Пензенский государственный университет

Кафедра «Вычислительная техника»

**ОТЧЕТ**

по лабораторной работе №4

по курсу «Разработка кроссплатформенных приложений»

на тему «Работа с файлами»

**Выполнил студент группы 20ВВВ1:**

Ионцев В.А.

**Приняли:**

Юрова О.В.

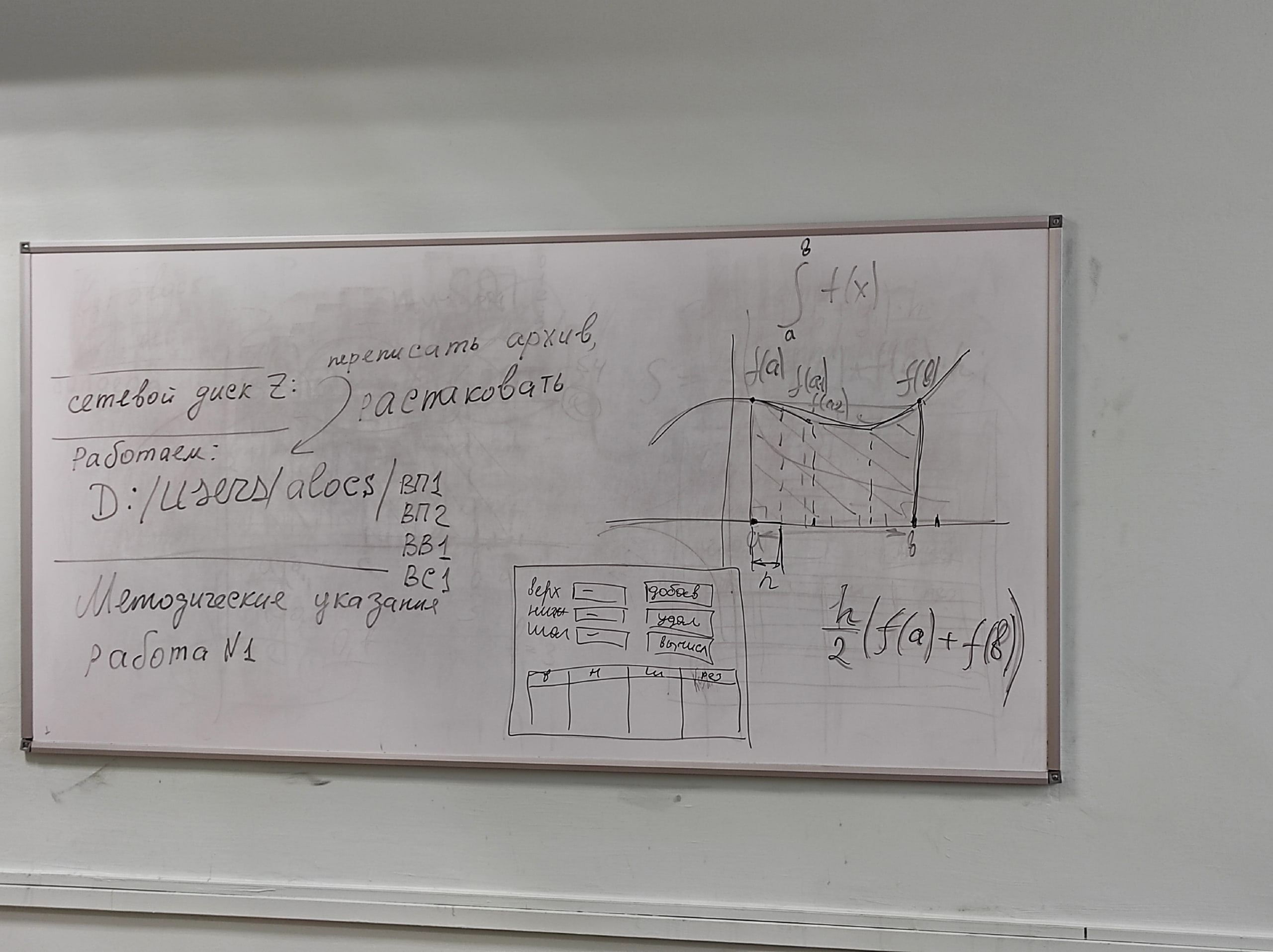
Карамышева Н.С.

Пенза 2023 г.

**Цель работы:** изучить работу с файлами и механизмы сериализации данных.

**Задание:**

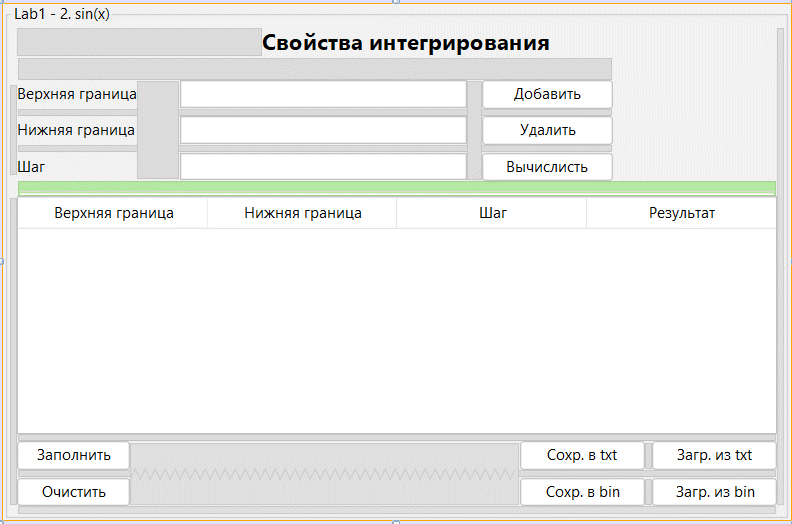
|  |  |
| --- | --- |
| Номер варианта | Функция |
| 2 |  |



Модифицировать приложение из предыдущей лабораторной работы, реализовав сохранение в файл и загрузку данных из файла. Предусмотреть сохранение данных, как в текстовом виде, так и в двоичном (с использованием механизма сериализации). Для этого нужно добавить 4 кнопки для сохранения и загрузки в текстовом и двоичном виде соответственно. Кроме того, в программе нужно предусмотреть использование стандартного диалога открытия файла (JFileChooser).

**Ход работы:**

1. Добавили кнопки «Сохр. в/загр. из txt» и «Сохр. в/загр. из bin».

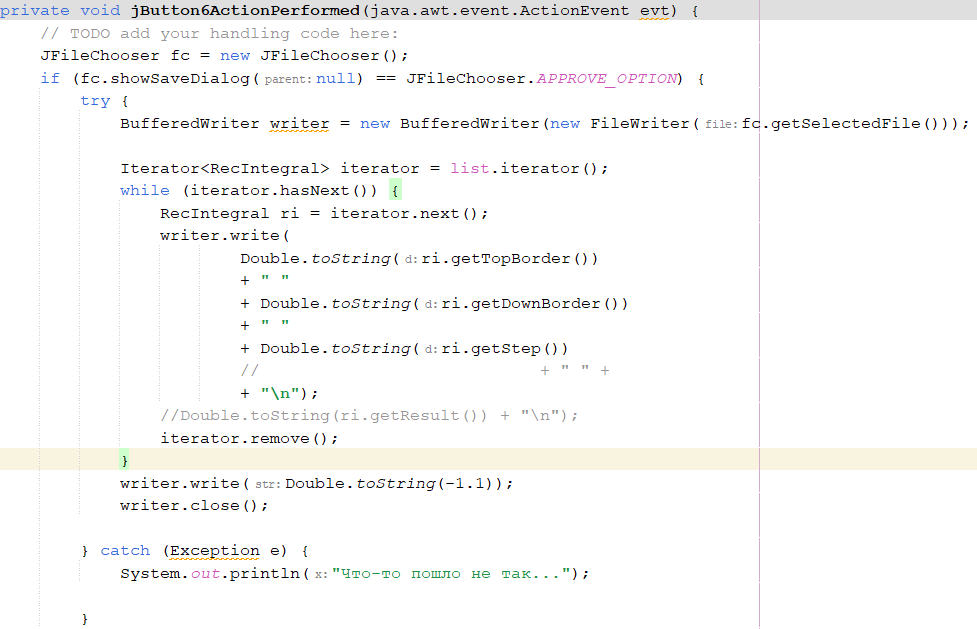


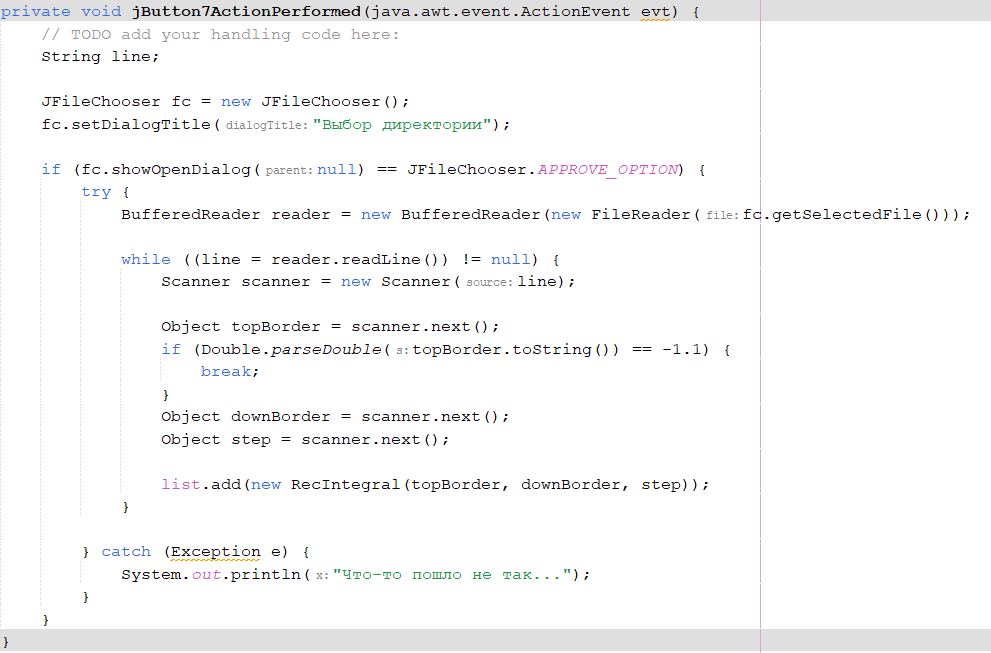
1. Дополнили класс RecIntegral.



1. Прописали функционал для добавленных кнопок.

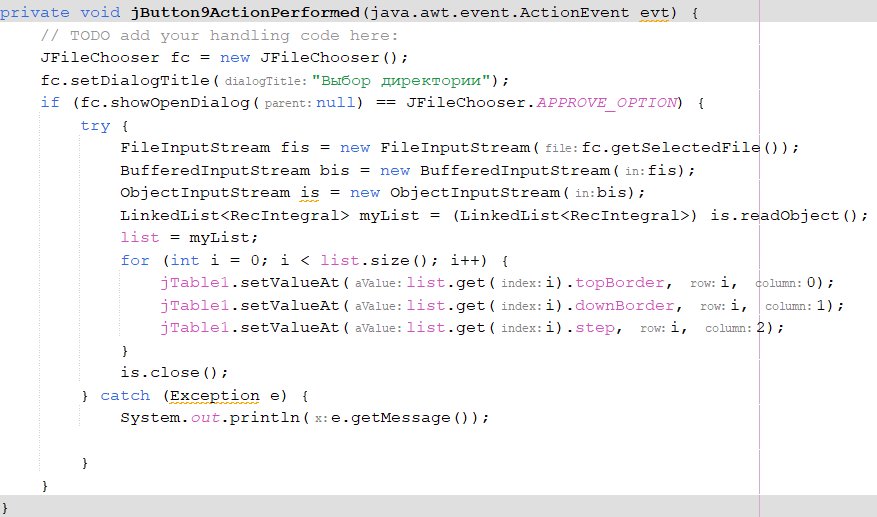
Сохр. в/загр. из txt





Сохр. в/загр. из bin





**Листинг:**

**RecIntegral.java**

public class RecIntegral implements Serializable

**ContactEditorUI.java**

private void jButton6ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

JFileChooser fc = new JFileChooser();

if (fc.showSaveDialog(null) == JFileChooser.APPROVE\_OPTION) {

try {

BufferedWriter writer = new BufferedWriter(new FileWriter(fc.getSelectedFile()));

Iterator<RecIntegral> iterator = list.iterator();

while (iterator.hasNext()) {

RecIntegral ri = iterator.next();

writer.write(

Double.toString(ri.getTopBorder())

+ " "

+ Double.toString(ri.getDownBorder())

+ " "

+ Double.toString(ri.getStep())

// + " " +

+ "\n");

//Double.toString(ri.getResult()) + "\n");

iterator.remove();

}

writer.write(Double.toString(-1.1));

writer.close();

} catch (Exception e) {

System.out.println("Что-то пошло не так...");

}

}

}

private void jButton7ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

String line;

JFileChooser fc = new JFileChooser();

fc.setDialogTitle("Выбор директории");

if (fc.showOpenDialog(null) == JFileChooser.APPROVE\_OPTION) {

try {

BufferedReader reader = new BufferedReader(new FileReader(fc.getSelectedFile()));

while ((line = reader.readLine()) != null) {

Scanner scanner = new Scanner(line);

Object topBorder = scanner.next();

if (Double.parseDouble(topBorder.toString()) == -1.1) {

break;

}

Object downBorder = scanner.next();

Object step = scanner.next();

list.add(new RecIntegral(topBorder, downBorder, step));

}

} catch (Exception e) {

System.out.println("Что-то пошло не так...");

}

}

}

private void jButton8ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

JFileChooser fc = new JFileChooser();

if (fc.showSaveDialog(null) == JFileChooser.APPROVE\_OPTION) {

try {

FileOutputStream fos = new FileOutputStream(fc.getSelectedFile());

BufferedOutputStream bos = new BufferedOutputStream(fos);

ObjectOutputStream os = new ObjectOutputStream(bos);

os.writeObject(list);

os.flush();

os.close();

} catch (Exception e) {

System.out.println(e.getMessage());

}

}

}

private void jButton9ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

JFileChooser fc = new JFileChooser();

fc.setDialogTitle("Выбор директории");

if (fc.showOpenDialog(null) == JFileChooser.APPROVE\_OPTION) {

try {

FileInputStream fis = new FileInputStream(fc.getSelectedFile());

BufferedInputStream bis = new BufferedInputStream(fis);

ObjectInputStream is = new ObjectInputStream(bis);

LinkedList<RecIntegral> myList = (LinkedList<RecIntegral>) is.readObject();

list = myList;

for (int i = 0; i < list.size(); i++) {

jTable1.setValueAt(list.get(i).topBorder, i, 0);

jTable1.setValueAt(list.get(i).downBorder, i, 1);

jTable1.setValueAt(list.get(i).step, i, 2);

}

is.close();

} catch (Exception e) {

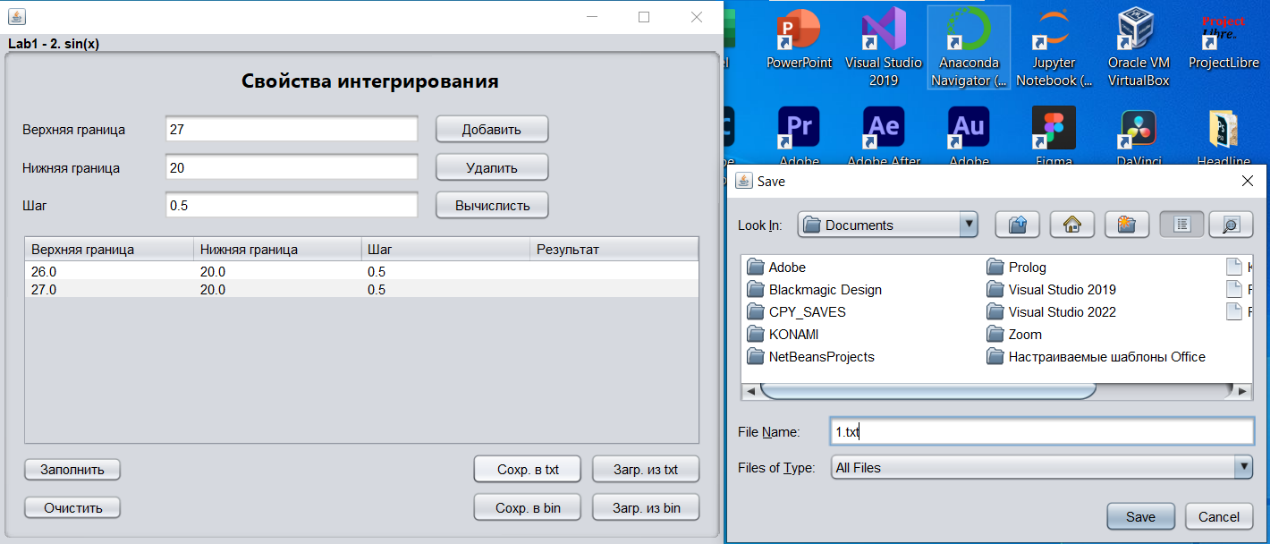
System.out.println(e.getMessage());

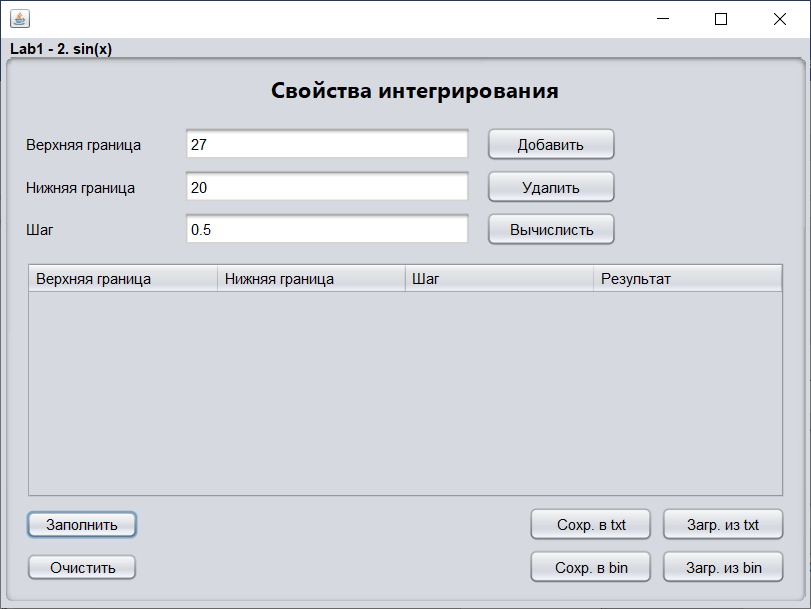
}

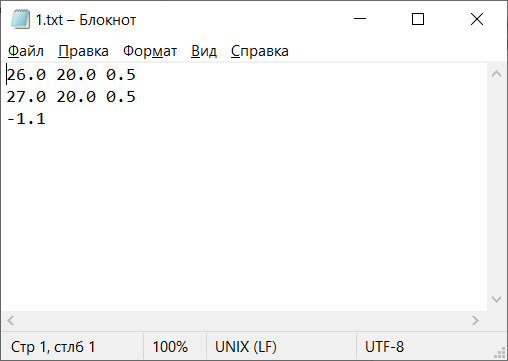
}

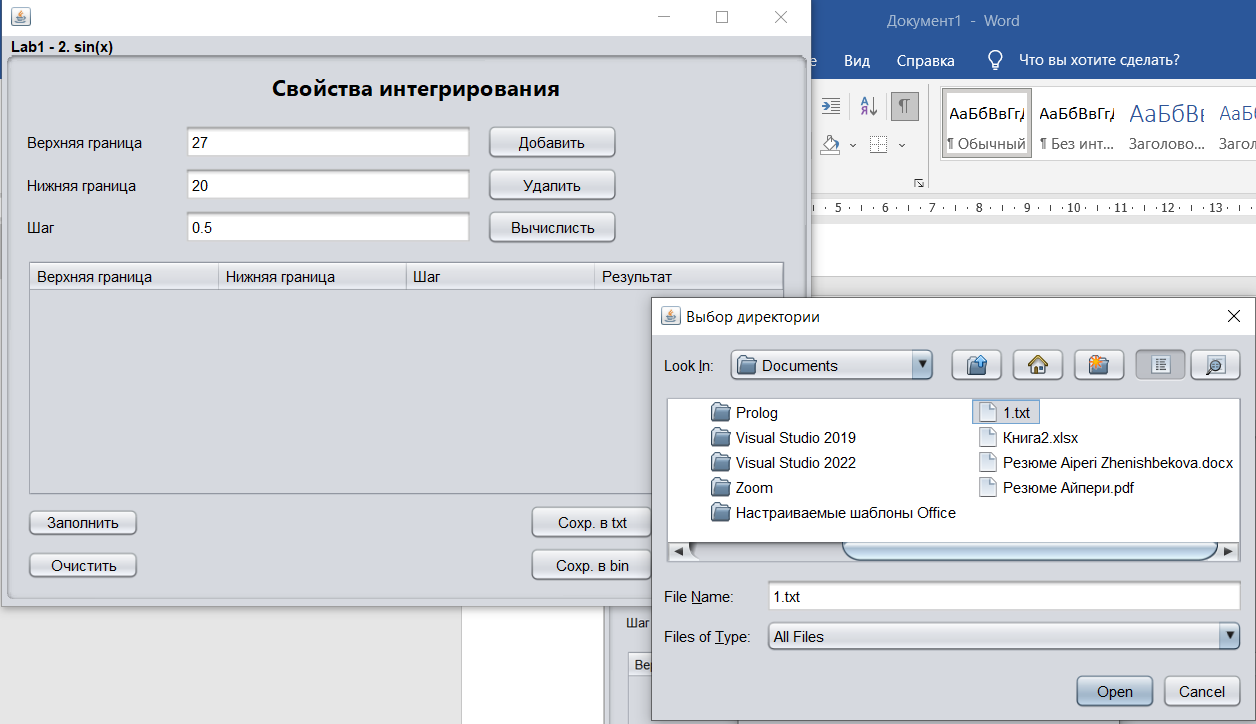
}

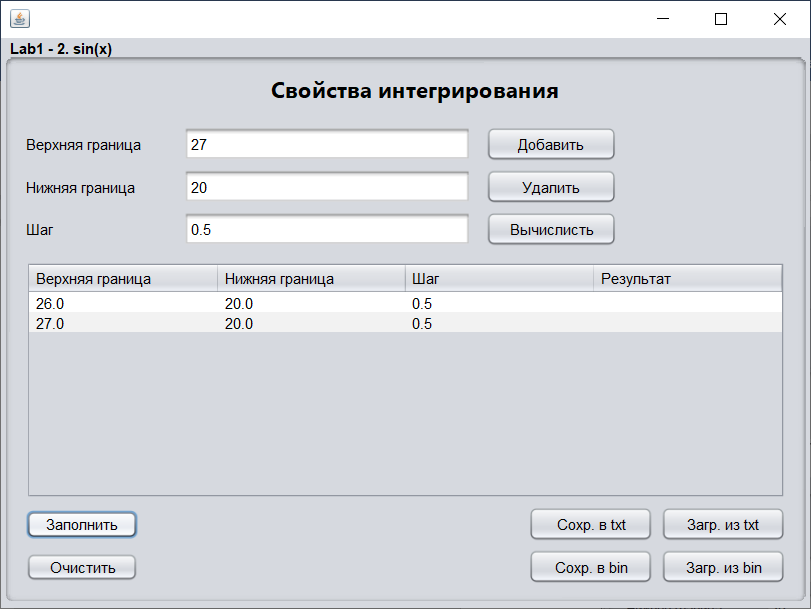
**Результаты выполнения работы:**

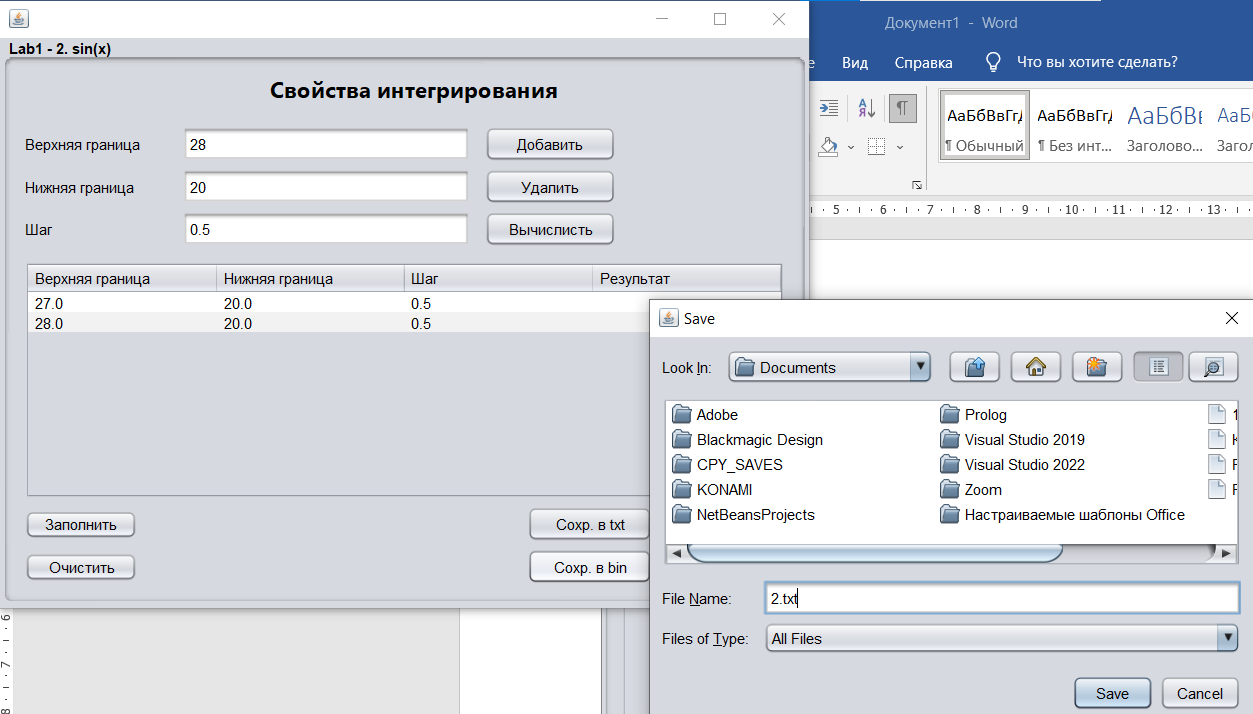


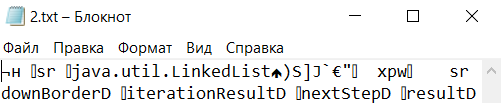


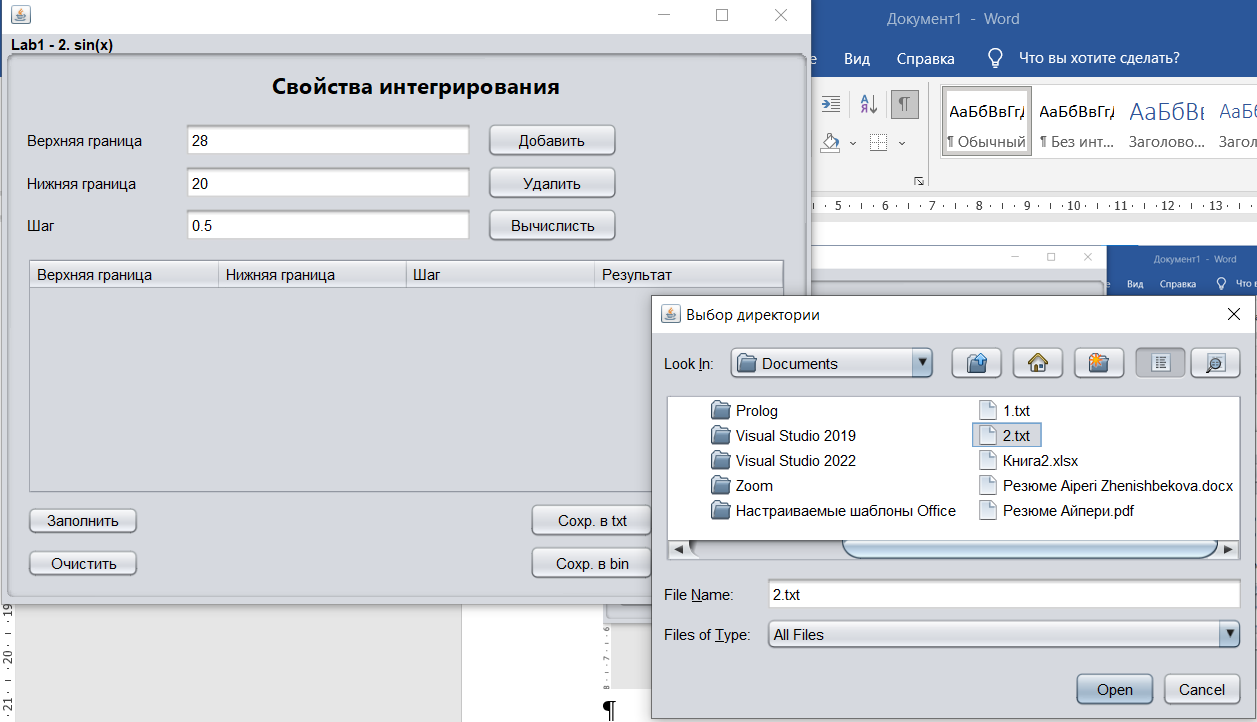


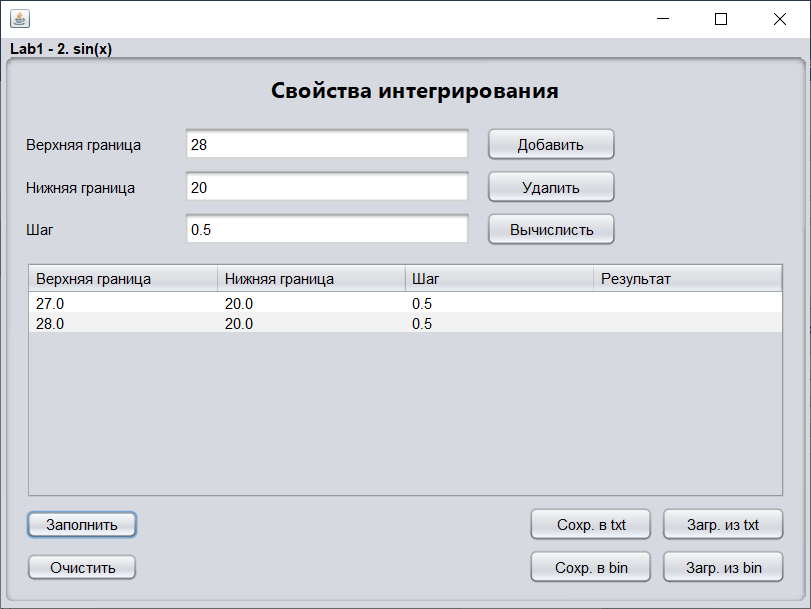




****

****

****



**Выводы:**

В процессе выполнения работы мы изучили работу с файлами и механизмы сериализации данных.