



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 _____ «Теоретическая информатика и компьютерные технологии»

Лабораторная работа № 2
по курсу «Языки и методы программирования»
«Разработка простейшего класса на языке Java»

Студент группы ИУ9-21Б Яннаев А. С.

Преподаватель Посевин Д. П.

Москва 2025

1 Задание

1. Выполнение лабораторной работы заключается в составлении на языке Java одного из классов, приведённых в таблице. В классе обязательно должны присутствовать конструктор и метод toString.
2. Отладку разработанного класса необходимо осуществить в методе main вспомогательного класса Test. Использование контейнерных классов из стандартной библиотеки языка Java не разрешается.
3. Класс стреловидных матриц размера $n \times n$ с операцией вычисления определителя. Все элементы стреловидной матрицы, кроме принадлежащих первой строке, первому столбцу или главной диагонали, равны нулю. Матрица должна быть представлена в виде, исключающем хранение заведомо нулевых элементов.

2 Результаты

Исходный код программы представлен в 1, 2. Результат запуска представлен на рисунке 1, 2.

Листинг 1 — Файл ArrowMatrix.java

```

1 public class ArrowMatrix {
2     private int n;
3     private double [][] matrix;
4     public ArrowMatrix(int n, double [] firstRow, double [] firstColumn,
5         double [] mainDiagonal) {
6         this.n = n;
7         this.matrix = new double [n][n];
8         for (int i = 0; i < n; i++) {
9             matrix[0][i] = firstRow[i];
10            matrix[i][0] = firstColumn[i];
11            if (i > 0) { matrix[i][i] = mainDiagonal[i]; }
12        }
13    public double determinant() {
14        return computeDeterminant(matrix, n);}
15    private double computeDeterminant(double [][] mat, int size) {
16        if (size == 1) { return mat[0][0]; }
17        if (size == 2) { return mat[0][0] * mat[1][1] - mat[0][1] * mat
18[1][0]; }
19        double det = 0;
20        for (int i = 0; i < size; i++) {
21            double [][] subMatrix = getSubMatrix(mat, size, i);
22            double cofactor = (i % 2 == 0 ? 1 : -1) * mat[0][i] *
23computeDeterminant(subMatrix, size - 1);
24            det += cofactor;
25        }
26        return det;
27    }
28    private double [][] getSubMatrix(double [][] mat, int size, int column
29) {
30        double [][] subMatrix = new double [size - 1][size - 1];
31        for (int i = 1; i < size; i++) {
32            int subCol = 0;
33            for (int j = 0; j < size; j++) {
34                if (j == column) continue;
35                subMatrix[i - 1][subCol] = mat[i][j];
36                subCol++;
37            }
38        }
39        return subMatrix;
40    }
41    public String toString() {
42        StringBuilder result = new StringBuilder();
43        for (int i = 0; i < n; i++) {
44            for (int j = 0; j < n; j++) {
45                result.append(String.format("%10.1f ", matrix[i][j]));
46            }
47            result.append("\n");
48        }
49        return result.toString();
50    }
51 }

```

Листинг 2 — Файл Test.java

```

1 import java.util.Scanner;
2 public class Test {
3     public static void main(String[] args) {
4         Scanner scanner = new Scanner(System.in);
5         int n;
6         while (true) {
7             System.out.print("size (n): ");
8             n = scanner.nextInt();
9             if (n > 0) break;
10            System.out.println("size > 0.");
11        }
12        double[] firstRow = new double[n];
13        double[] firstColumn = new double[n];
14        double[] mainDiagonal = new double[n];
15        while (true) {
16            System.out.println("\n1st row:");
17            for (int i = 0; i < n; i++) {
18                System.out.print("firstRow[" + i + "]: ");
19                firstRow[i] = scanner.nextDouble();
20            }
21            System.out.println("\n1st column:");
22            for (int i = 0; i < n; i++) {
23                System.out.print("firstColumn[" + i + "]: ");
24                firstColumn[i] = scanner.nextDouble();
25            }
26            System.out.println("\nmain diag:");
27            for (int i = 1; i < n; i++) {
28                System.out.print("mainDiagonal[" + i + "]: ");
29                mainDiagonal[i] = scanner.nextDouble();
30            }
31            ArrowMatrix matrix = new ArrowMatrix(n, firstRow,
firstColumn, mainDiagonal);
32            System.out.println("\nmatrix:");
33            System.out.println(matrix);
34            System.out.print("right?: ");
35            String answer = scanner.next();
36            if (answer.equals("yes")) {
37                System.out.println("\ndet: " + matrix.determinant());
38                break;
39            }
40            System.out.println("try again");
41        }
42        scanner.close();
43    }
44 }

```

```
→ src java Test.java
введите размер матрицы (n): 3

введите первую строку матрицы:
firstRow[0]: 1
firstRow[1]: 2
firstRow[2]: 3

введите первый столбец матрицы:
firstColumn[0]: 4
firstColumn[1]: 5
firstColumn[2]: 6

введите главную диагональ матрицы:
mainDiagonal[1]: 7
mainDiagonal[2]: 8

введённая матрица:
      4,0      2,0      3,0
      5,0      7,0      0,0
      6,0      0,0      8,0

правильно? (да/нет): да

определитель матрицы: 18.0
```

Рис. 1 — Результат вывода в консоли

```
/home/keiichi/jdk-23.0.2/bin/java -javaagent:/home/keiichi/idea-23.0.2/lib/idea-runtime.jar -jar /home/keiichi/idea-23.0.2/lib/idea-runtime.jar
введите размер матрицы (n): 3

введите первую строку матрицы:
firstRow[0]: 1
firstRow[1]: 2
firstRow[2]: 3

введите первый столбец матрицы:
firstColumn[0]: 4
firstColumn[1]: 5
firstColumn[2]: 6

введите главную диагональ матрицы:
mainDiagonal[1]: 7
mainDiagonal[2]: 8

введённая матрица:
      4,0      2,0      3,0
      5,0      7,0      0,0
      6,0      0,0      8,0

правильно? (да/нет): да

определитель матрицы: 18.0

Process finished with exit code 0
```

Рис. 2 — Результат вывода в IntelliJ IDEA