



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, ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ И
СИСТЕМЫ УПРАВЛЕНИЯ

КАФЕДРА КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)

НАПРАВЛЕНИЕ ПОДГОТОВКИ 09.04.01 Интеллектуальные системы анализа,
обработки и интерпретации больших данных

ОТЧЕТ

по лабораторной работе № 2

Вариант 13

Название: Арифметические операции

Дисциплина: Языки программирования для работы с большими данными

Студент

ИУ6-22М

(Группа)

(Подпись, дата)

В.А.Ловцов

(И.О. Фамилия)

Преподаватель

(Подпись, дата)

П.В. Степанов

(И.О. Фамилия)

Москва, 2024

Цель: освоить принципы арифметических операций на языке программирования Java.

Задание 1: ввести n строк с консоли. Вывести на консоль те строки, длина которых меньше (больше) средней, а также длину.

Код класса Main:

```
import java.util.Scanner;

//TIP To <b>Run</b> code, press <shortcut actionId="Run"/> or
// click the <icon src="AllIcons.Actions.Execute"/> icon in the
gutter.
public class Third_1 {
    public static void main(String[] args) {
        //TIP Press <shortcut actionId="ShowIntentionActions"/>
with your caret at the highlighted text
        // to see how IntelliJ IDEA suggests fixing it.
        Scanner scanner = new Scanner(System.in);
        System.out.print("Введите число строк n: ");
        int length = scanner.nextInt();
        scanner.nextLine();
        String[] array = new String[length];
        System.out.println("Введите строки (через Enter):");
        int total_len = 0;
        for (int i = 0; i < length; i++) {
            System.out.print(i + 1 + " " );
            array[i] = scanner.nextLine();
            total_len = total_len + array[i].length();
        }

        float mean_len = (float) total_len / length;

        System.out.println("Средняя длина строки:" + mean_len);
        System.out.println("Строки длиннее средней:");
        for (String elem : array) {
            if (elem.length() > mean_len) {
                System.out.println("\t\"" + elem + "\"" (" +
elem.length() + " chars)" );
            }
        }
        System.out.println("\nСтроки короче средней:");
        for (String elem : array) {
            if (elem.length() < mean_len) {
                System.out.println("\t\"" + elem + "\"" (" +
elem.length() + " chars)" );
            }
        }
    }
}
```

Работа программы показана на рисунке 1.

```
/Library/Java/JavaVirtualMachines/jdk-21
Введите число строк n: 5
Введите строки (через Enter):
1) a a a
2) aaaaaaaaaaaaaa
3) sdjgdjksbfjkdhfjshdj
4) aa
5)
Средняя длина строки:8.0
Строки длиннее средней:
    "aaaaaaaaaaaaa" (13 chars)
    "sdjgdjksbfjkdhfjshdj" (20 chars)

Строки короче средней:
    "a a a" (5 chars)
    "aa" (2 chars)
    "" (0 chars)

Process finished with exit code 0
```

Рисунок 1 – Работа программы

Задание 2: ввести n слов с консоли. Найти слово, в котором число различных символов минимально. Если таких слов несколько, найти первое из них.

Код класса Main:

```
import java.util.Scanner;

class UniqChars {
    public static int count(String word) {
        int num_char = 0;
        StringBuilder charset = new StringBuilder();
        for (int i = 0; i < word.length(); i++) {
            if (charset.toString().indexOf(word.charAt(i)) == -
1) {
                charset.append(word.charAt(i));
                num_char++;
            }
        }

        return num_char;
    }
}
```

```

public class Fourth_1 {
    public static void main(String[] args) {
        //TIP Press <shortcut actionId="ShowIntentionActions"/>
        with your caret at the highlighted text
        // to see how IntelliJ IDEA suggests fixing it.
        Scanner scanner = new Scanner(System.in);
        System.out.print("Введите число слов n: ");
        int length = scanner.nextInt();
        scanner.nextLine();
        String[] array = new String[length];
        System.out.println("Введите слова (через пробел):");
        int min_chars = 999999999;
        String min_word = "";
        for (int i = 0; i < length; i++) {
            array[i] = scanner.next();
            int cur_chars_num = UniqChars.count(array[i]);
            if (cur_chars_num < min_chars) {
                min_chars = cur_chars_num;
                min_word = array[i];
            }
        }
        System.out.println("Слово \"" + min_word + "\" имеет
        наименьшее число различных символов: " + min_chars);
    }
}

```

Работа программы показана на рисунке 2.

```

/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents/Home/bin/
Введите число слов n: 6
Введите слова (через пробел):
abcde aaaaaa a abewjakbfjdbrelwqfh aabbcc djkbv
Слово "aaaaaa" имеет наименьшее число различных символов: 1

Process finished with exit code 0

```

Рисунок 2 – Работа программы

Задание 3: найти и вывести наибольшее число возрастающих (убывающих) элементов матрицы, идущих подряд.

Код класса Main:

```

import java.util.Arrays;
import java.util.Scanner;

public class Third_2 {
    public static void main(String[] args) {

```

```

Scanner scanner = new Scanner(System.in);
System.out.print("Введите n: ");
int n = scanner.nextInt();
int[][] arr = new int[n][n];

for (int i = 0; i < n; i++) {
    for (int j = 0; j < n; j++) {
        arr[i][j] = (int) (Math.random() * 2 * n - n);
        System.out.print(arr[i][j] + "\t");
    }
    System.out.println();
}

int max_seq = 1;
int cur_seq = 1;
for (int i = 0; i < n; i++) {
    for (int j = 0; j < n; j++) {

        if (j == 0 & i > 0) {
            if (arr[i][j] > arr[i - 1][n - 1]) {
                cur_seq++;
            }
            else {
                if (cur_seq > max_seq) {
                    max_seq = cur_seq;

                    System.out.println(i + " " + j);
                }
                cur_seq = 1;
            }
        } else if (j > 0) {
            if (arr[i][j] > arr[i][j - 1])
                cur_seq++;
            else {
                if (cur_seq > max_seq) {
                    max_seq = cur_seq;
                    System.out.println(i + " " + j);
                }
                cur_seq = 1;
            }
        }
    }
}

System.out.println("max " + max_seq);
}
}

```

Работа программы показана на рисунке 3.

```
/Library/Java/JavaVirtualMachines/jdk-21.jdk/Content
Введите n: 6
-3 -5 0 0 5 -5
5 5 -2 -5 1 -2
-2 -1 -3 0 -5 -4
1 -1 3 0 -4 -5
-5 0 -2 -5 -3 2
2 0 -3 -2 1 0
max 3

Process finished with exit code 0
```

Рисунок 3 – Работа программы

Задание 4: найти сумму элементов матрицы, расположенных между первым и вторым положительными элементами каждой строки.

Код класса Main:

```
import java.util.Scanner;

public class Fourth_2 {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Введите n: ");
        int n = scanner.nextInt();
        int[][] arr = new int[n][n];

        for (int i = 0; i < n; i++) {
            for (int j = 0; j < n; j++) {
                arr[i][j] = (int) (Math.random() * 2 * n - n);
                System.out.print(arr[i][j] + "\t");
            }
            System.out.println();
        }

        int summ = 0;
        for (int i = 0; i < n; i++) {
            int count_pos = 0;
            int cur_summ = 0;
            boolean count = false;
            for (int j = 0; j < n; j++) {
                if (arr[i][j] > 0) {
                    count_pos += 1;
                    if (count_pos <= 2)
                        count = !count;
                    if (count_pos == 2)
                        summ += cur_summ;
                }
            }
            System.out.println(cur_summ);
        }
    }
}
```

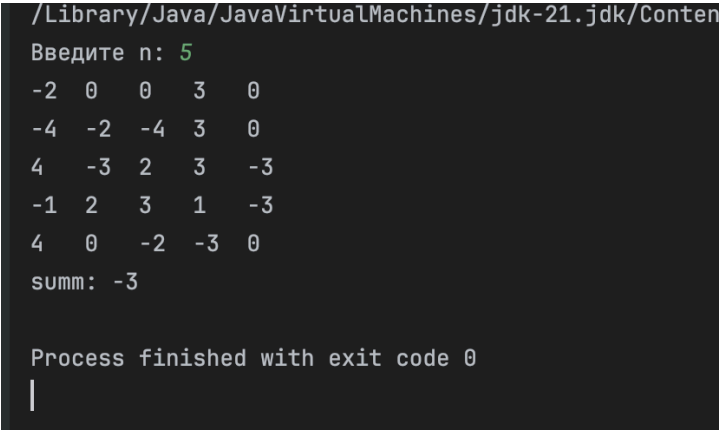
```

        }
        else if (count) {
            cur_summ += arr[i][j];
            System.out.println(arr[i][j]);
        }
    }

    System.out.println("summ: " + summ);
}
}

```

Работа программы показана на рисунке 4.



```

/Library/Java/JavaVirtualMachines/jdk-21.jdk/Content
Введите n: 5
-2 0 0 3 0
-4 -2 -4 3 0
4 -3 2 3 -3
-1 2 3 1 -3
4 0 -2 -3 0
summ: -3

Process finished with exit code 0
|

```

Рисунок 4 – Работа программы

Вывод: были освоены принципы арифметических операций на языке программирования Java.