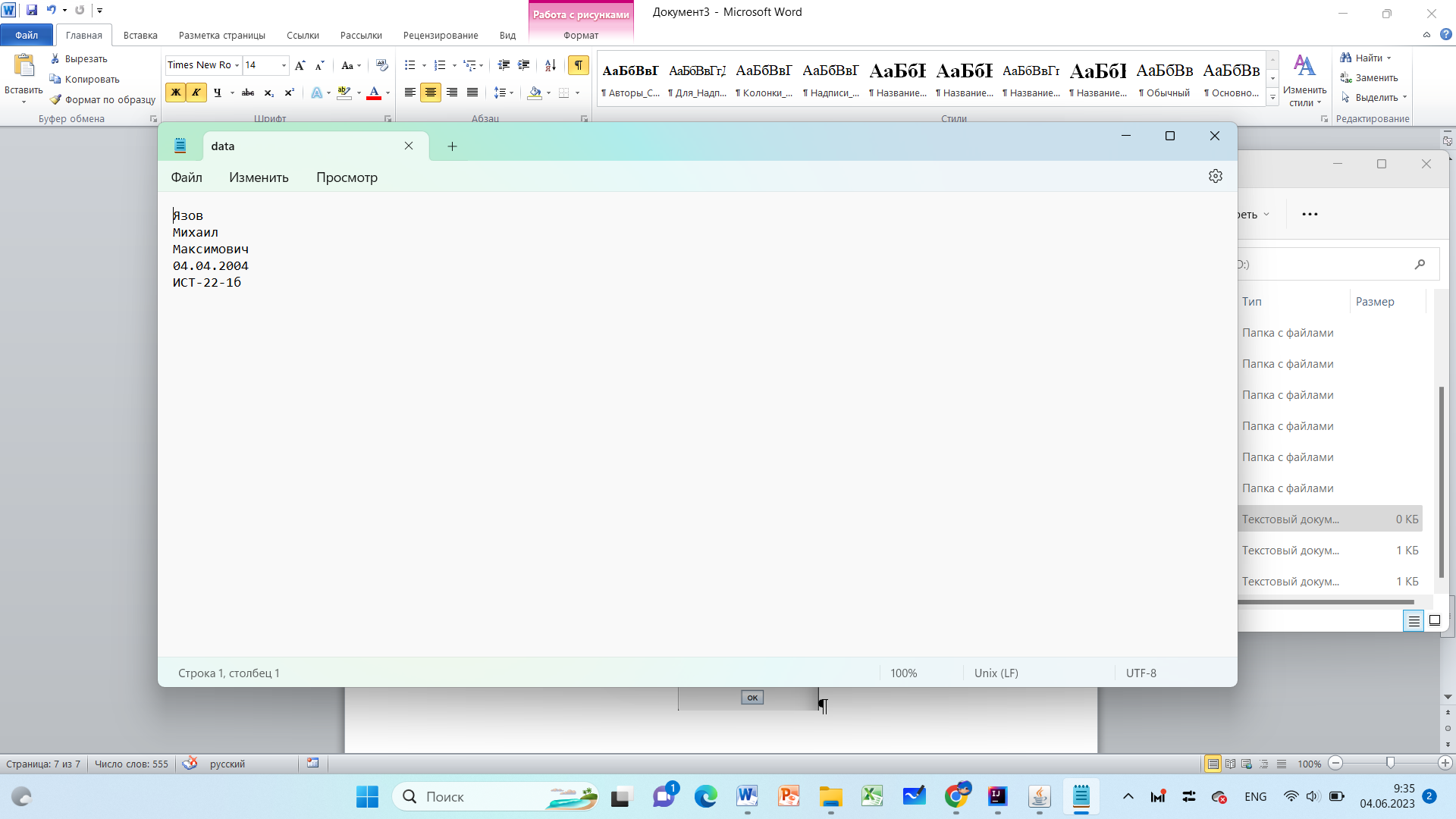
Создать программу с графическим интерфейсом, которая предлагает два варианта использования:

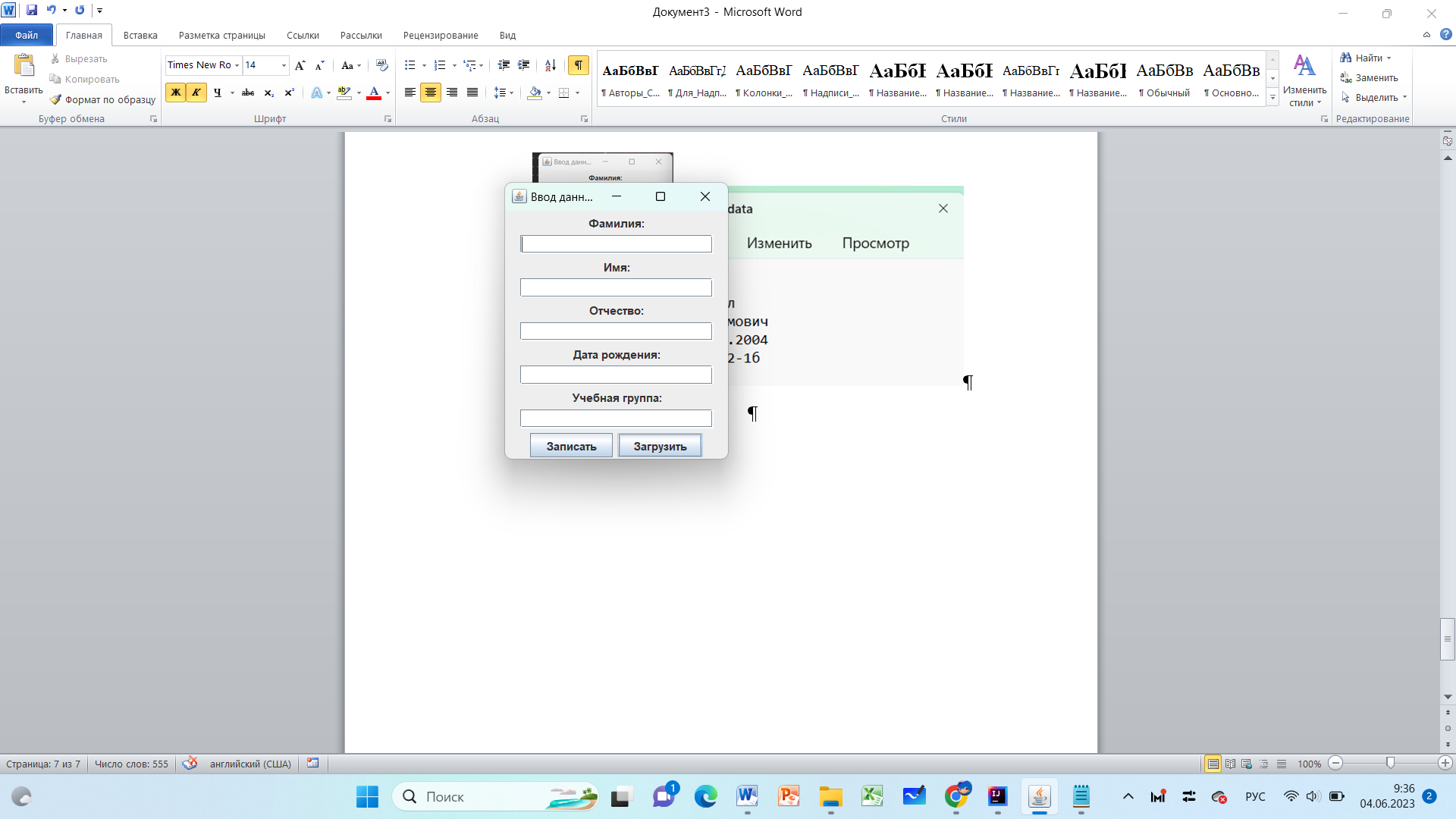
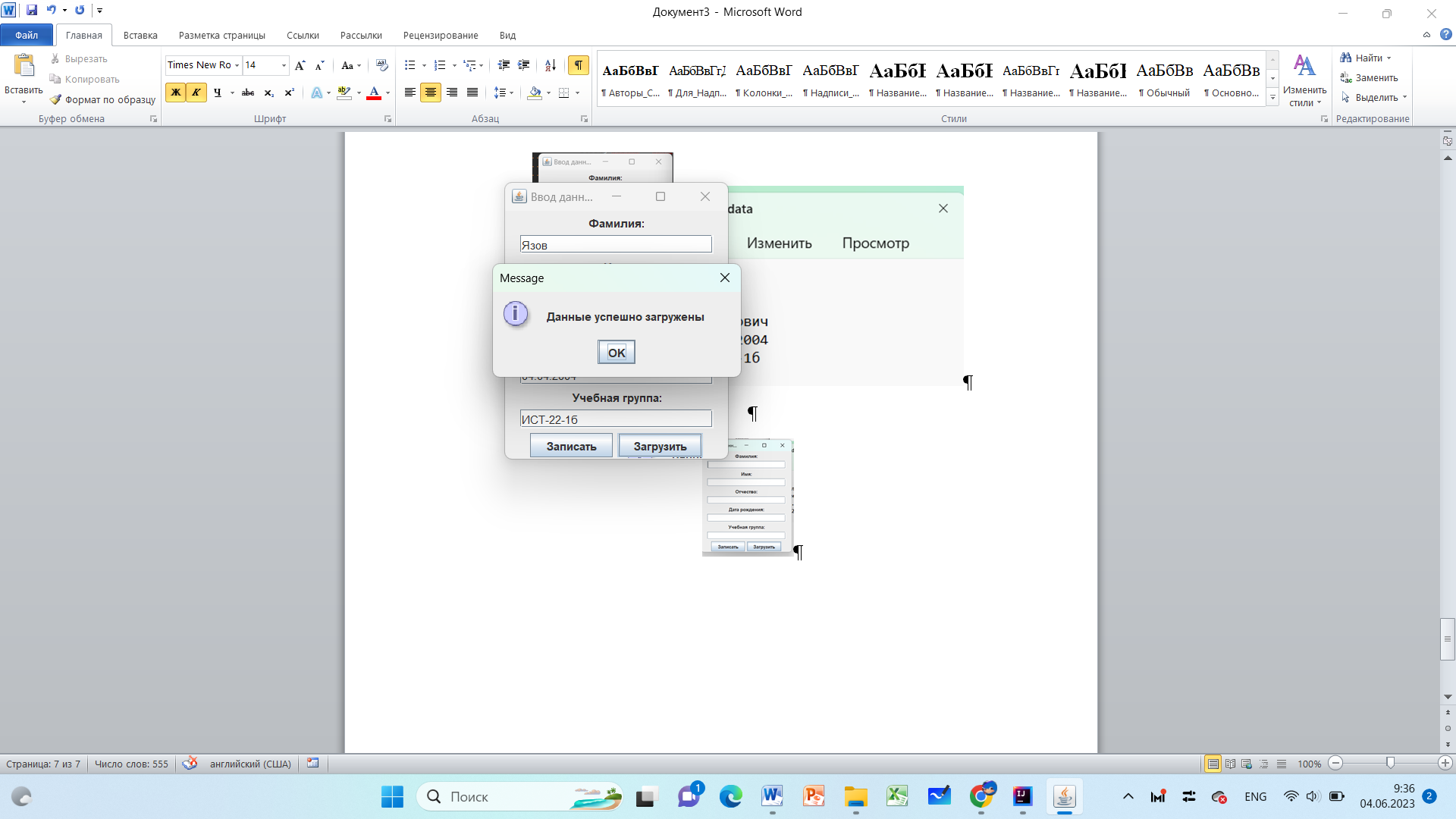
1. Отображает поля ввода имени, фамилии, отчества, даты рождения, учебной группы. При нажатии на кнопку «Записать» сохраняет данные в файл
2. При нажатии кнопки «Загрузить» извлекает из файла данные и отображает в полях ввода.

2.2. Решение задачи, код программы

package com.example.sadanie\_11\_4;  
  
import javax.swing.\*;  
import java.io.\*;  
  
public class sadanie\_11\_2 extends JFrame {  
 public JTextField last;  
 public JTextField first;  
 public JTextField mid;  
 public JTextField birth;  
 public JTextField group;  
  
 public sadanie\_11\_2() {  
 setTitle("Ввод данных");  
 setSize(250, 300);  
 setDefaultCloseOperation(JFrame.*EXIT\_ON\_CLOSE*);  
 JLabel sur = new JLabel("Фамилия:");  
 last = new JTextField(20);  
 JLabel firstName = new JLabel("Имя:");  
 first = new JTextField(20);  
 JLabel midName = new JLabel("Отчество:");  
 mid = new JTextField(20);  
 JLabel birthDate = new JLabel("Дата рождения:");  
 birth = new JTextField(20);  
 JLabel groupLabel = new JLabel("Учебная группа:");  
 group = new JTextField(20);  
 JButton save = new JButton("Записать");  
 JButton load = new JButton("Загрузить");  
 save.addActionListener(I -> {  
 String surN = last.getText();  
 String firstN = first.getText();  
 String midN = mid.getText();  
 String birthD = birth.getText();  
 String groupN = group.getText();  
 File F = new File("D://data.txt");  
 try {  
 FileOutputStream FF = new FileOutputStream(F);  
 FF.flush();  
 surN = surN + "\n";  
 firstN = firstN + "\n";  
 midN = midN + "\n";  
 birthD = birthD + "\n";  
 groupN = groupN + "\n";  
 FF.write(surN.getBytes());  
 FF.write(firstN.getBytes());  
 FF.write(midN.getBytes());  
 FF.write(birthD.getBytes());  
 FF.write(groupN.getBytes());  
 JOptionPane.*showMessageDialog*(this, "Данные записаны успешно");  
 } catch (FileNotFoundException e) {  
 throw new RuntimeException(e);  
 } catch (IOException e) {  
 JOptionPane.*showMessageDialog*(this, "Возникла ошибка");  
 }  
 });  
 load.addActionListener(I -> {  
 File F = new File("D://data.txt");  
 try {  
 BufferedReader FR = new BufferedReader(new FileReader(F));  
 String lastN = FR.readLine();  
 String firstN = FR.readLine();  
 String midN = FR.readLine();  
 String birthD = FR.readLine();  
 String groupN = FR.readLine();  
 last.setText(lastN);  
 first.setText(firstN);  
 mid.setText(midN);  
 birth.setText(birthD);  
 group.setText(groupN);  
 JOptionPane.*showMessageDialog*(this, "Данные успешно загружены");  
 } catch (IOException e) {  
 System.*out*.println("Возникла ошибка");  
 }  
 });  
 JPanel panel = new JPanel();  
 panel.add(sur);  
 panel.add(last);  
 panel.add(firstName);  
 panel.add(first);  
 panel.add(midName);  
 panel.add(mid);  
 panel.add(birthDate);  
 panel.add(birth);  
 panel.add(groupLabel);  
 panel.add(group);  
 panel.add(save);  
 panel.add(load);  
 getContentPane().add(panel);  
  
 setVisible(true);  
 }  
  
 public static void main(String[] args) {  
 new sadanie\_11\_2();  
 }  
}

2.3. Тестирование работы программы

Задание 3

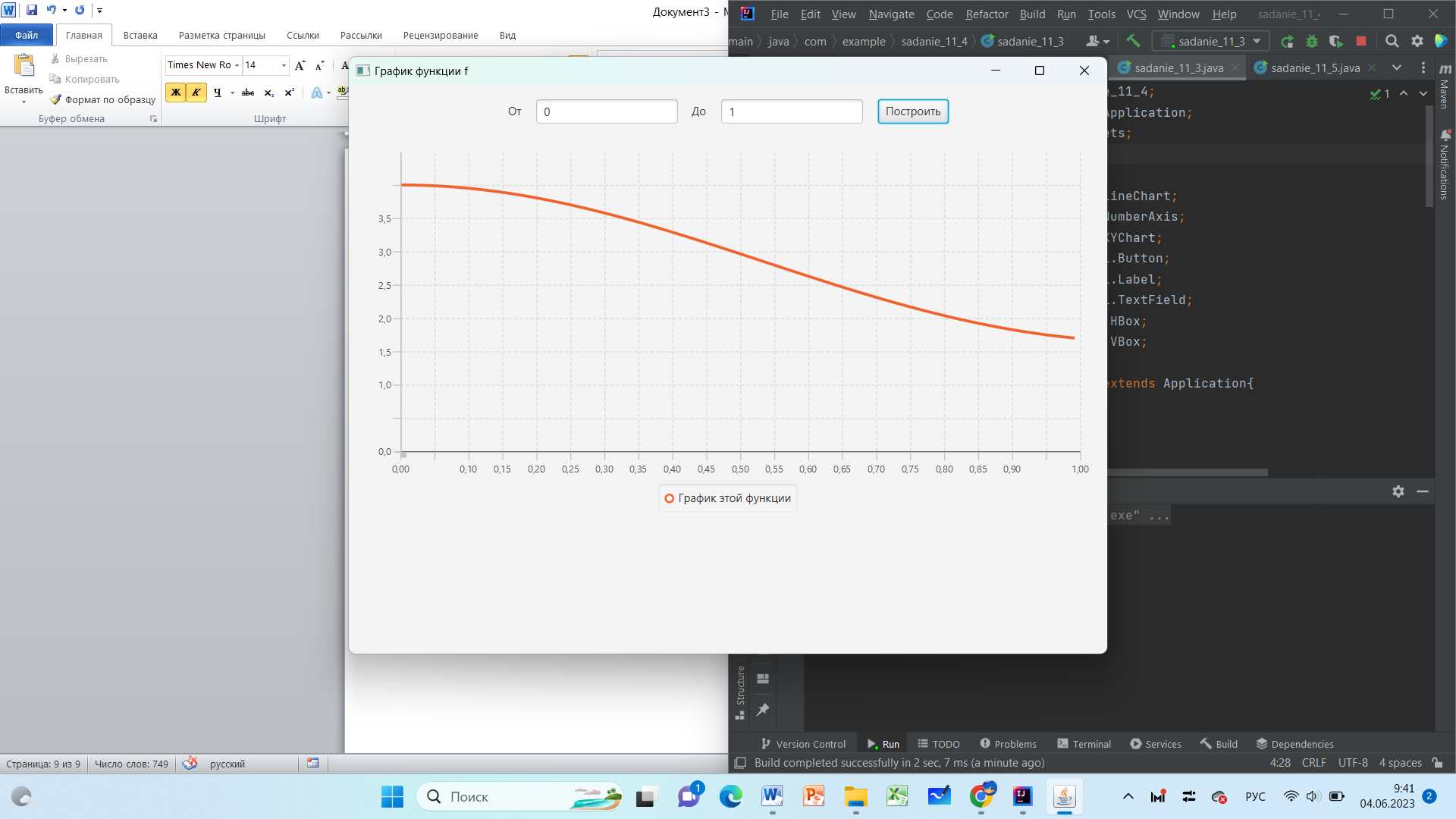
3.1. Постановка задания

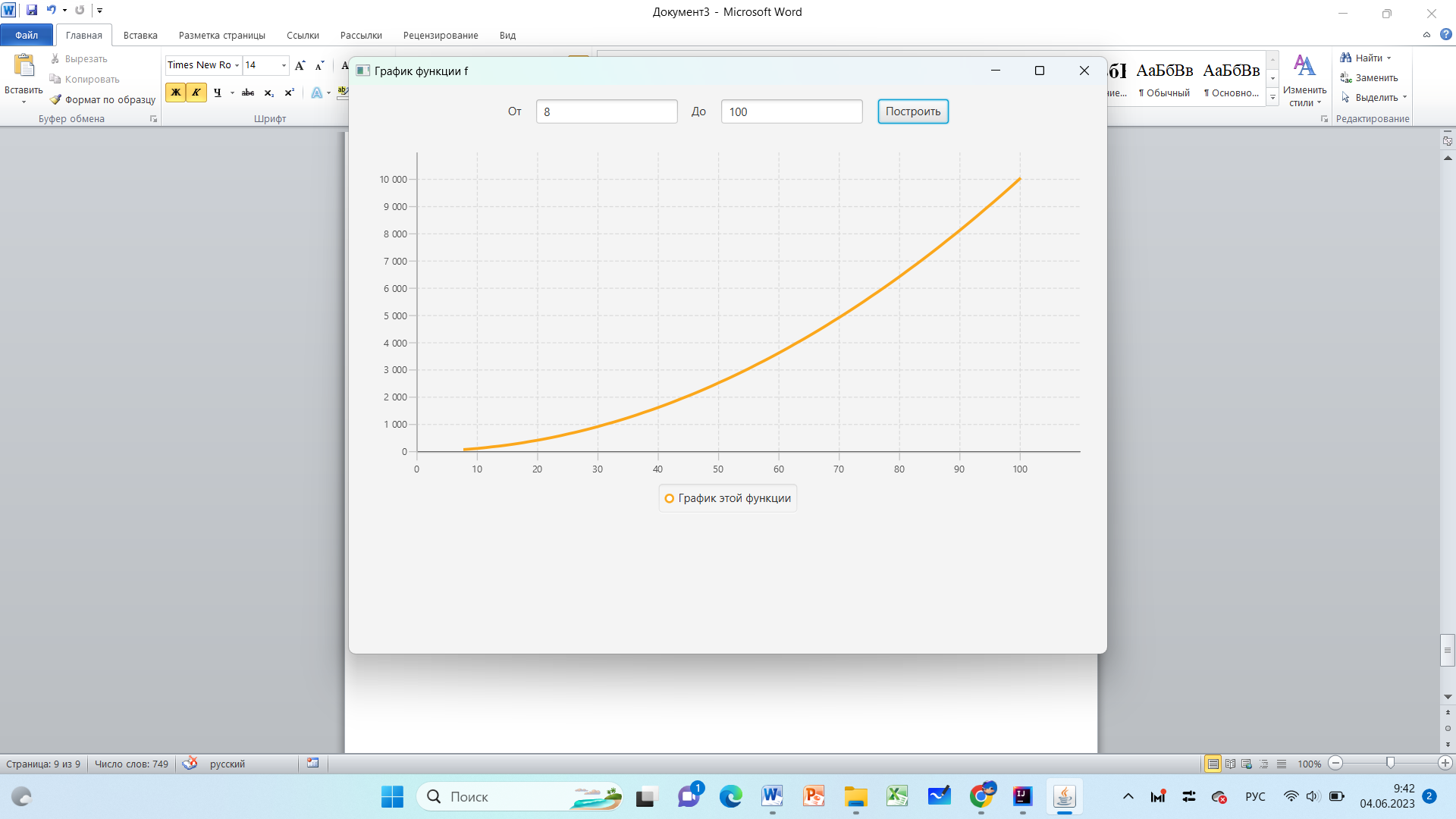
Создать программу, которая запрашивает у пользователя интервал и строит на нем график функции из задания №10 лабораторной работы «Циклы» вашего варианта.

3.2. Решение задачи, код программы

package com.example.sadanie\_11\_4;  
import javafx.application.Application;  
import javafx.geometry.Insets;  
import javafx.geometry.Pos;  
import javafx.scene.Scene;  
import javafx.scene.chart.LineChart;  
import javafx.scene.chart.NumberAxis;  
import javafx.scene.chart.XYChart;  
import javafx.scene.control.Button;  
import javafx.scene.control.Label;  
import javafx.scene.control.TextField;  
import javafx.scene.layout.HBox;  
import javafx.scene.layout.VBox;  
import javafx.stage.Stage;  
public class sadanie\_11\_3 extends Application{  
 TextField FTF;  
 TextField TTF;  
 Button button;  
 LineChart<Number, Number> lineChart;  
 @Override  
 public void start(Stage prime){  
 prime.setTitle("График функции f");  
 NumberAxis x = new NumberAxis();  
 NumberAxis y = new NumberAxis();  
 lineChart = new LineChart<>(x,y);  
 Label FL = new Label ("От");  
 Label TL = new Label ("До");  
 FTF = new TextField();  
 TTF = new TextField();  
 button = new Button("Построить");  
 button.setOnAction(I ->{  
 double fr = Double.*parseDouble*(FTF.getText());  
 double to = Double.*parseDouble*(TTF.getText());  
 XYChart.Series<Number, Number> series = new XYChart.Series<>();  
 series.setName("График этой функции");  
 for(double xx = fr; xx<=to; xx+=0.01){  
 double yy = Math.*pow*(xx,2)+Math.*pow*(Math.*cos*(xx)+Math.*pow*(Math.*cos*(xx),2),2);  
 series.getData().add(new XYChart.Data<>(xx,yy));  
 }  
 lineChart.getData().clear();  
 lineChart.setCreateSymbols(false);  
 lineChart.getData().add(series);  
 });  
 HBox IP = new HBox(15);  
 IP.setAlignment(Pos.*CENTER*);  
 IP.getChildren().addAll(FL,FTF,TL,TTF,button);  
 VBox MP = new VBox(15);  
 MP.setPadding(new Insets(15));  
 MP.getChildren().addAll(IP, lineChart);  
 Scene sc = new Scene(MP, 800, 600);  
 prime.setScene(sc);  
 prime.show();  
 }  
  
 public static void main(String[] args) {  
 *launch*(args);  
 }  
}

3.3. Тестирование работы программы





Задание 4

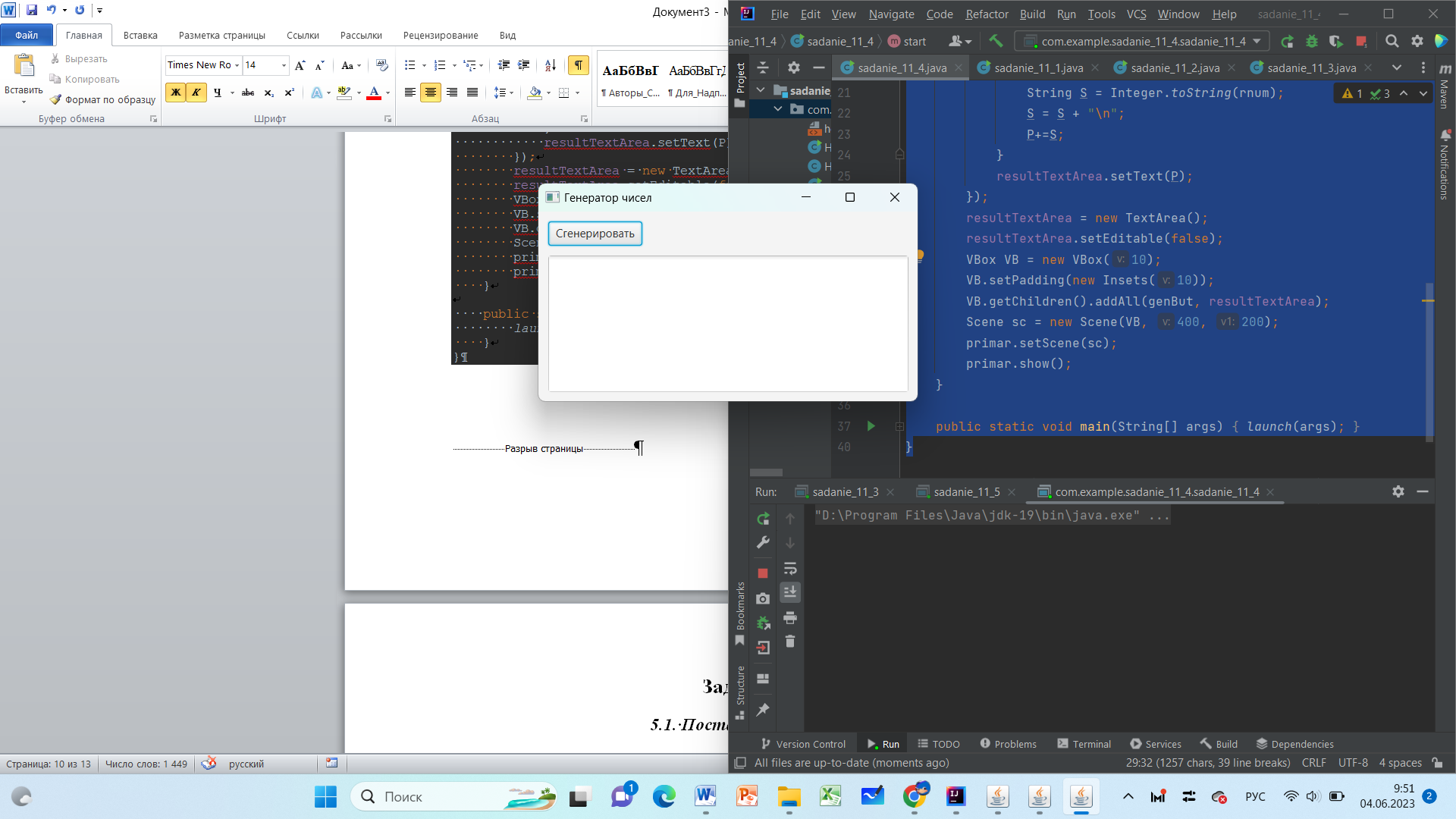
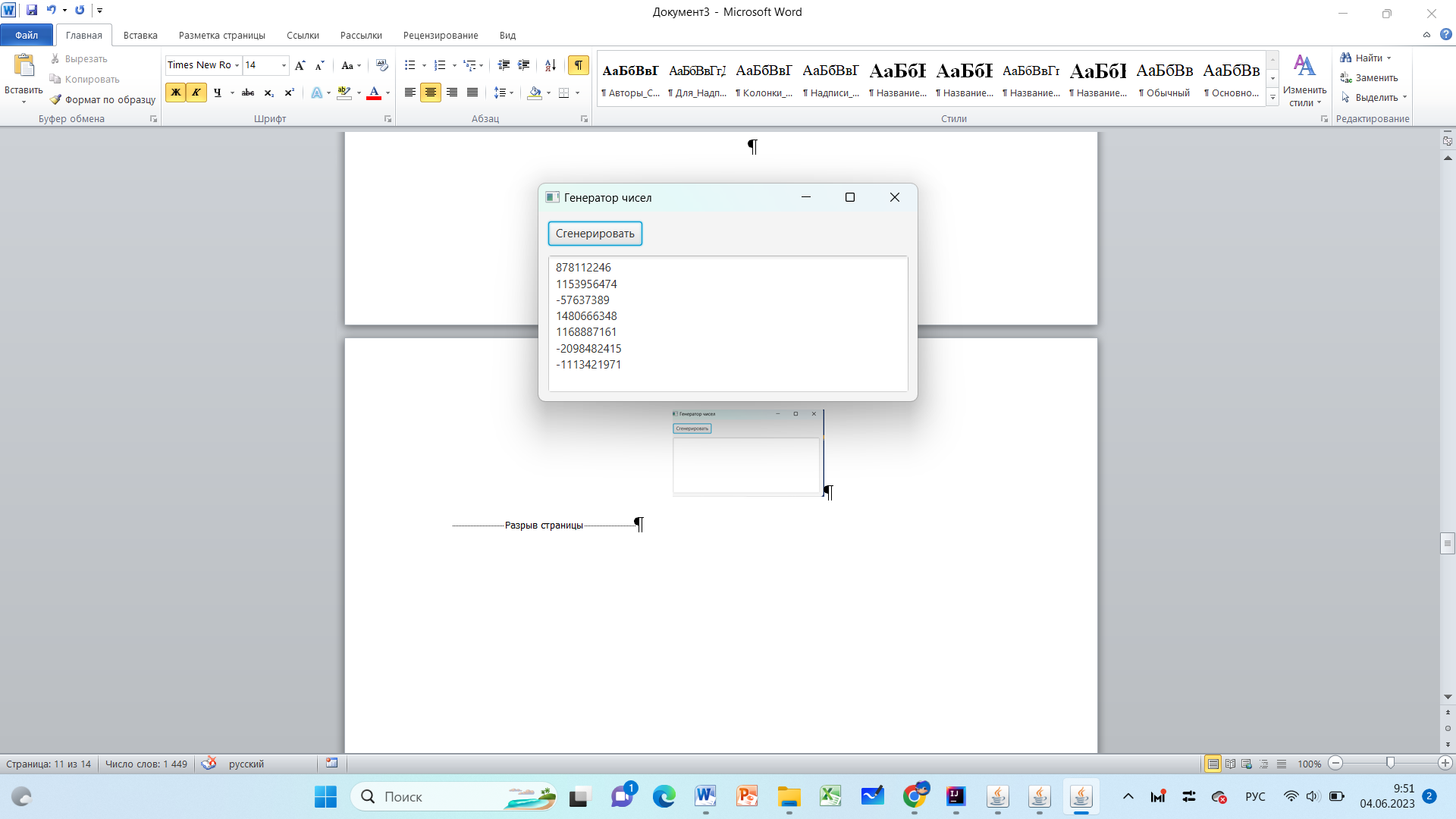
4.1. Постановка задания

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

package com.example.sadanie\_11\_4;  
import javafx.application.Application;  
import javafx.geometry.Insets;  
import javafx.scene.Scene;  
import javafx.scene.control.Button;  
import javafx.scene.control.TextArea;  
import javafx.scene.layout.VBox;  
import javafx.stage.Stage;  
import java.util.Random;  
public class sadanie\_11\_4 extends Application{  
 private TextArea resultTextArea;  
 @Override  
 public void start(Stage primar){  
 primar.setTitle("Генератор чисел");  
 Button genBut = new Button("Сгенерировать");  
 genBut.setOnAction(I -> {  
 Random r = new Random();  
 String P="";  
 for (int i = 0; i < 7; i++) {  
 int rnum = r.nextInt();  
 String S = Integer.*toString*(rnum);  
 S = S + "\n";  
 P+=S;  
 }  
 resultTextArea.setText(P);  
 });  
 resultTextArea = new TextArea();  
 resultTextArea.setEditable(false);  
 VBox VB = new VBox(10);  
 VB.setPadding(new Insets(10));  
 VB.getChildren().addAll(genBut, resultTextArea);  
 Scene sc = new Scene(VB, 400, 200);  
 primar.setScene(sc);  
 primar.show();  
 }  
  
 public static void main(String[] args) {  
 *launch*(args);  
 }  
}

4.3. Тестирование работы программы

Задание 5

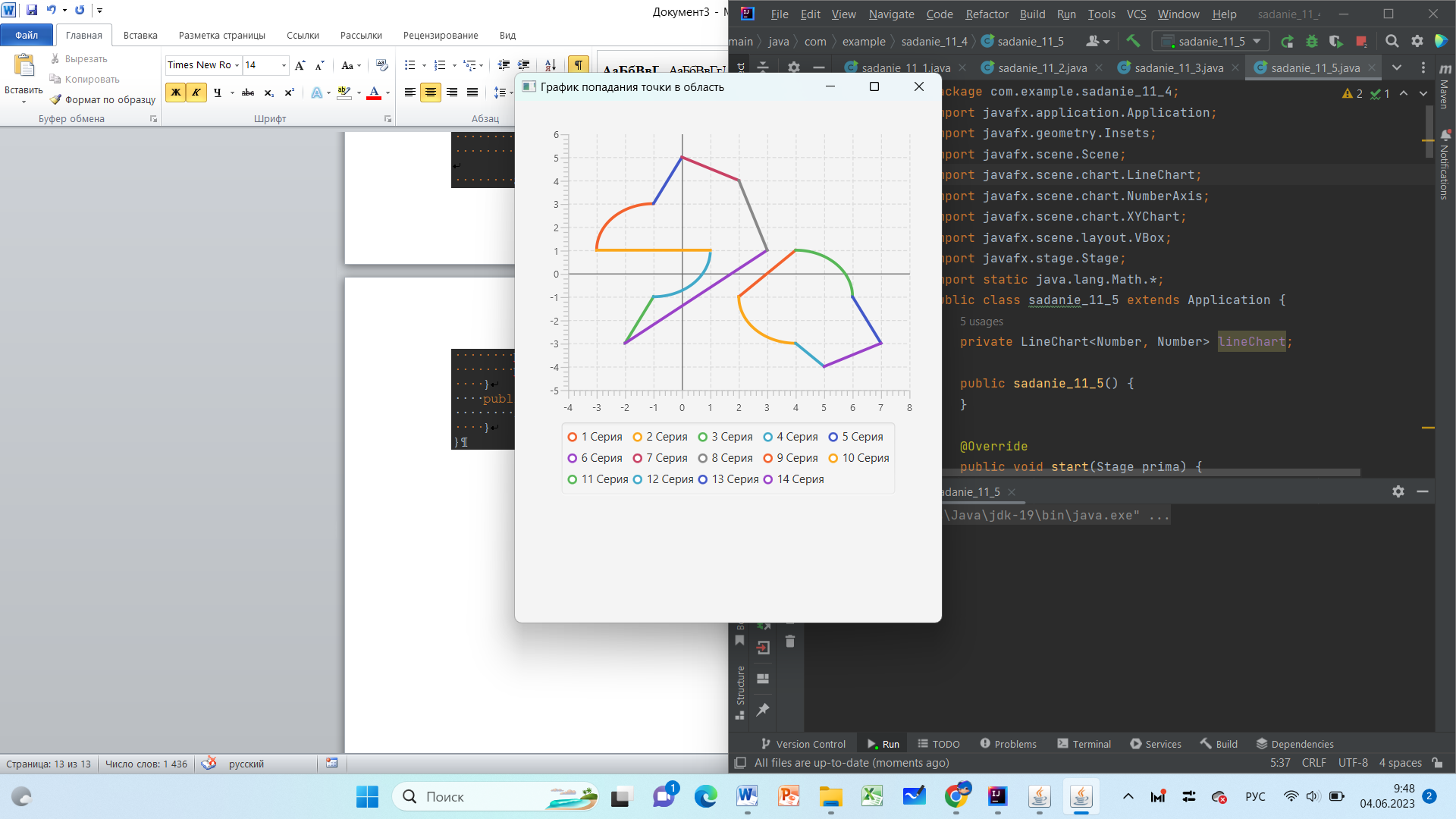
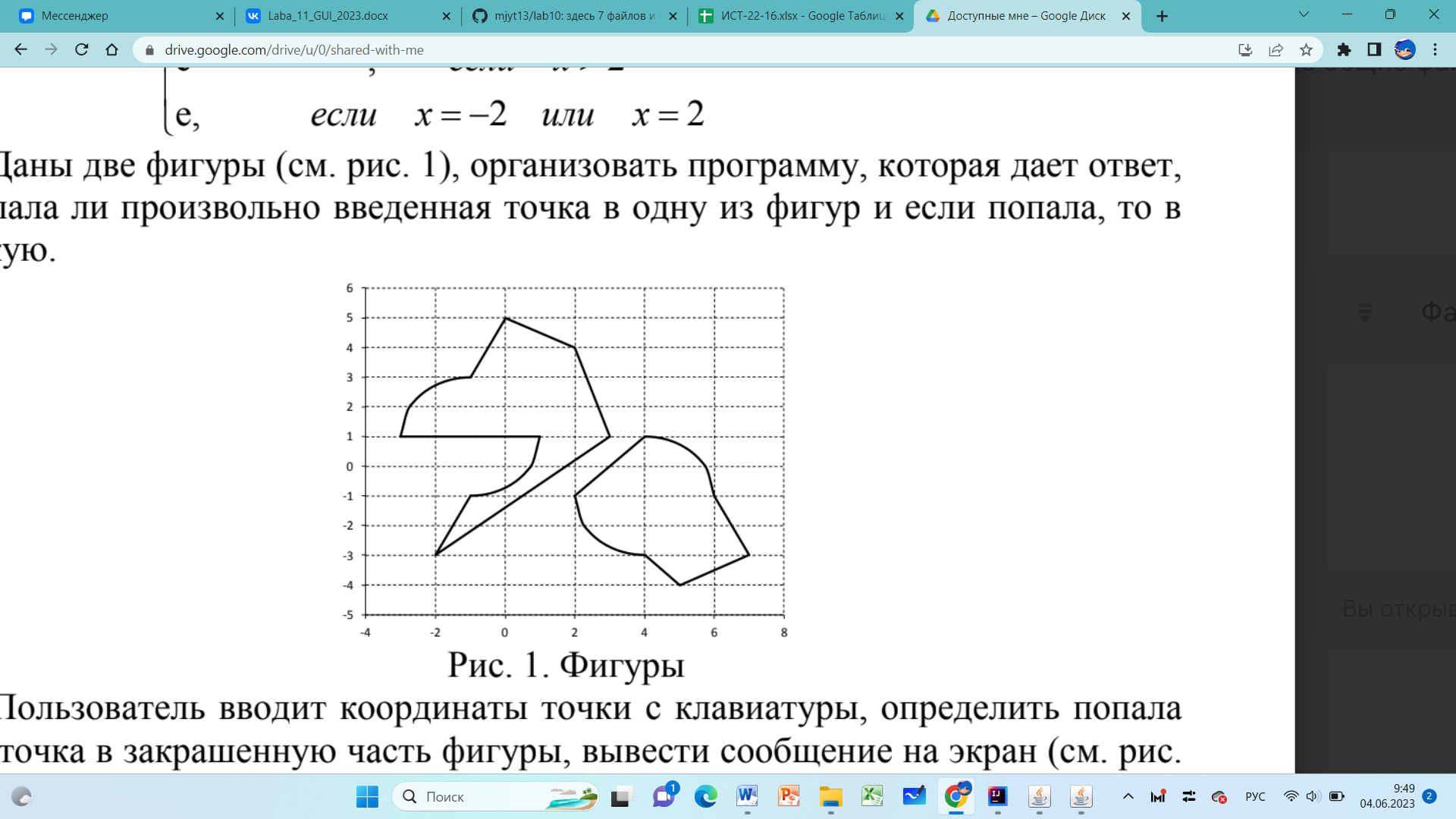
5.1. Постановка задания

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

5.2. Решение задачи, код программы

package com.example.sadanie\_11\_4;  
import javafx.application.Application;  
import javafx.geometry.Insets;  
import javafx.scene.Scene;  
import javafx.scene.chart.LineChart;  
import javafx.scene.chart.NumberAxis;  
import javafx.scene.chart.XYChart;  
import javafx.scene.layout.VBox;  
import javafx.stage.Stage;  
import static java.lang.Math.\*;  
public class sadanie\_11\_5 extends Application {  
 private LineChart<Number, Number> lineChart;  
  
 public sadanie\_11\_5() {  
 }  
  
 @Override  
 public void start(Stage prima) {  
 prima.setTitle("График попадания точки в область");  
 NumberAxis xA = new NumberAxis();  
 NumberAxis yA = new NumberAxis();  
 lineChart = new LineChart<>(xA, yA);  
 VBox main = new VBox(20);  
 main.setPadding(new Insets(20));  
 main.getChildren().add(lineChart);  
 XYChart.Series<Number, Number> ser1 = new XYChart.Series<>();  
 ser1.setName("1 Серия");  
 for (double x = -3; x <= -1; x += 0.005) {  
 double y = *sqrt*(4-*pow*(x+1,2)) + 1;  
 ser1.getData().add(new XYChart.Data<>(x, y));  
 }  
 XYChart.Series<Number, Number> ser2 = new XYChart.Series<>();  
 ser2.setName("2 Серия");  
 for (double x = -3; x <= 1; x += 0.005) {  
 double y = 1;  
 ser2.getData().add(new XYChart.Data<>(x, y));  
 }  
 XYChart.Series<Number, Number> ser3 = new XYChart.Series<>();  
 ser3.setName("3 Серия");  
 for (double x = -2; x <= -1; x += 0.005) {  
 double y = 2\*x+1;  
 ser3.getData().add(new XYChart.Data<>(x, y));  
 }  
 XYChart.Series<Number, Number> ser4 = new XYChart.Series<>();  
 ser4.setName("4 Серия");  
 for (double x = -1; x <= 1; x += 0.005) {  
 double y = -*sqrt*(4-*pow*(x+1,2))+1;  
 ser4.getData().add(new XYChart.Data<>(x, y));  
 }  
 XYChart.Series<Number, Number> ser5 = new XYChart.Series<>();  
 ser5.setName("5 Серия");  
 for (double x = -1; x <= 0; x += 0.005) {  
 double y = 2\*x+5;  
 ser5.getData().add(new XYChart.Data<>(x, y));  
 }  
 XYChart.Series<Number, Number> ser6 = new XYChart.Series<>();  
 ser6.setName("6 Серия");  
 for (double x = -2; x <= 3; x += 0.01) {  
 double y = (4/5.0)\*x-(7/5.0);  
 ser6.getData().add(new XYChart.Data<>(x, y));  
 }  
 XYChart.Series<Number, Number> ser7 = new XYChart.Series<>();  
 ser7.setName("7 Серия");  
 for (double x = 0; x <= 2; x += 0.005) {  
 double y = (-1/2.0)\*x+5;  
 ser7.getData().add(new XYChart.Data<>(x, y));  
 }  
 XYChart.Series<Number, Number> ser8 = new XYChart.Series<>();  
 ser8.setName("8 Серия");  
 for (double x = 2; x <= 3; x += 0.005) {  
 double y = -3 \* x + 10;  
 ser8.getData().add(new XYChart.Data<>(x, y));  
 }  
 XYChart.Series<Number, Number> ser9 = new XYChart.Series<>();  
 ser9.setName("9 Серия");  
 for (double x = 2; x <= 4; x += 0.005) {  
 double y = x-3;  
 ser9.getData().add(new XYChart.Data<>(x, y));  
 }  
 XYChart.Series<Number, Number> ser10 = new XYChart.Series<>();  
 ser10.setName("10 Серия");  
 for (double x = 2; x <= 4; x += 0.005) {  
 double y = -Math.*sqrt*(4-*pow*(x-4,2)) - 1;  
 ser10.getData().add(new XYChart.Data<>(x, y));  
 }  
 XYChart.Series<Number, Number> ser11 = new XYChart.Series<>();  
 ser11.setName("11 Серия");  
 for (double x = 4; x <= 6; x += 0.005) {  
 double y = Math.*sqrt*(4-*pow*(x-4,2)) - 1;  
 ser11.getData().add(new XYChart.Data<>(x, y));  
 }  
 XYChart.Series<Number, Number> ser12 = new XYChart.Series<>();  
 ser12.setName("12 Серия");  
 for (double x = 4; x <= 5; x += 0.005) {  
 double y = -x+1;  
 ser12.getData().add(new XYChart.Data<>(x, y));  
 }  
 XYChart.Series<Number, Number> ser13 = new XYChart.Series<>();  
 ser13.setName("13 Серия");  
 for (double x = 6; x <= 7; x += 0.005) {  
 double y = -2\*x+11;  
 ser13.getData().add(new XYChart.Data<>(x, y));  
 }  
 XYChart.Series<Number, Number> ser14 = new XYChart.Series<>();  
 ser14.setName("14 Серия");  
 for (double x = 5; x <= 7; x += 0.01) {  
 double y = (1/2.0)\*x-(13/2.0);  
 ser14.getData().add(new XYChart.Data<>(x, y));  
 }  
 lineChart.getData().clear();  
 lineChart.setCreateSymbols(false);  
 lineChart.getData().addAll(ser1, ser2, ser3, ser4, ser5,  
 ser6, ser7, ser8, ser9, ser10,  
 ser11, ser12, ser13, ser14);  
  
 Scene sc = new Scene(main, 450, 550);  
 prima.setScene(sc);  
 prima.show();  
 }  
 public static void main(String[] args) {  
 *launch*(args);  
 }  
}

5.3 Тестирование работы программы с проверкой

Видно, что графики построенный и изначальный совпадают, что говорит о правильности насписанной программы.