**Лабораторная работа №7**

**Тема: Двумерные массивы (матрицы). Анализ элементов матрицы**

Выполнил студент: **Иванов Иван Иванович**

Группа: **235**

**Вариант: 0**

**Цель работы:**

**\* Закрепить принципы тестирования массивов.**

**\* Разработать тест-кейсы для проверки корректности работы методов.**

**\* Провести тестирование на основе позитивных и негативных сценариев.**

**Задание**

**Реализовать метод:**

public static int rowWithMaxSum(int[][] matrix)

Метод должен возвращать номер строки матрицы, сумма элементов которой максимальна.

Если матрица пустая (например, 0 строк или 0 столбцов), можно вернуть -1.Пример класса для тестирования

public class Lab7 {  
 public static int rowWithMaxSum(int[][] matrix) {  
 if (matrix == null || matrix.length == 0) {  
 return -1; // Возвращаем -1, если матрица пустая  
 }  
  
 int maxSum = Integer.MIN\_VALUE;  
 int rowIndex = -1;  
  
 for (int i = 0; i < matrix.length; i++) {  
 int sum = 0;  
 for (int value : matrix[i]) {  
 sum += value;  
 }  
 if (sum > maxSum) {  
 maxSum = sum;  
 rowIndex = i;  
 }  
 }  
  
 return rowIndex;  
 }  
  
}

**Пример тестового класса**

import org.junit.jupiter.api.Test;  
import static org.junit.jupiter.api.Assertions.\*;  
  
public class Lab7Test {  
  
 @Test  
 void testTypicalCase() {  
 int[][] matrix = {  
 {1, 2, 3},  
 {4, 5, 6},  
 {7, 8, 9}  
 };  
 assertEquals(2, Lab7.rowWithMaxSum(matrix)); // Суммы: 6, 15, 24  
 }  
  
}

**тест-кейс**

|  |  |
| --- | --- |
| **Поле** | **Значение** |
| **ID тест-кейса** | TC001 |
| **Название тест-кейса** | Проверка функции rowWithMaxSum |
| **Описание** | Проверить, что функция rowWithMaxSum(int[][] matrix) корректно находит индекс строки с наибольшей суммой элементов в матрице. |
| **Предусловия** | Матрица содержит данные, размеры матрицы > 0 |
| **Шаги выполнения** | 1. Создать входную матрицу:  int[][] expected = {  {1, 2, 3},  {4, 5, 6},  {7, 8, 9}  }; |
|  | 2. Вызвать метод:  int actual = Lab7.rowWithMaxSum(expected); |
|  | 3. Сравнить ожидаемую и фактическую матрицы с помощью Assertions.assertEquals(2, actual);. |
| **Ожидаемый результат** | Метод возвращает значение 2, так как третья строка имеет наибольшую сумму (24) |
| **Фактический результат** | Результат совпадает с ожидаемым: 2 |
| **Статус** | Pass |
| **Примечания** | Тест проверяет типовой случай. Дополнительные тесты должны покрывать граничные случаи (например, отрицательные числа, все строки с одинаковой суммой и т. д.). |

### Контрольные вопросы

1. Как в Java объявить и инициализировать двумерный массив? Приведите пример.

2. Что означает, что двумерный массив в Java "прямоугольный" или "ступенчатый"? В чём разница?

3. Как найти сумму элементов заданной строки матрицы?

4. Как определить номер строки матрицы с наибольшей суммой элементов?

5. Как обработать случай, когда матрица пустая или содержит нулевые ссылки?