|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **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 **09.04.01/12 Интеллектуальный анализ больших**

**данных в системах поддержки принятия решений.**

**Отчет**

**по лабораторной работе № 5**

**Вариант № 13**

**Название:** коллекции

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

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студент | ИУ6-23М |  |  | С.П.Пантелеев |
|  | (Группа) |  | (Подпись, дата) | (И.О. Фамилия) |
|  |  |  |  |  |
| Преподаватель |  |  |  | П.В. Степанов |
|  |  |  | (Подпись, дата) | (И.О. Фамилия) |

Москва, 2023

**Цель:** изучить работу с коллекциями в java.

**Задание 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.

Код класса Main:

package Zadanie\_1;

import java.util.HashSet;

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner console = new Scanner(System.in);

HashSet<Integer> array = new HashSet<Integer>();

System.out.println("Сколько будет чисел: ");

int col = console.nextInt();

for (int i = 0; i < col; i++) {

array.add(console.nextInt());

}

int kol = -1;

while (kol != 1){

int i = 0;

int[] arr = new int[array.size()];

for (int ele:array){

arr[i++] = ele;

}

int kol\_array = array.size();

array.clear();

for (int j = 0; j <= kol\_array - 1; j += 2) {

if (j+1 == kol\_array){

array.add(arr[j]);

}else {

array.add(arr[j] + arr[j+1]);

}

}

System.out.println(array);

kol = array.size();

}

}

}

Работа программы показана на рисунке 1.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 1 – Работа программы

**Задание 2:** сложить два многочлена заданной степени, если коэффициенты многочленов хранятся в объекте HashMap.

Код класса Main:

package Zadanie\_2;

import java.util.HashMap;

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner console = new Scanner(System.in);

System.out.println("Степень многочлена: ");

int st = console.nextInt();

HashMap<Integer, Integer> hm1 = new HashMap<>();

HashMap<Integer, Integer> hm2 = new HashMap<>();

HashMap<Integer, Integer> hm3 = new HashMap<>();

hm1.put(1, 10);

hm1.put(2, 11);

hm1.put(3, 13);

hm1.put(4, 15);

hm2.put(1, 10);

hm2.put(2, 11);

hm2.put(3, 13);

hm2.put(4, 15);

for (int i = 1; i <= st; i++) {

hm3.put(i, hm1.get(i) + hm2.get(i));

}

System.out.println(hm3);

}

}

Работа программы показана на рисунке 2.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 2 – Работа программы

**Задание 3:** Во входном файле хранятся две разреженные матрицы А и В. Построить циклически связанные списки СА и СВ, содержащие ненулевые элементы соответственно матриц А и В. Просматривая списки, вычислить: а) сумму S = A + B; б) произведение P = A \* B.

Код класса Main:

package Zadanie\_3;

import java.io.FileNotFoundException;

import java.io.FileReader;

import java.io.IOException;

import java.util.HashMap;

public class Main {

public static void main(String[] args) {

HashMap<Integer, Integer> CA = new HashMap<>();

HashMap<Integer, Integer> CB = new HashMap<>();

char[] matrix = new char[54];

int[] matrix1 = new int[9];

int[] matrix2 = new int[9];

int kol = 0;

try(FileReader reader = new FileReader("C:/University/Java\_labs/laba\_6/Matrix.txt"))

{

int c;

while((c=reader.read())!=-1){

matrix[kol] = (char)c;

kol++;

//System.out.print((char)c);

}

} catch (IOException e) {

}

int first = 0;

boolean first\_check = false;

int t = 0;

int j = 0;

for (int i = 0; i < 27; i +=3) {

matrix1[j] = Integer.parseInt(String.valueOf(matrix[i]));

if (matrix1[j] > 0){

CA.put(matrix1[j], t);

t = matrix1[j];

if (!first\_check){

first = matrix1[j];

first\_check = true;

}

}

j++;

}

CA.replace(first, t);

System.out.println(CA);

first = 0;

first\_check = false;

t = 0;

j = 0;

for (int i = 0; i < 27; i +=3) {

matrix2[j] = Integer.parseInt(String.valueOf(matrix[i + 27]));

if (matrix2[j] > 0){

CB.put(matrix2[j], t);

t = matrix2[j];

if (!first\_check){

first = matrix2[j];

first\_check = true;

}

}

j++;

}

CB.replace(first, t);

System.out.println(CB);

int[] matrix\_S = new int[9];

int[] matrix\_P = new int[9];

System.out.println("Сумма матриц: ");

int st = 1;

for (int i = 0; i < 9; i++) {

matrix\_S[i] = matrix1[i] + matrix2[i];

System.out.print(matrix\_S[i] + " ");

if (st % 3 == 0 && i != 0){

System.out.println();

}

st++;

}

System.out.println("Умножение матриц: ");

st = 1;

for (int i = 0; i < 9; i++) {

matrix\_P[i] = matrix1[i] \* matrix2[i];

System.out.print(matrix\_S[i] + " ");

if (st % 3 == 0 && i != 0){

System.out.println();

}

st++;

}

}

}

Работа программы показана на рисунке 3.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 3 – Работа программы

**Задание 4:** 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.

Код класса Main:

package Zadanie\_4;

import java.io.FileReader;

import java.io.IOException;

import java.util.\*;

public class Main {

public static void main(String[] args) {

String[] objects = new String[9];

HashMap<Integer, String> C1 = new HashMap<>();

int kol = 0;

try(FileReader reader = new FileReader("C:/University/Java\_labs/laba\_6/Objects.txt"))

{

int c;

while((c=reader.read())!=-1){

objects[kol] = Character.toString((char)c);

kol++;

}

} catch (IOException e) {

}

int j = 3;

for (int i = 0; i < 9; i += 3) {

C1.put(j, objects[i]);

j--;

}

TreeMap<Integer, String> sorted\_C1 = new TreeMap<>();

sorted\_C1.putAll(C1);

System.out.println(C1);

Iterator<Integer> keyIter = C1.keySet().iterator();

HashMap<String, Integer> C2 = new HashMap<>();

while (keyIter.hasNext()) {

Integer key = keyIter.next();

String value = C1.get(key);

C2.put(value, key);

}

Iterator<Integer> keyIter2 = C2.values().iterator();

HashMap<Integer, String> C3 = new HashMap<>();

while (keyIter2.hasNext()) {

Integer key = keyIter2.next();

String value = C1.get(key);

C3.put(key, value);

}

System.out.println(C3);

}

}

Работа программы показана на рисунке 4.

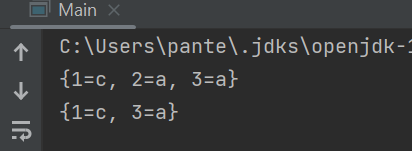


Рисунок 4 – Работа программы

**Вывод:** были изучена работа с коллекциями в java.