**Java - Introduction to Programming**

**Lecture 11**

**2D Arrays In Java**

It is similar to 2D matrices that we studied in 11th and 12th class.

1. Creating a 2D Array - with new keyword

int[][] marks = new int[3][3];

1. Taking a matrix as an input and printing its elements.

import java.util.\*;

public class TwoDArrays {

public static void main(String args[]) {

Scanner sc = new Scanner(System.in);

int rows = sc.nextInt();

int cols = sc.nextInt();

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

//input

//rows

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

//columns

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

numbers[i][j] = sc.nextInt();

}

}

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

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

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

}

System.out.println();

}

}

}

1. Searching for an element x in a matrix.

**import java.util.\*;**

**public class TwoDArrays {**

**public static void main(String args[]) {**

**Scanner sc = new Scanner(System.in);**

**int rows = sc.nextInt();**

**int cols = sc.nextInt();**

**int[][] numbers = new int[rows][cols];**

**//input**

**//rows**

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

**//columns**

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

**numbers[i][j] = sc.nextInt();**

**}**

**}**

**int x = sc.nextInt();**

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

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

**//compare with x**

**if(numbers[i][j] == x) {**

**System.out.println("x found at location (" + i + ", " + j + ")");**

**}**

**}**

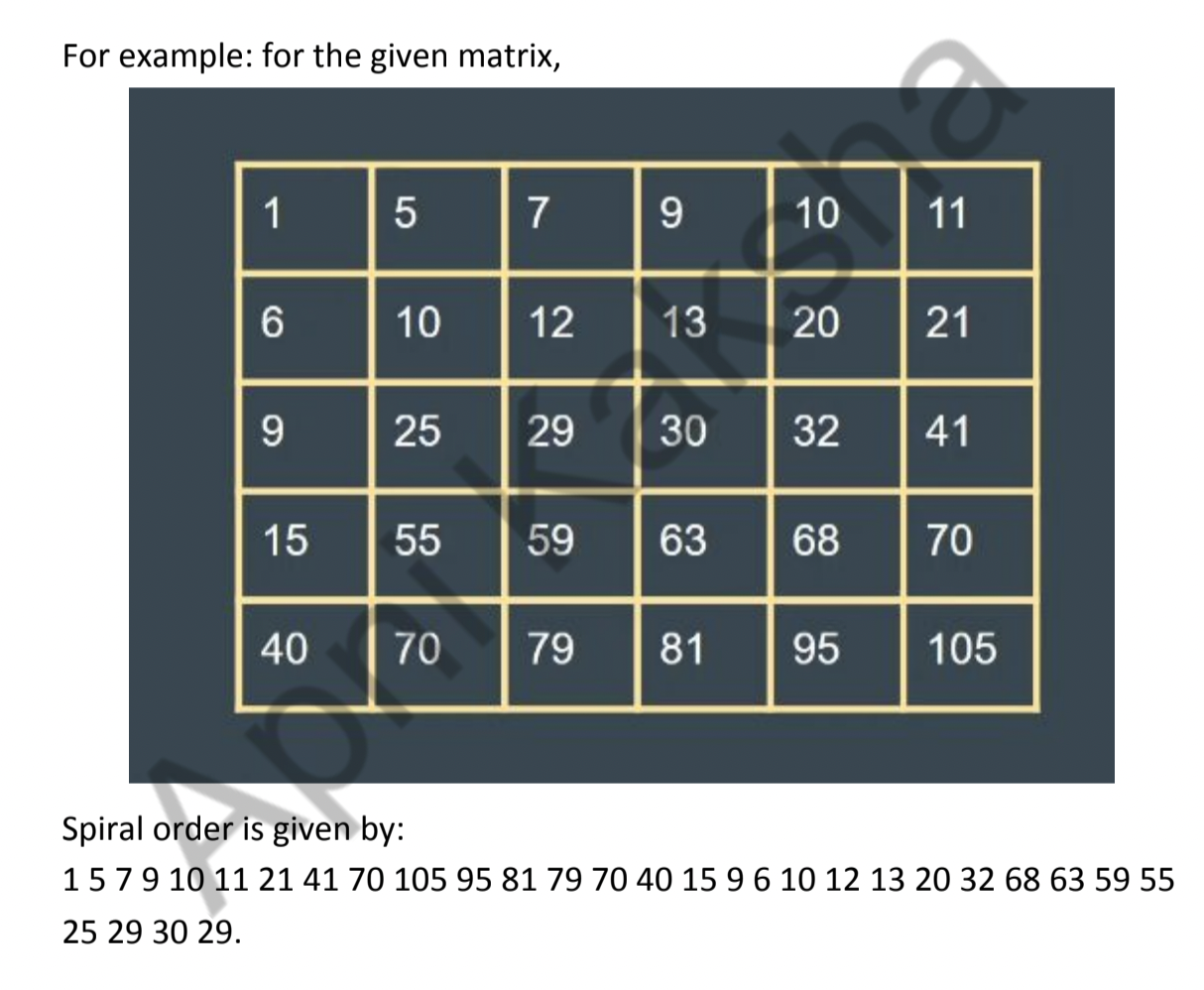
**}**

**}**

**}**

**Homework Problems**

1. Print the spiral order matrix as output for a given matrix of numbers. ***DOUBT*** [Difficult for Beginners]



**APPROACH** :

Algorithm: (We are given a 2D matrix of n X m ).

*1. We will need 4 variables:*

*a. row\_start - initialized with 0.*

*b. row\_end - initialized with n-1.*

*c. column\_start - initialized with 0.*

*d. column\_end - initialized with m-1.*

*2. First of all, we will traverse in the row row\_start from column\_start*

*to column\_end and we will increase the row\_start with 1 as we have*

*traversed the starting row.*

*3. Then we will traverse in the column column\_end from row\_start to*

*row\_end and decrease the column\_end by 1.*

*4. Then we will traverse in the row row\_end from column\_end to*

*column\_start and decrease the row\_end by 1.*

*5. Then we will traverse in the column column\_start from row\_end to*

*row\_start and increase the column\_start by 1.*

*6. We will do the above steps from 2 to 5 until row\_start <= row\_end*

*and column\_start <= column\_end.*

import java.util.\*;

public class Arrays {

public static void main(String args[]) {

Scanner sc = new Scanner(System.in);

int n = sc.nextInt();

int m = sc.nextInt();

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

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

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

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

}

}

System.out.println("The Spiral Order Matrix is : ");

int rowStart = 0;

int rowEnd = n-1;

int colStart = 0;

int colEnd = m-1;

//To print spiral order matrix

while(rowStart <= rowEnd && colStart <= colEnd) {

//1

for(int col=colStart; col<=colEnd; col++) {

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

}

rowStart++;

//2

for(int row=rowStart; row<=rowEnd; row++) {

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

}

colEnd--;

//3

for(int col=colEnd; col>=colStart; col--) {

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

}

rowEnd--;

//4

for(int row=rowEnd; row>=rowStart; row--) {

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

}

colStart++;

System.out.println();

}

}

}

1. For a given matrix of N x M, print its transpose.

import java.util.\*;

public class Arrays {

public static void main(String args[]) {

Scanner sc = new Scanner(System.in);

int n = sc.nextInt();

int m = sc.nextInt();

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

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

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

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

}

}

System.out.println("The transpose is : ");

//To print transpose

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

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

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

}

System.out.println();

}

}

}