



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ

КАФЕДРА КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)

НАПРАВЛЕНИЕ ПОДГОТОВКИ 09.04.01 Информатика и вычислительная техника

МАГИСТЕРСКАЯ ПРОГРАММА 09.04.01/07 Интеллектуальные системы анализа,
обработки и интерпретации больших данных.

О Т Ч Е Т

по лабораторной работе № 6

Вариант 14

Название: Коллекции

Дисциплина: Языки программирования для работы с большими
данными

Студент

ИУ6-23М

(Группа)

(Подпись, дата)

В.Е. Санталов

(И.О. Фамилия)

Преподаватель

(Подпись, дата)

П.В. Степанов

(И.О. Фамилия)

Москва, 2022

Цель работы: получение навыков работы с коллекциями в Java.

Задание 1:

4. Сложить два многочлена заданной степени, если коэффициенты многочленов хранятся в объекте HashMap.
5. Умножить два многочлена заданной степени, если коэффициенты многочленов хранятся в различных списках.

Подзадача 1.

Код программы:

```
import java.util.ArrayList;
import java.util.Arrays;
import java.util.HashMap;

public class Laba6_1 {

    public static void printPoly(ArrayList<Double> list) {
        for (double val : list) {
            System.out.print(val + "x^" + (list.indexOf(val) + 1));
            if (list.indexOf(val) != list.size() - 1) System.out.print(s: " + ");
        }

        System.out.println();
    }

    Run | Debug
    public static void main(String[] args) throws Exception {
        ArrayList<Double> list1 = new ArrayList<Double>(Arrays.asList(...a: 1.5, 2.73, 4.89, 3.0, 9.0));
        ArrayList<Double> list2 = new ArrayList<Double>(Arrays.asList(...a: 3.1, 2.2, 5.11, 4.7, 1.0));

        HashMap<Integer, Double> values = new HashMap<Integer, Double>();

        System.out.println(x: "Исходные многочлены:");
        printPoly(list1);
        printPoly(list2);

        for (int i = 0; i < list1.size(); i++) {
            for (int j = 0; j < list2.size(); j++) {
                int key = i + j + 2;

                if (!values.keySet().contains(key)) {
                    values.put(key, value: 0.0);
                }

                Double val = values.get(key);
                values.put(key, (double) Math.round((val + list1.get(i) * list2.get(j)) * 100) / 100);
            }
        }

        System.out.println(x: "\nИтоговый многочлен:");
        values.forEach((key, val) -> {
            System.out.print(val + "x^" + key);
            if (key != list1.size() + list2.size()) System.out.print(s: " + ");
        });
    }
}
```

Подзадача 2.

Код программы:

```
import java.io.BufferedReader;
import java.io.File;
import java.io.FileInputStream;
import java.io.InputStreamReader;
import java.util.ArrayList;
import java.util.Collections;

public class Laba6_2 {
    Run | Debug
    public static void main(String[] args) {
        ArrayList<String> list = new ArrayList<String>();

        System.out.println();
        try (BufferedReader br = new BufferedReader(new InputStreamReader(new FileInputStream(new File(System.getProperty("user.dir") + "\\src\\file.txt"))),
            for (String line; (line = br.readLine()) != null;) {
                list.add(line);
            }
        ) catch (Exception e) {
            System.out.println(x: "File not found");
            System.exit(status: 0);
        }

        System.out.println("Before sort: " + list);
        Collections.sort(list);
        System.out.println("After sort " + list);
    }
}
```

Задание 2:

1. Во входном файле хранятся наименования некоторых объектов. Построить список С1, элементы которого содержат наименования и шифры данных объектов, причем элементы списка должны быть упорядочены по возрастанию шифров. Затем “сжать” список С1, удаляя дублирующие наименования объектов.
2. Во входном файле расположены два набора положительных чисел; между наборами стоит отрицательное число. Построить два списка С1 и С2, элементы которых содержат соответственно числа 1-го и 2-го набора таким образом, чтобы внутри одного списка числа были упорядочены по возрастанию. Затем объединить списки С1 и С2 в один упорядоченный список, изменяя только значения полей ссылочного типа.

Подзадача 1.

Код программы:

```

import java.util.Objects;

public class CustomObject {
    String name;
    int hash;

    public CustomObject(String name) {
        this.name = name;
        this.hash = Objects.hash(name);
    }

    @Override
    public int hashCode() {
        return Objects.hash(name, hash);
    }

    @Override
    public String toString() {
        return "{" +
            " name='" + this.name + "'" +
            ", hash='" + this.hash + "'" +
            "}";
    }
}

```

```

import java.util.ArrayList;
import java.util.Collections;
import java.util.Comparator;
import java.io.BufferedReader;
import java.io.File;
import java.io.FileInputStream;
import java.io.InputStreamReader;

public class Laba6_3 {

    Run | Debug
    public static void main(String[] args) throws Exception {
        ArrayList<CustomObject> list = new ArrayList<CustomObject>();

        System.out.println();
        try (BufferedReader br = new BufferedReader(new InputStreamReader(new FileInputStream(new File(System.getProperty("key: "user.dir") + "\\src\\objects.txt"))),
            for (String line; (line = br.readLine()) != null; ) {
                list.add(new CustomObject(line));
            }
        } catch (Exception e) {
            System.out.println(x: "File does not exists");
            System.exit(status: 0);
        }

        System.out.println("Начальный список: " + list + "\n");

        Collections.sort(list, new Comparator<CustomObject>() {
            public int compare(CustomObject o1, CustomObject o2) {
                if (o1.hash == o2.hash) return 0;
                return o1.hash < o2.hash ? -1 : 1;
            }
        });

        System.out.println("Список после сортировки: " + list + "\n");

        for (CustomObject obj : new ArrayList<CustomObject>(list)) if (list.indexOf(obj) != list.lastIndexOf(obj)) list.remove(obj);
        System.out.println("Список после удаления одинаковых записей: " + list);
    }
}

```

Подзадача 2.

Код программы:

```
Run | Debug
public static void main(String[] args) {
    LinkedList<Integer> list1 = new LinkedList<Integer>();
    LinkedList<Integer> list2 = new LinkedList<Integer>();

    System.out.println();
    try (BufferedReader br = new BufferedReader(new InputStreamReader(new FileInputStream(new File(System.getProperty("key: "user.dir") + "\\src\\numbers.txt"))),
        for (String line; (line = br.readLine()) != null;) {
            String[] strNum = line.split(regex; " ");

            Boolean gotNegative = false;

            for (String num : strNum) {
                int number = Integer.parseInt(num);

                if (number < 0) {
                    gotNegative = true;
                }
                else if (gotNegative) {
                    list2.add(number);
                }
                else {
                    list1.add(number);
                }
            }
        }
    } catch (Exception e) {
        System.out.println(x; "File does not exists");
        System.exit(status; 0);
    }

    System.out.println("Списки до сортировки:\nC1: " + list1 + "\nC2: " + list2);

    Collections.sort(list1);
    Collections.sort(list2);

    System.out.println("\nСписки после сортировки:\nC1: " + list1 + "\nC2: " + list2);

    list1.addAll(list2);
    System.out.println("\nИтоговый список: " + list1);
}
```

Ссылка на программное решение:

<https://github.com/Time2HackJS/BigDataLanguages/tree/master/lr6>

Вывод: в ходе лабораторной работы были получены навыки с коллекциями в Java.