МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний технічний університет

«Дніпровська політехніка»

Изображение выглядит как текст

Автоматически созданное описание

ЗВІТ

з лабораторної роботи №2

з дисципліни

**“Поглиблене програмування в середовищі Java”**

Виконав

студент групи

122-21-2

Гудошник Олександр Вадимович

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. Диапазон рандомних чисел для створення елементів матриці повинен зверігатись в спеціальних константах.

Як завдання підвищеної складності додати розрахунок середнього геометричного елементів матриці.

Хід роботи

1. Створюю гілку LR\_2.



Зображення, що містить текст, знімок екрана, Шрифт

Вміст, створений ШІ, може бути неправильним.

1. Створюю новий пакет для другої лабораторної роботи.

Створюю новий клас та код програми:

package Lab.LR\_2;  
  
import java.util.Random;  
import java.util.Scanner;  
  
public class MatrixProcessor {  
  
 // Constants for random number range  
 private static final int *MIN\_VALUE* = -100;  
 private static final int *MAX\_VALUE* = 100;  
 private static final int *MAX\_SIZE* = 20;  
  
 // Creating a matrix manually  
 public static int[][] inputMatrix(int rows, int cols) {  
 Scanner scanner = new Scanner(System.*in*);  
 int[][] matrix = new int[rows][cols];  
  
 System.*out*.println("Enter matrix elements:");  
 for (int r = 0; r < rows; r++) {  
 for (int c = 0; c < cols; c++) {  
 System.*out*.printf("Element [%d][%d]: ", r, c);  
 matrix[r][c] = scanner.nextInt();  
 }  
 }  
 return matrix;  
 }  
  
 // Generating a matrix randomly  
 public static int[][] generateRandomMatrix(int rows, int cols) {  
 Random random = new Random();  
 int[][] matrix = new int[rows][cols];  
  
 for (int r = 0; r < rows; r++) {  
 for (int c = 0; c < cols; c++) {  
 matrix[r][c] = random.nextInt(*MAX\_VALUE* \* 2 + 1) - *MAX\_VALUE*;  
 }  
 }  
 return matrix;  
 }  
  
 // Finding the minimum value  
 public static int getMinElement(int[][] matrix) {  
 int minValue = matrix[0][0];  
 for (int[] row : matrix) {  
 for (int num : row) {  
 if (num < minValue) {  
 minValue = num;  
 }  
 }  
 }  
 return minValue;  
 }  
  
 // Finding the maximum value  
 public static int getMaxElement(int[][] matrix) {  
 int maxValue = matrix[0][0];  
 for (int[] row : matrix) {  
 for (int num : row) {  
 if (num > maxValue) {  
 maxValue = num;  
 }  
 }  
 }  
 return maxValue;  
 }  
  
 // Calculating the arithmetic mean  
 public static double computeArithmeticMean(int[][] matrix) {  
 double sum = 0;  
 int count = 0;  
  
 for (int[] row : matrix) {  
 for (int num : row) {  
 sum += num;  
 count++;  
 }  
 }  
 return sum / count;  
 }  
  
 // Calculating the geometric mean  
 public static double computeGeometricMean(int[][] matrix) {  
 for (int[] row : matrix) {  
 for (int num : row) {  
 if (num < 0) {  
 System.*out*.println("Geometric mean cannot be calculated due to negative values.");  
 return Double.*NaN*;  
 }  
 }  
 }  
  
 double product = 1;  
 int count = 0;  
 for (int[] row : matrix) {  
 for (int num : row) {  
 product \*= num;  
 count++;  
 }  
 }  
 return Math.*pow*(product, 1.0 / count);  
 }  
  
 // Printing the matrix  
 public static void displayMatrix(int[][] matrix) {  
 for (int[] row : matrix) {  
 for (int num : row) {  
 System.*out*.print(num + "\t");  
 }  
 System.*out*.println();  
 }  
 }  
  
 public static void main(String[] args) {  
 Scanner scanner = new Scanner(System.*in*);  
  
 // Input matrix size  
 System.*out*.print("Enter number of rows: ");  
 int rows = scanner.nextInt();  
 System.*out*.print("Enter number of columns: ");  
 int cols = scanner.nextInt();  
  
 if (rows > *MAX\_SIZE* || cols > *MAX\_SIZE*) {  
 System.*out*.println("Matrix size cannot exceed 20x20.");  
 return;  
 }  
  
 // Choosing matrix creation method  
 System.*out*.println("Choose matrix creation method:");  
 System.*out*.println("1. Manual input");  
 System.*out*.println("2. Random generation");  
 int option = scanner.nextInt();  
  
 int[][] matrix;  
 if (option == 1) {  
 matrix = *inputMatrix*(rows, cols);  
 } else if (option == 2) {  
 matrix = *generateRandomMatrix*(rows, cols);  
 } else {  
 System.*out*.println("Invalid choice.");  
 return;  
 }  
  
 // Displaying and calculating results  
 System.*out*.println("Matrix:");  
 *displayMatrix*(matrix);  
  
 int minValue = *getMinElement*(matrix);  
 int maxValue = *getMaxElement*(matrix);  
 double avgArithmetic = *computeArithmeticMean*(matrix);  
 double avgGeometric = *computeGeometricMean*(matrix);  
  
 System.*out*.println("Minimum element: " + minValue);  
 System.*out*.println("Maximum element: " + maxValue);  
 System.*out*.println("Arithmetic mean: " + avgArithmetic);  
 System.*out*.println("Geometric mean: " + (Double.*isNaN*(avgGeometric) ? "Cannot be calculated" : avgGeometric));  
 }  
}

Додаю всі зміни та виконую коміт:

Зображення, що містить текст, знімок екрана, Шрифт

Вміст, створений ШІ, може бути неправильним.

1. Запушую гілку на гітхаб.
2. Створюю Pull Request на GitHub та виконую Merge pull request.
3. Також оновлюю локальну версію гілки.

Бачимо результат:

Зображення, що містить текст, знімок екрана, програмне забезпечення, Мультимедійне програмне забезпечення

Вміст, створений ШІ, може бути неправильним.

Результат виконання створеної програми:

Зображення, що містить текст, знімок екрана, Шрифт

Вміст, створений ШІ, може бути неправильним.

Або так:

Зображення, що містить текст, знімок екрана, Шрифт

Вміст, створений ШІ, може бути неправильним.

Висновок. Під час виконання лабораторної роботи №2 з дисципліни «Поглиблене програмування в середовищі Java» було вивчено основи мови Java та створено програму згідно завданню.