2D Array Solutions

Q1: Take m and n input from the user and m \* n integer inputs from user and print the following:

number of positive numbers

number of negative numbers

number of odd numbers

number of even numbers

number of 0. Input 1:  Output :  number of positive numbers = 7 number of negative numbers = 6 number of odd numbers = 7 number of even numbers = 9 number of 0 = 3

import java.util.Scanner;

public class MatrixStatistics {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

// Input the dimensions of the matrix

System.out.print("Enter the number of rows (m): ");

int m = scanner.nextInt();

System.out.print("Enter the number of columns (n): ");

int n = scanner.nextInt();

int[][] matrix = new int[m][n];

// Variables to hold the counts

int positiveCount = 0;

int negativeCount = 0;

int oddCount = 0;

int evenCount = 0;

int zeroCount = 0;

// Input the matrix elements and compute the statistics

System.out.println("Enter the elements of the matrix:");

for (int i = 0; i < m; i++) {

for (int j = 0; j < n; j++) {

matrix[i][j] = scanner.nextInt();

// Counting positive, negative, odd, even, and zero

if (matrix[i][j] > 0) {

positiveCount++;

} else if (matrix[i][j] < 0) {

negativeCount++;

}

if (matrix[i][j] == 0) {

zeroCount++;

} else if (matrix[i][j] % 2 == 0) {

evenCount++;

} else {

oddCount++;

}

}

}

// Output the results

System.out.println("Number of positive numbers = " + positiveCount);

System.out.println("Number of negative numbers = " + negativeCount);

System.out.println("Number of odd numbers = " + oddCount);

System.out.println("Number of even numbers = " + evenCount);

System.out.println("Number of 0 = " + zeroCount);

}

}

Q2: write a program to print the elements above the secondary diagonal in a user inputted square matrix.

Input 1:

………….

Output : 1 2 4

import java.util.Scanner;

public class ElementsAboveSecondaryDiagonal {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

// Input the size of the square matrix

System.out.print("Enter the size of the square matrix (n x n): ");

int n = scanner.nextInt();

int[][] matrix = new int[n][n];

// Input the matrix elements

System.out.println("Enter the elements of the matrix:");

for (int i = 0; i < n; i++) {

for (int j = 0; j < n; j++) {

matrix[i][j] = scanner.nextInt();

}

}

// Print elements above the secondary diagonal

System.out.print("Elements above the secondary diagonal: ");

for (int i = 0; i < n; i++) {

for (int j = 0; j < n - i - 1; j++) {

System.out.print(matrix[i][j] + " ");

}

}

}

}

Q3: write a program to print the elements of both the diagonals in a user inputted square matrix in any order.

Input 1:

Output 1: 1 3 5 7  9

import java.util.Scanner;

public class DiagonalElements {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

// Input the size of the square matrix

System.out.print("Enter the size of the square matrix (n x n): ");

int n = scanner.nextInt();

int[][] matrix = new int[n][n];

// Input the matrix elements

System.out.println("Enter the elements of the matrix:");

for (int i = 0; i < n; i++) {

for (int j = 0; j < n; j++) {

matrix[i][j] = scanner.nextInt();

}

}

// Print the elements of the primary diagonal

System.out.print("Primary diagonal elements: ");

for (int i = 0; i < n; i++) {

System.out.print(matrix[i][i] + " ");

}

// Print the elements of the secondary diagonal

System.out.print("\nSecondary diagonal elements: ");

for (int i = 0; i < n; i++) {

System.out.print(matrix[i][n - 1 - i] + " ");

}

}

}

Q4: Write a program to find the largest element of a given 2D array of integers.

import java.util.Scanner;

public class LargestElement2DArray {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

// Input the number of rows and columns for the 2D array

System.out.print("Enter the number of rows: ");

int rows = scanner.nextInt();

System.out.print("Enter the number of columns: ");

int cols = scanner.nextInt();

int[][] array = new int[rows][cols];

// Input the elements of the 2D array

System.out.println("Enter the elements of the 2D array:");

for (int i = 0; i < rows; i++) {

for (int j = 0; j < cols; j++) {

array[i][j] = scanner.nextInt();

}

}

// Find the largest element

int largest = array[0][0];

for (int i = 0; i < rows; i++) {

for (int j = 0; j < cols; j++) {

if (array[i][j] > largest) {

largest = array[i][j];

}

}

}

// Output the largest element

System.out.println("The largest element in the 2D array is: " + largest);

}

}

Q5: Write a function which accepts a 2D array of integers and its size as arguments and displays the elements of middle row and the elements of middle column. Printing can be done in any order.

[Assuming the 2D Array to be a square matrix with odd dimensions i.e. 3x3, 5x5, 7x7 etc...]

Input 1:

………………..

Output : 3 5 5 6 37 7 6 4 3

public class MiddleElements {

public static void main(String[] args) {

// Example usage

int[][] matrix = {

{3, 5, 7},

{6, 37, 8},

{4, 3, 2}

};

// Size of the matrix (must be odd)

int size = 3;

// Call the function

printMiddleRowAndColumn(matrix, size);

}

public static void printMiddleRowAndColumn(int[][] matrix, int size) {

// Check if the matrix size is valid (must be odd)

if (size % 2 == 0) {

System.out.println("Matrix size must be odd.");

return;

}

// Calculate the index of the middle row and column

int midIndex = size / 2;

// Print the elements of the middle row

System.out.print("Middle row elements: ");

for (int j = 0; j < size; j++) {

System.out.print(matrix[midIndex][j] + " ");

}

// Print the elements of the middle column

System.out.print("\nMiddle column elements: ");

for (int i = 0; i < size; i++) {

System.out.print(matrix[i][midIndex] + " ");

}

System.out.println(); // For a newline after the output

}

}