



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

О Т Ч Е Т

по лабораторной работе № 2

Название: Арифметические операции

Дисциплина: Языки программирования для работы с большими
данными

Студент

ИУ6-23М

(Группа)

(Подпись, дата)

Д.В. Авдонин

(И.О. Фамилия)

Преподаватель

(Подпись, дата)

(И.О. Фамилия)

Москва, 2022

Лабораторная работа № 2

Задание:

Вариант 1

В приведенных ниже заданиях необходимо вывести внизу фамилию разработчика, дату и время получения задания, а также дату и время сдачи задания. Для получения последней даты и времени следует использовать класс Date.

1. Ввести n строк с консоли, найти самую короткую и самую длинную строки. Вывести найденные строки и их длину.
2. Ввести n строк с консоли. Упорядочить и вывести строки в порядке возрастания (убывания) значений их длины.

Вариант 1

В приведенных ниже заданиях необходимо вывести внизу фамилию разработчика, дату и время получения задания, а также дату и время сдачи задания. Для получения последней даты и времени следует использовать класс Date.

1. Ввести n строк с консоли, найти самую короткую и самую длинную строки. Вывести найденные строки и их длину.
2. Ввести n строк с консоли. Упорядочить и вывести строки в порядке возрастания (убывания) значений их длины.

Ход работы:

Код программы:

```
package com.company.laboratorka2;

import java.util.Arrays;
import java.util.Collections;
import java.util.Date;
import java.util.Scanner;
import com.company.Library;

public class Main {
    private static void toSwap(int[] first, int[] second){
        for (int i = 0; i < first.length; i++) {
            int temp = first[i];
            first[i] = second[i];
            second[i] = temp;
        }
    }
    public static int[][] convert(int[][] b) {
        int[][] temp = new int[b[0].length][b.length];
        for (int i = 0; i < b.length; i++) {
            for (int j = 0; j < b[i].length; j++) {
                temp[i][j]=b[b[i].length-j-1][i];
            }
        }
        return temp;
    }
    private static void reverse(int[][] mas, int n){
        int tape;
        for (int i = 0; i < mas.length; i++) {
            int j = 0; // От начала
            int k = n; // До N
            while (j<k){
                tape = mas[i][j];
```

```

        mas[i][j] = mas[i][k];
        mas[i][k] = tape;
        ++j;
        --k;
    }
}
}
private static void reverse(int[][] mas, int n, int x){
    int tape;
    for (int i = 0; i < mas.length; i++) {
        int j = x; // От начала
        int k = n; // До N
        while (j < k){
            tape = mas[i][j];
            mas[i][j] = mas[i][k];
            mas[i][k] = tape;
            ++j;
            --k;
        }
    }
}
private static void shift(int[][] mas, int n, int k){
    reverse(mas, n-k-1); // Тут с 0 до K-1. Ниже с K до N
    reverse(mas, n-1, n-k); // Блин, ну тут перегрузка методов, иначе свою простую реализацию придется усложнять
    reverse(mas, n-1); // Тут с 0 до N.
}
public static void main(String[] args) {
    // Лабораторная работа 2 - Вариант 1 - Задание 1
    // В приведенных ниже заданиях необходимо вывести внизу фамилию разработчика, дату и время получения
    задания,
    // а также дату и время сдачи задания. Для получения последней даты и времени следует использовать класс
    Date.
    Date date = new Date();
    System.out.println("Фамилия разработчика: Авдонин. Дата получения 13.02.2022");
    System.out.println("Дата сдачи: " + date.toString());
    // Ввести n строк с консоли, найти самую короткую и самую длинную строки. Вывести найденные строки и их
    длину.
    System.out.println("Введите n строк через пробел, как надоеет нажмите enter \n");
    Scanner in = new Scanner(System.in);
    String str = in.nextLine().replace(" ", " ");
    String[] words = str.split(" ");
    String max_word = words[0];
    String min_word = words[0];
    for (String word : words) {
        if (max_word.length() < word.length())
            max_word = word;
        else if (min_word.length() > word.length())
            min_word = word;
    }
    System.out.println(max_word + " " + max_word.length());
    System.out.println(min_word + " " + min_word.length());
    // Лабораторная работа 2 - Вариант 1 - Задание 2
    // Ввести n строк с консоли. Упорядочить и вывести строки в порядке возрастания (убывания) значений их длины.
    Arrays.sort(words);
    for (String word : words) {
        System.out.print(word + " ");
    }
    System.out.println("- По возрастанию");
    Arrays.sort(words, Collections.reverseOrder());
    for (String word : words) {
        System.out.print(word + " ");
    }
    System.out.println("- По убыванию");
}

```

```

// Лабораторная работа 2 - Вариант 2 - Задание 1
// Упорядочить строки (столбцы) матрицы в порядке возрастания значений элементов k-го столбца (строки).
System.out.println("Введите n: ");
int n = in.nextInt();
int[][] mas_int = new int[n][n];
for (int i=0; i < n; i++){
    for (int j=0; j < n; j++){
        mas_int[i][j] = (int) Math.floor(Math.random() * (n - -n + 1)) + -n;
    }
}
//Вывожу матрицу
System.out.println("\n Матрица n*n: ");
Library.print_array(mas_int);
// for (int i = 0; i < mas_int.length; i++) {
//     Arrays.sort(mas_int[i]);
// }
// System.out.println("Отсортированная матрица: ");
// Library.print_array(mas_int);
System.out.println("Введите k-ый столбец по которому производить упорядочивание: ");
int k = in.nextInt();

for (int out = n - 1; out >= 1; out--){ //Внешний цикл
    for (int inn = 0; inn < out; inn++){ //Внутренний цикл
        if(mas_int[inn][k] > mas_int[inn + 1][k])
            toSwap(mas_int[inn], mas_int[inn + 1]); //вызвать метод, меняющий местами
    }
}
Library.print_array(mas_int);
System.out.println("Введите k-ую строку по которому производить упорядочивание: ");
int k_row = in.nextInt();
int swap_int = 0;
for (int x = 0; x < n; x++)
    for (int i = 0; i < n-1; i++) {
        if (mas_int[k_row][i] > mas_int[k_row][i + 1])
            for (int j = 0; j < n; j++) {
                swap_int = mas_int[j][i];
                mas_int[j][i] = mas_int[j][i + 1];
                mas_int[j][i + 1] = swap_int;
            }
    }
}
Library.print_array(mas_int);

// Лабораторная работа 2 - Вариант 2 - Задание 2
// Выполнить циклический сдвиг заданной матрицы на k позиций вправо (влево, вверх, вниз).
// Сдвиг вправо
System.out.println("Сдвиг вправо, введите k позиций сдвига:");
int k_shift = in.nextInt();
shift(mas_int, n, k_shift); // Тут происходит магия
Library.print_array(mas_int);

// Сдвиг влево
System.out.println("Сдвиг влево, введите k позиций сдвига:");
k_shift = in.nextInt();
mas_int = convert(mas_int); // А тут уличная магия
mas_int = convert(mas_int);
shift(mas_int, n, k_shift);
mas_int = convert(mas_int);
mas_int = convert(mas_int);
Library.print_array(mas_int);

// Сдвиг вверх
System.out.println("Сдвиг вверх, введите k позиций сдвига:");
k_shift = in.nextInt();

```

```
mas_int = convert(mas_int); // и тут уличная магия
shift(mas_int, n, k_shift);
mas_int = convert(mas_int);
mas_int = convert(mas_int);
mas_int = convert(mas_int);
Library.print_array(mas_int);

// Сдвиг вниз
System.out.println("Сдвиг вниз, введите k позиций сдвига:");
k_shift = in.nextInt();
mas_int = convert(mas_int); // все еще она
mas_int = convert(mas_int);
mas_int = convert(mas_int);
shift(mas_int, n, k_shift);
mas_int = convert(mas_int);
Library.print_array(mas_int);
in.close();
}
}
```

```

Фамилия разработчика: Авдонин. Дата получения 13.02.2022
Дата сдачи: Sun Jun 19 15:45:19 MSK 2022
Введите n строк через пробел, как надоест нажмите enter

1231 121 232
1231 4
121 3
121 1231 232 - По возрастанию
232 1231 121 - По убыванию
Введите n:
4

Матрица n*n:
0 2 -1 -1
-3 -4 -2 0
2 3 4 0
2 -1 -4 0
Введите k-ый столбец по которому производить упорядочивание:
1
-3 -4 -2 0
2 -1 -4 0
0 2 -1 -1
2 3 4 0
Введите k-ую строку по которому производить упорядочивание:
1
-2 -4 0 -3
-4 -1 0 2
-1 2 -1 0
4 3 0 2
Сдвиг вправо, введите k позиций сдвига:
2
0 -3 -2 -4
0 2 -4 -1
-1 0 -1 2
0 2 4 3
Сдвиг влево, введите k позиций сдвига:

```

Рисунок 1 – Результат работы программы

```
Сдвиг вправо, введите k позиций сдвига:
2
0 -3 -2 -4
0 2 -4 -1
-1 0 -1 2
0 2 4 3
Сдвиг влево, введите k позиций сдвига:
2
-2 -4 0 -3
-4 -1 0 2
-1 2 -1 0
4 3 0 2
Сдвиг вверх, введите k позиций сдвига:
1
-4 -1 0 2
-1 2 -1 0
4 3 0 2
-2 -4 0 -3
Сдвиг вниз, введите k позиций сдвига:
4
-4 -1 0 2
-1 2 -1 0
4 3 0 2
-2 -4 0 -3

Process finished with exit code 0
```

Рисунок 2 – Результат работы программы

Вывод: лабораторная работа выполнена в соответствии с заданием и вариантом.