



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.

О Т Ч Е Т
по лабораторной работе № 2
Вариант № 11

Название: Арифметические операции

Дисциплина: Языки программирования для работы с большими данными

Студент

ИУ6-23М

(Группа)

(Подпись, дата)

С.В.Мельников

(И.О. Фамилия)

Преподаватель

(Подпись, дата)

П.В. Степанов

(И.О. Фамилия)

Москва, 2023

Цель работы

Изучить арифметических операций на языке программирования Java.

Задание 1 (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.

Ввести n строк с консоли, найти самую короткую и самую длинную строки. Вывести найденные строки и их длину.

Листинг программы:

Код класса StringLength:

```
public class StringLength {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter count of strings: ");
        int n = scanner.nextInt();
        scanner.nextLine();
        int min_length = -1;
        int max_length = -1;
        String max_str = "";
        String min_str = "";
        System.out.println("Enter " + n + " strings: ");
        for (int i = 0; i < n; i++) {
            String str = scanner.nextLine();
            if (min_length == -1 || min_length > str.length()) {
                min_length = str.length();
                min_str = str;
            }
            if (max_length == -1 || max_length < str.length()) {
                max_length = str.length();
                max_str = str;
            }
        }
        System.out.println("The largest string: " + max_str + " size: " +
max_length);
        System.out.println("The smallest string: " + min_str + " size: " +
min_length);
        String developer = "Melnikov";
        Date dateOfGettingTask = new Date(2023 - 1900, Calendar.FEBRUARY, 17, 15,
40);
        Date dateOfPassingTask = new Date(2023 - 1900, Calendar.MARCH, 3, 15, 40);
        System.out.println("Developer: " + developer);
        System.out.println("Task got: " + dateOfGettingTask);
        System.out.println("Task passed: " + dateOfPassingTask);
    }
}
```

Работа программы представлена на рисунке 1.



```
Run: StringLength x
C:\Users\suslik13\jdk\openjdk-19.0.2\bin\java.exe "-javaagent:D:\JetBrains\IntelliJ IDEA 2022.3.2\lib\idea_rt.jar"
Enter count of strings: 9
Enter 9 strings:
Diving is a popular water sport
Recreational diving involves using scuba diving equipment
Diving offers the opportunity to explore the underwater world
There are many different types of diving
Kayaking is a physically demanding sport
Kayaking is a popular water sport
To participate in kayaking, you will need a kayak and appropriate clothing and gear
There are several different types of kayaking
Sea kayaking involves paddling in the ocean
The largest string: To participate in kayaking, you will need a kayak and appropriate clothing and gear size: 83
The smallest string: Diving is a popular water sport size: 31
Developer: Melnikov
Task got: Fri Feb 17 15:40:00 MSK 2023
Task passed: Fri Mar 03 15:40:00 MSK 2023

Process finished with exit code 0
```

Рисунок 1 – Работа программы StringLength.java

Задание 2 (Вариант 1, Задание 2):

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.

Ввести n строк с консоли. Упорядочить и вывести строки в порядке возрастания (убывания) значений их длины.

Листинг программы:

Код класса StringSort:

```
public class StringSort {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter count of strings: ");
        int n = scanner.nextInt();
        scanner.nextLine();
        ArrayList<String> array = new ArrayList<>();
        System.out.println("Enter " + n + " strings: ");
        for (int i = 0; i < n; i++) {
            String str = scanner.nextLine();
            array.add(str);
        }
        System.out.print("Enter 1 to sort in descending order and 2 to sort in ascending order: ");
        int rule = scanner.nextInt();
        if (rule == 1) {
            array.sort((s1, s2) -> s2.length() - s1.length());
        } else {
            array.sort((s1, s2) -> s1.length() - s2.length());
        }
    }
}
```

```

        array.sort(Comparator.comparingInt(String::length));
    }
    System.out.println("Sorted strings:");
    for (int i = 0; i < n; i++) {
        System.out.println(array.get(i));
    }
    String developer = "Melnikov";
    Date dateOfGettingTask = new Date(2023 - 1900, Calendar.FEBRUARY, 17, 15,
40);
    Date dateOfPassingTask = new Date(2023 - 1900, Calendar.MARCH, 3, 15, 40);
    System.out.println("Developer: " + developer);
    System.out.println("Task got: " + dateOfGettingTask);
    System.out.println("Task passed: " + dateOfPassingTask);
}
}

```

Работа программы представлена на рисунке 2.

```

Run: StringSort x
C:\Users\suslik13\.jdk\openjdk-19.0.2\bin\java.exe "-javaagent:D:\JetBrains\IntelliJ
Enter count of strings: 9
Enter 9 strings:
Diving is a popular water sport
Recreational diving involves using scuba diving equipment
Diving offers the opportunity to explore the underwater world
There are many different types of diving
Kayaking is a physically demanding sport
Kayaking is a popular water sport
To participate in kayaking, you will need a kayak and appropriate clothing and gear
There are several different types of kayaking
Sea kayaking involves paddling in the ocean
Enter 1 to sort in descending order and 2 to sort in ascending order: 2
Sorted strings:
Diving is a popular water sport
Kayaking is a popular water sport
There are many different types of diving
Kayaking is a physically demanding sport
Sea kayaking involves paddling in the ocean
There are several different types of kayaking
Recreational diving involves using scuba diving equipment
Diving offers the opportunity to explore the underwater world
To participate in kayaking, you will need a kayak and appropriate clothing and gear
Developer: Melnikov
Task got: Fri Feb 17 15:40:00 MSK 2023
Task passed: Fri Mar 03 15:40:00 MSK 2023

Process finished with exit code 0

```

Рисунок 2 – Работа программы StringSort.java

Задание 3 (Вариант 2, Задание 1):

Ввести с консоли n – размерность матрицы $a[n][n]$. Задать значения элементов матрицы в интервале значений от $-n$ до n с помощью датчика случайных чисел.

Упорядочить строки (столбцы) матрицы в порядке возрастания значений элементов k -го столбца (строки).

Листинг программы:

Код класса OrderByK:

```
public class OrderByK {

    public static void printMatrix(int[][] matrix) {
        for(int[] row: matrix) {
            for (int element: row) {
                System.out.printf("%4d", element);
            }
            System.out.println();
        }
    }

    public static void sortMatrixByKRow(int[][] matrix, int k) {
        for (int i = 0; i < matrix.length - 1; i++) {
            for (int j = 0; j < matrix.length - i - 1; j++) {
                if (matrix[k][j + 1] < matrix[k][j]) {
                    for (int t = 0; t < matrix.length; t++) {
                        int tmp_element = matrix[t][j + 1];
                        matrix[t][j + 1] = matrix[t][j];
                        matrix[t][j] = tmp_element;
                    }
                }
            }
        }
    }

    public static void sortMatrixByKColumn(int[][] matrix, int k) {
        for (int i = 0; i < matrix.length - 1; i++) {
            for (int j = 0; j < matrix.length - i - 1; j++) {
                if (matrix[j + 1][k] < matrix[j][k]) {
                    for (int t = 0; t < matrix.length; t++) {
                        int tmp_element = matrix[j + 1][t];
                        matrix[j + 1][t] = matrix[j][t];
                        matrix[j][t] = tmp_element;
                    }
                }
            }
        }
    }

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter size of matrix: ");
        int n = scanner.nextInt();
        int[][] matrix = new int[n][n];
        for(int i = 0; i < n; i++) {
            for (int j = 0; j < n; j++) {
```

```

        int element = (int) (Math.random() * (2 * n + 1)) - n;
        matrix[i][j] = element;
    }
}
printMatrix(matrix);
System.out.print("Enter 1 to sort by row or 2 to sort by column: ");
int rule = scanner.nextInt();
if (rule == 1) {
    System.out.print("Enter row number (1 - " + n + "): ");
} else {
    System.out.print("Enter column number (1 - " + n + "): ");
}
int k = scanner.nextInt() - 1;
if (rule == 1) {
    sortMatrixByKRow(matrix, k);
} else {
    sortMatrixByKColumn(matrix, k);
}
System.out.println("Sorted matrix:");
printMatrix(matrix);
}
}

```

Работа программы представлена на рисунке 3.

```

Run: OrderByK x
C:\Users\suslik13\.jdk\openjdk-19.0.2\bin\java.exe "-javaagent
Enter size of matrix: 4
-2 -4 -3 -1
4 -4 4 4
-2 -1 -1 4
0 0 4 1
Enter 1 to sort by row or 2 to sort by column: 1
Enter row number (1 - 4): 1
Sorted matrix:
-4 -3 -2 -1
-4 4 4 4
-1 -1 -2 4
0 4 0 1
Process finished with exit code 0

```

Рисунок 3 – Работа программы OrderByK.java

Задание 4 (Вариант 2, Задание 2):

Ввести с консоли n – размерность матрицы $a[n][n]$. Задать значения элементов матрицы в интервале значений от $-n$ до n с помощью датчика случайных чисел.

Выполнить циклический сдвиг заданной матрицы на k позиций вправо (влево, вверх, вниз).

Листинг программы:

Код класса CyclicShift:

```
enum Direction { UP, DOWN, LEFT, RIGHT }

public class CyclicShift {

    public static void printMatrix(int[][] matrix) {
        for(int[] row: matrix) {
            for (int element: row) {
                System.out.printf("%4d", element);
            }
            System.out.println();
        }
    }

    public static void matrixShift(int[][] matrix, int k, Direction direction) {
        for (int t = 0; t < k; t++) {
            switch (direction) {
                case UP -> {
                    for (int j = 0; j < matrix.length; j++) {
                        int first_element = matrix[0][j];
                        for (int i = 0; i < matrix.length - 1; i++) {
                            matrix[i][j] = matrix[i + 1][j];
                        }
                        matrix[matrix.length - 1][j] = first_element;
                    }
                }
                case DOWN -> {
                    for (int j = 0; j < matrix.length; j++) {
                        int last_element = matrix[matrix.length - 1][j];
                        for (int i = 0; i < matrix.length - 1; i++) {
                            matrix[i + 1][j] = matrix[i][j];
                        }
                        matrix[0][j] = last_element;
                    }
                }
                case LEFT -> {
                    for (int i = 0; i < matrix.length; i++) {
                        int first_element = matrix[i][0];
                        for (int j = 0; j < matrix.length - 1; j++) {
                            matrix[i][j] = matrix[i][j + 1];
                        }
                        matrix[i][matrix.length - 1] = first_element;
                    }
                }
                case RIGHT -> {
                    for (int i = 0; i < matrix.length; i++) {
                        int last_element = matrix[i][matrix.length - 1];
                        for (int j = 0; j < matrix.length - 1; j++) {
                            matrix[i][j + 1] = matrix[i][j];
                        }
                        matrix[i][0] = last_element;
                    }
                }
            }
        }
    }
}
```

```

public static void main(String[] args) {
    Scanner scanner = new Scanner(System.in);
    System.out.print("Enter size of matrix: ");
    int n = scanner.nextInt();
    int[][] matrix = new int[n][n];
    for(int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            int element = (int)(Math.random() * (2 * n + 1)) - n;
            matrix[i][j] = element;
        }
    }
    printMatrix(matrix);
    System.out.print("Enter a value for shift: ");
    int k = scanner.nextInt();
    System.out.print("Enter direction (up, down, left, right): ");
    scanner.nextLine();
    String direction = scanner.nextLine();
    switch (direction) {
        case "up" -> matrixShift(matrix, k, Direction.UP);
        case "down" -> matrixShift(matrix, k, Direction.DOWN);
        case "left" -> matrixShift(matrix, k, Direction.LEFT);
        case "right" -> matrixShift(matrix, k, Direction.RIGHT);
    }
    System.out.println("Shifted matrix:");
    printMatrix(matrix);
}
}

```

Работа программы представлена на рисунке 4.

```

Run: CyclicShift x
C:\Users\suslik13\jdk\openjdk-19.0.2\bin\java.exe "-javaagent
Enter size of matrix: 5
0 -4 5 4 4
3 3 4 0 5
-2 -1 -1 -3 4
5 -5 2 5 2
3 1 -1 3 -3
Enter a value for shift: 3
Enter direction (up, down, left, right): up
Shifted matrix:
5 -5 2 5 2
3 1 -1 3 -3
0 -4 5 4 4
3 3 4 0 5
-2 -1 -1 -3 4
Process finished with exit code 0

```

Рисунок 4 – Работа программы CyclicShift.java

Вывод

В ходе выполнения лабораторной работы были изучены арифметические операции на языке программирования Java.