

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

## Вариант 13

**Дисциплина:** Языки программирования для работы с большими данными

(Подпись, дата)

П.В. Степанов  
 (И.О. Фамилия)

Москва, 2022

**Цель лабораторной работы:** получение первичных навыков работы с коллекциями языка программирования Java.

## Ход работы:

### Задание №1:

1. С использованием множества выполнить попарное суммирование произвольного конечного ряда чисел по следующим правилам: на первом этапе суммируются попарно рядом стоящие числа, на втором этапе суммируются результаты первого этапа и т.д. до тех пор, пока не останется одно число.
2. Сложить два многочлена заданной степени, если коэффициенты многочленов хранятся в объекте HashMap.

### Листинг выполнения подзадачи 1:

```
package laba_6.task_1;
import java.util.Iterator;
import java.util.LinkedHashSet;
import java.util.Random;
//3. С использованием множества выполнить попарное суммирование произвольного
// конечного ряда чисел по следующим правилам: на первом этапе суммируются попарно
// рядом стоящие числа, на втором этапе суммируются результаты первого этапа и
// т.д. до тех пор, пока не останется одно число.

public class Row_numbers {
    public static void main(String[] args) {
        LinkedHashSet<Integer> set = new LinkedHashSet<>();

        Random random = new Random();
        int length = random.nextInt(1, 20);
        for (int i = 0; i < length; i++) {
            set.add(random.nextInt(1, 20));
        }

        System.out.println(set);
        Iterator<Integer> it = set.iterator();

        boolean iteration_done = false, global_done = false;
        int first, second;
        LinkedHashSet<Integer> new_set = new LinkedHashSet<>();

        while (!global_done) {
            new_set.clear();
            iteration_done = false;
            while (!iteration_done) {
                if (it.hasNext()) {
                    first = it.next();
                    if (it.hasNext()) {
                        second = it.next();
                        new_set.add(first + second);
                    }
                }
            }
            iteration_done = true;
            global_done = true;
        }
    }
}
```

```

        } else {
            new_set.add(first);
            iteration_done = true;
        }
    } else {
        iteration_done = true;
    }
}
set = new LinkedHashSet<>(new_set);
it = set.iterator();
System.out.println(set);
if(set.size() == 1) global_done = true;
}
}
}

```

### Результат выполнения подзадачи 1:

```

C:\Users\metel\.jdk\openjdk-18.0.1.1\bin\java.exe -Didea.config.path=C:\Users\metel\AppData\Local\JetBrains\Idea\config\ -Didea.home.path=C:\Program Files\JetBrains\Idea\ -Didea.platform.prefix=JDK Educational Edition 2021.3.2\lib\idea_2021.3.2\bin" -Dfile.encoding=UTF-8 C:\Users\metel\IdeaProjects\my_lab_6\src\main\java\Task1.java
[8, 19, 10, 3, 14, 2]
[27, 13, 16]
[40, 16]
[56]

```

### Листинг выполнения подзадачи 2:

```

package laba_6.task_1;
import java.util.HashMap;
import java.util.Random;
import java.util.Scanner;
//Сложить два многочлена заданной степени, если коэффициенты многочленов хранятся в
объекте HashMap.
public class Add_mnogochl {
    public static void main(String[] args) {

        HashMap<Integer, Integer> mnogochl_1 = new HashMap<>();
        HashMap<Integer, Integer> mnogochl_2 = new HashMap<>();

        Random random = new Random();
        Scanner scanner = new Scanner(System.in);
        System.out.print("Stepen mnogochlena = ");
        int stepen = scanner.nextInt();

        for (int i = 0; i < stepen; ++i) {
            mnogochl_1.put(i, random.nextInt(50));
            mnogochl_2.put(i, random.nextInt(50));
        }
        System.out.println(mnogochl_1.values());
        System.out.println(mnogochl_2.values());
    }
}

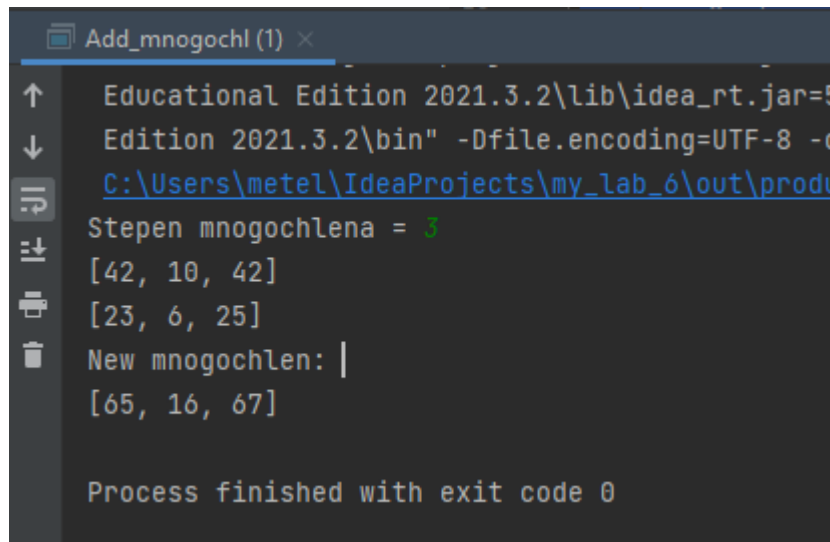
```

```

HashMap<Integer, Integer> mnogochl_3 = new HashMap<>();
for (int key : mnogochl_1.keySet()) {
    mnogochl_3.put(key, mnogochl_1.get(key) + mnogochl_2.get(key));
}
System.out.println("New mnogochlen: ");
System.out.println(mnogochl_3.values());
}
}

```

### Результат выполнения подзадачи 2:



```

Add_mnogochl (1) x
Educational Edition 2021.3.2\lib\idea_rt.jar=5
Edition 2021.3.2\bin" -Dfile.encoding=UTF-8 -c
C:\Users\metel\IdeaProjects\my_lab_6\out\produ
Stepen mnogochlena = 3
[42, 10, 42]
[23, 6, 25]
New mnogochlen: |
[65, 16, 67]

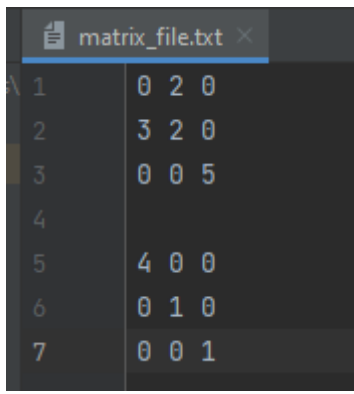
Process finished with exit code 0

```

### Задание №2:

1. Во входном файле хранятся две разреженные матрицы A и B. Построить циклически связанные списки SA и SB, содержащие ненулевые элементы соответственно матриц A и B. Просматривая списки, вычислить: а) сумму  $S = A + B$ ; б) произведение  $P = A * B$ .
2. Во входном файле хранятся наименования некоторых объектов. Построить список C1, элементы которого содержат наименования и шифры данных объектов, причем элементы списка должны быть упорядочены по возрастанию шифров. Затем “сжать” список C1, удаляя дублирующие наименования объектов.

### Входной файл:



|   |   |   |   |
|---|---|---|---|
| 1 | 0 | 2 | 0 |
| 2 | 3 | 2 | 0 |
| 3 | 0 | 0 | 5 |
| 4 |   |   |   |
| 5 | 4 | 0 | 0 |
| 6 | 0 | 1 | 0 |
| 7 | 0 | 0 | 1 |

### Листинг выполнения подзадачи 1:

#### Класс Elem:

```
package laba_6.task_2;

public class Elem {
    private int row;
    private int col;
    private int meaning;

    public Elem(int row, int col, int meaning) {
        this.row = row;
        this.col = col;
        this.meaning = meaning;
    }

    public int getRow() {
        return row;
    }

    public int getCol() {
        return col;
    }

    public int getMeaning() {
        return meaning;
    }

    @Override
    public String toString() {
        return "Element{" +
            "row=" + row +
            ", col=" + col +
            ", meaning=" + meaning +
            '}';
    }
}
```

#### Основная программа Lists\_m:

```
package laba_6.task_2.matrix;
import laba_6.task_2.matrix.Elem;

import java.io.File;
import java.io.FileNotFoundException;
import java.util.LinkedList;
import java.util.Scanner;
```

```

//3. Во входном файле хранятся две разреженные матрицы A и B.
// Построить циклически связанные списки CA и CB, содержащие ненулевые элементы
// соответственно матриц A и B. Просматривая списки, вычислить: а) сумму  $S = A + B$ ;
б) произведение  $P = A * B$ .
public class Lists_m {
    private LinkedList<Elem> list_A;
    private LinkedList<Elem> list_B;

    public Lists_m(File mtx_file){
        this.list_A = new LinkedList<>();
        this.list_B = new LinkedList<>();

        Scanner scanner = null;
        try {
            scanner = new Scanner(mtx_file);
        } catch (FileNotFoundException e) {
            e.printStackTrace();
        }

        boolean got_matrix = false;
        int row_number = 1;
        int col_number = 1;
        while(scanner.hasNextLine()) {
            String line = scanner.nextLine();
            String[] nums = line.split(" ");
            if (nums.length < 2) {
                got_matrix = true;
                row_number = 1;
            } else {
                col_number = 1;
                for (String num : nums) {
                    if(Integer.parseInt(num) != 0) {
                        if (!got_matrix) {
                            list_A.addLast(new Elem(row_number, col_number,
Integer.parseInt(num)));
                        } else {
                            list_B.addLast(new Elem(row_number, col_number,
Integer.parseInt(num)));
                        }
                    }
                    ++col_number;
                }
                ++row_number;
            }
        }
    }

    public LinkedList<Elem> getList_A(){
        return this.list_A;
    }
    public LinkedList<Elem> getList_B(){
        return this.list_B;
    }
}

```

## Результат выполнения подзадачи 1:

```
MatrixMain x
C:\Users\metel\.jdk\openjdk-18.0.1.1\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Educational Edition 2021.3.2\lib\idea_rt.jar=63115:C:\Program Files\JetBrains\IntelliJ IDEA Educational Edition 2021.3.2\bin" -Dfile.encoding=UTF-8 -classpath C:\Users\metel\IdeaProjects\my_lab_6\out\production\my_lab_6 laba_6.task_2.MatrixMain
[Element{row=1, col=2, meaning=2}, Element{row=2, col=1, meaning=3}, Element{row=2, col=2, meaning=2}, Element{row=3, col=3, meaning=5}]
[Element{row=1, col=1, meaning=4}, Element{row=2, col=2, meaning=1}, Element{row=3, col=3, meaning=1}]
SUM: [Element{row=1, col=1, meaning=4}, Element{row=1, col=2, meaning=2}, Element{row=2, col=1, meaning=3}, Element{row=2, col=2, meaning=3}, Element{row=3, col=3, meaning=6}]
MUL: [Element{row=1, col=2, meaning=2}, Element{row=2, col=1, meaning=12}, Element{row=2, col=2, meaning=2}, Element{row=3, col=3, meaning=5}]
```

## Листинг выполнения подзадачи 2:

### Класс MyObject:

```
package laba_6.task_2;
import java.util.Objects;

public class MyObject {
    String name;
    int hash;

    public MyObject(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 + "'" +
            "}";
    }
}
```

### Основная программа ObjMain:

```
package laba_6.task_2;
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 ObjMain {
    public static void main(String[] args) throws Exception {
        ArrayList<MyObject> list = new ArrayList<MyObject>();
    }
}
```

```

        System.out.println();
        try (BufferedReader br = new BufferedReader(new InputStreamReader(new
FileInputStream(new
File("C:\\Users\\metel\\IdeaProjects\\my_lab_6\\src\\laba_6\\task_2\\obj.txt")),
"UTF-8"))) {
            for (String line; (line = br.readLine()) != null;) {
                list.add(new MyObject(line));
            }
        } catch (Exception e) {
            System.out.println("File does not exists");
            System.exit(0);
        }

        System.out.println("Start List: " + list + "\n");

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

        System.out.println("End List: " + list + "\n");

        for (MyObject obj : new ArrayList<MyObject>(list)) if (list.indexOf(obj) !=
list.lastIndexOf(obj)) list.remove(obj);
        System.out.println("CList without similar records: " + list);
    }
}

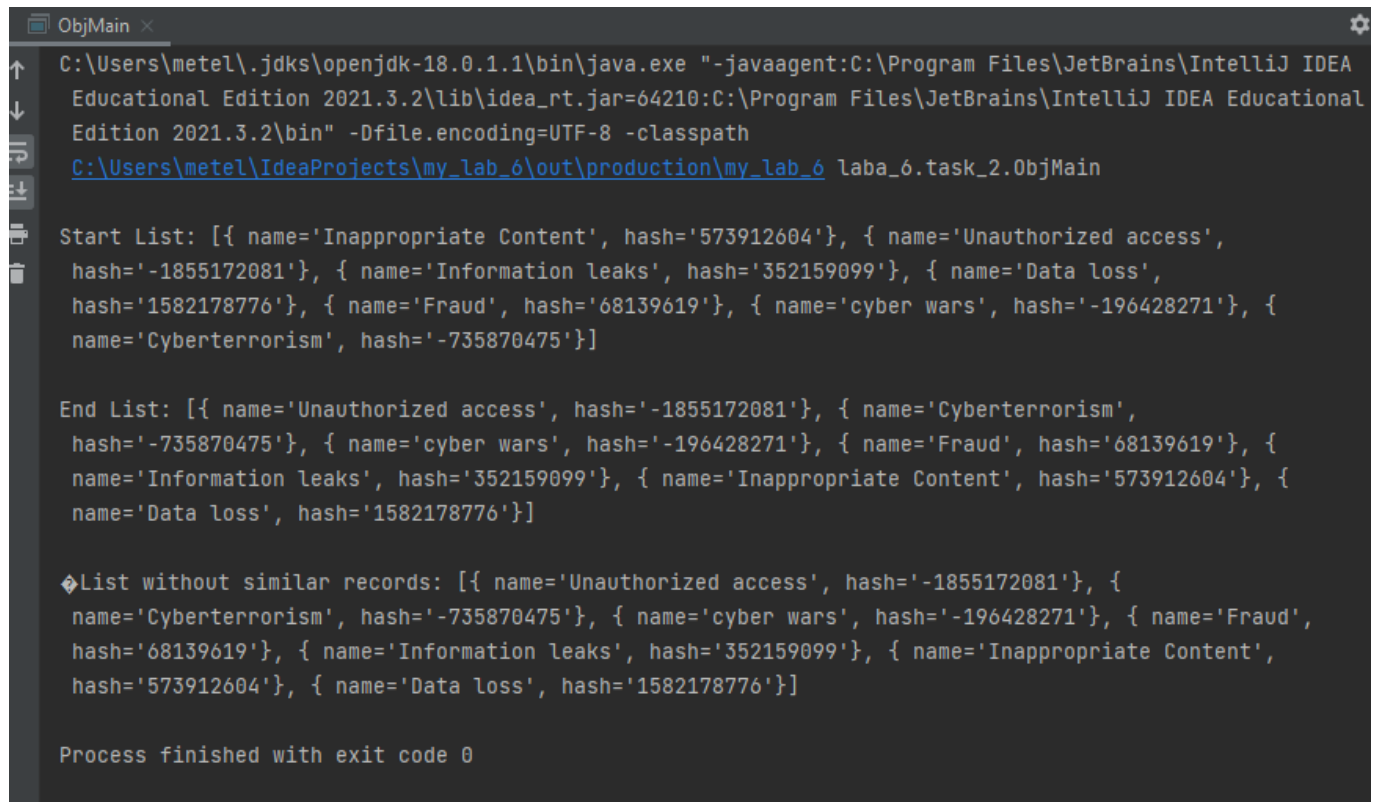
```

## Входной файл подзадачи 2:

| obj.txt |                       |
|---------|-----------------------|
| 1       | Inappropriate Content |
| 2       | Unauthorized access   |
| 3       | Information leaks     |
| 4       | Data loss             |
| 5       | Fraud                 |
| 6       | cyber wars            |
| 7       | Cyberterrorism        |



## Результат выполнения подзадачи 2:



```
ObjMain x
C:\Users\metel\.jdk\openjdk-18.0.1.1\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Educational Edition 2021.3.2\lib\idea_rt.jar=64210:C:\Program Files\JetBrains\IntelliJ IDEA Educational Edition 2021.3.2\bin" -Dfile.encoding=UTF-8 -classpath C:\Users\metel\IdeaProjects\my_lab_6\out\production\my_lab_6 laba_6.task_2.ObjMain

Start List: [{ name='Inappropriate Content', hash='573912604'}, { name='Unauthorized access', hash='-1855172081'}, { name='Information leaks', hash='352159099'}, { name='Data loss', hash='1582178776'}, { name='Fraud', hash='68139619'}, { name='cyber wars', hash='-196428271'}, { name='Cyberterrorism', hash='-735870475'}]

End List: [{ name='Unauthorized access', hash='-1855172081'}, { name='Cyberterrorism', hash='-735870475'}, { name='cyber wars', hash='-196428271'}, { name='Fraud', hash='68139619'}, { name='Information leaks', hash='352159099'}, { name='Inappropriate Content', hash='573912604'}, { name='Data loss', hash='1582178776'}]

◆List without similar records: [{ name='Unauthorized access', hash='-1855172081'}, { name='Cyberterrorism', hash='-735870475'}, { name='cyber wars', hash='-196428271'}, { name='Fraud', hash='68139619'}, { name='Information leaks', hash='352159099'}, { name='Inappropriate Content', hash='573912604'}, { name='Data loss', hash='1582178776'}]

Process finished with exit code 0
```

Программное решение представлено в репозитории распределённой системы управления версиями Git:

[https://github.com/matvilen/BigDataLanguages/tree/main/my\\_lab\\_6/src/laba\\_6](https://github.com/matvilen/BigDataLanguages/tree/main/my_lab_6/src/laba_6)

**Вывод:** при выполнении лабораторной работы были получены навыки работы с коллекциями в Java.