

Міністерство освіти і науки України  
НТУ «Дніпровська політехніка»  
Кафедра програмного забезпечення комп'ютерних систем



Звіт з лабораторної роботи №2  
З дисципліни «Поглиблене програмування Java»

Виконала:

студентка групи 122-21-2

Таран Ірина Віталіївна

Перевірив:

доцент Мінеєв Олександр Сергійович

Дніпро  
2025

## ЛАБОРАТОРНА РОБОТА №2

Тема: «Основи»

Розробити програму, що дозволить вам створити, як з клавіатури так і рандомно матрицю цілих чисел типу `int` заданої ширини та висоти(ввести з клавіатури), але не більше 20 на 20. Створити можливість пошуку в цій матриці мінімального і максимального елементу та розрахунок середнього арифметичного. Програма може бути написана в одному класі, обов'язково розбиття на методи. Обов'язкове використання клавіатури, під час вибору ручного чи рандомного створення матриці. Створення системи зчитування з клавіатури зробити будь-яким способом, наприклад завдяки класу `Scanner`. `Scanner` являє собою найпростішу систему сканування клавіатури. Діапазон рандомних чисел для створення елементів матриці повинен зверігатись в спеціальних константах.

## Програмований код

```
MatrixOperations.java x
1 package org.example;
2
3 import java.util.Random;
4 import java.util.Scanner;
5
6 public class MatrixOperations { new *
7     private static final int MAX_SIZE = 20; 2 usages
8     private static final int MIN_VALUE = -100; 2 usages
9     private static final int MAX_VALUE = 100; 1 usage
10
11     public static void main(String[] args) { new *
12         Scanner scanner = new Scanner(System.in);
13
14
15         System.out.print("Input the number of rows (not more than 20): ");
16         int rows = validateInput(scanner);
17
18         System.out.print("Input the number of columns (not more than 20): ");
19         int cols = validateInput(scanner);
20
21
22         System.out.print("Input '1' for manual fill in, '2' for random generated: ");
23         int choice = scanner.nextInt();
24         int[][] matrix;
25
26         if (choice == 1) {
27             matrix = fillMatrixManually(rows, cols, scanner);
28         } else {
29             matrix = fillMatrixRandomly(rows, cols);
30         }
31
32
33         printMatrix(matrix);
34
35
36         int min = findMin(matrix);
37         int max = findMax(matrix);
38         double average = calculateAverage(matrix);
39
40         System.out.println("Min value: " + min);
41         System.out.println("Max value: " + max);
```

```

42         System.out.println("Average: " + average);
43
44         scanner.close();
45     }
46
47
48 @ private static int validateInput(Scanner scanner) { 2 usages new *
49     int value;
50     do {
51         value = scanner.nextInt();
52         if (value < 1 || value > MAX_SIZE) {
53             System.out.print("Invalid value! Input the number between 1 and 20: ");
54         }
55     } while (value < 1 || value > MAX_SIZE);
56     return value;
57 }
58
59
60 @ private static int[][] fillMatrixManually(int rows, int cols, Scanner scanner) { 1 usage new *
61     int[][] matrix = new int[rows][cols];
62     System.out.println("Input matrix elements:");
63     for (int i = 0; i < rows; i++) {
64         for (int j = 0; j < cols; j++) {
65             System.out.print("Element [" + i + "][" + j + "]: ");
66             matrix[i][j] = scanner.nextInt();
67         }
68     }
69     return matrix;
70 }
71
72
73 @ private static int[][] fillMatrixRandomly(int rows, int cols) { 1 usage new *
74     int[][] matrix = new int[rows][cols];
75     Random random = new Random();
76     for (int i = 0; i < rows; i++) {
77         for (int j = 0; j < cols; j++) {
78             matrix[i][j] = random.nextInt( bound: MAX_VALUE - MIN_VALUE + 1) + MIN_VALUE;
79         }
80     }

```

```
81         return matrix;
82     }
83
84
85     @ private static void printMatrix(int[][] matrix) { 1 usage new *
86         System.out.println("Matrix:");
87         for (int[] row : matrix) {
88             for (int value : row) {
89                 System.out.printf("%4d ", value);
90             }
91             System.out.println();
92         }
93     }
94
95
96     @ private static int findMin(int[][] matrix) { 1 usage new *
97         int min = Integer.MAX_VALUE;
98         for (int[] row : matrix) {
99             for (int value : row) {
100                 if (value < min) {
101                     min = value;
102                 }
103             }
104         }
105         return min;
106     }
107
108     @ private static int findMax(int[][] matrix) { 1 usage new *
109         int max = Integer.MIN_VALUE;
110         for (int[] row : matrix) {
111             for (int value : row) {
112                 if (value > max) {
113                     max = value;
114                 }
115             }
116         }
117         return max;
118     }
119
```

```

121  @      private static double calculateAverage(int[][] matrix) { 1 usage new *
122      int sum = 0;
123      int count = 0;
124      for (int[] row : matrix) {
125          for (int value : row) {
126              sum += value;
127              count++;
128          }
129      }
130      return (double) sum / count;
131  }
132  }
133

```

### Результат виконання коду

```

Run  MatrixOperations x
C:\Program Files\Java\jdk-1.8\bin\java.exe" ...
Input the number of rows (not more than 20): 15
Input the number of columns (not more than 20): 2
Input '1' for manual fill in, '2' for random generated: 2
Матриця:
 63  95
 51 -42
 32 -55
-83 -41
-19 -13
 55  63
 98   5
-71 -52
 85  62
 77  59
-60  72
 46 -41
 58  34
-60  11
 69 -59
Min value: -83
Max value: 98
Average: 14.633333333333333
Process finished with exit code 0

```

```
"C:\Program Files\Java\jdk-1.8\bin\java.exe" ...
```

```
Input the number of rows (not more than 20): 4
```

```
Input the number of columns (not more than 20): 4
```

```
Input '1' for manual fill in, '2' for random generated: 2
```

```
Матриця:
```

```
-66 -17 82 87
```

```
-77 -96 -98 27
```

```
-12 -10 30 88
```

```
-90 -79 91 -60
```

```
Min value: -98
```

```
Max value: 91
```

```
Average: -12.5
```

```
Process finished with exit code 0
```