

Introduction To 2D Array[Day 4]

05 September 2024 18:52

Introduction To 2D Array [Day - 4]

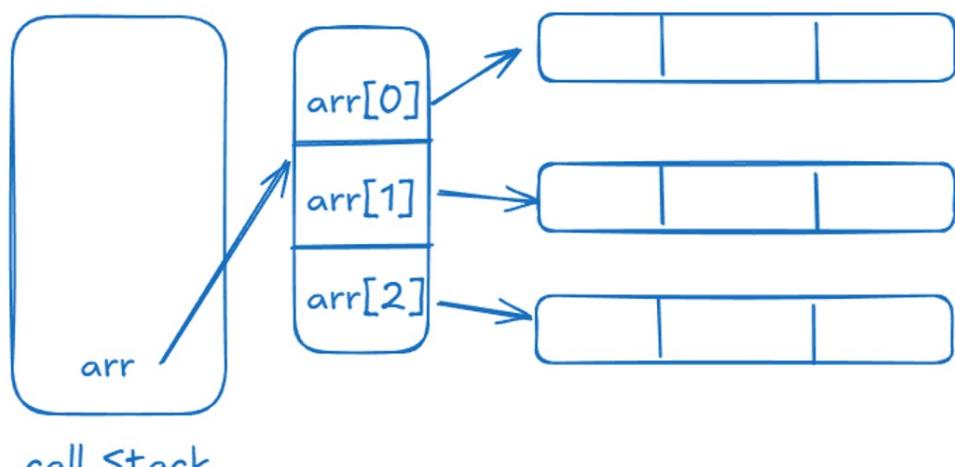
| | 0 | 1 | 2 |
|---|----------|----------|----------|
| 0 | $A(0,0)$ | $A(0,1)$ | $A(0,2)$ |
| 1 | $A(1,0)$ | $A(1,1)$ | $A(1,2)$ |
| 2 | $A(2,0)$ | $A(2,1)$ | $A(2,2)$ |

Arr - 3*3 Matrix

`int[][] arr = new int[n][n]`

row - 0 to arr.length-1

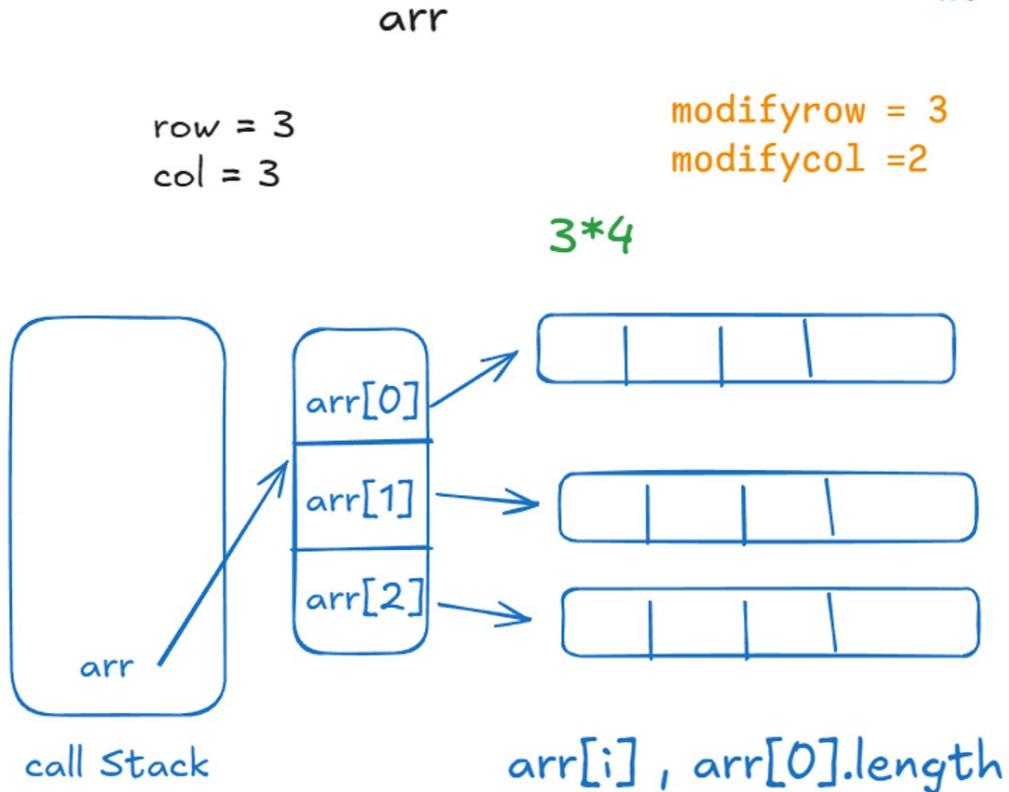
col - 0 to arr[i].length



call Stack

- All operations Of 2D Array.

| | 0 | 1 | 2 |
|---|----|----|---------------------|
| 0 | 6 | 4 | 5 |
| 1 | 9 | 11 | 7 |
| 2 | 21 | 19 | $14 \rightarrow 12$ |



```

public static void modifyElement(int[][] arr , int row , int col , int newValue){
    if(row >=0 && row < arr.length && col >=0 && col< arr[row].length){
        arr[row][col] = newValue;
    }else{
        System.out.println("Invalid Index");
    }
}

System.out.println("Enter the row index to modify: ");
int modifyRow = scn.nextInt();
System.out.println("Enter the column index to modify: ");
int modifyCol = scn.nextInt();
System.out.println("Enter the new Value");
int newValue = scn.nextInt();

modifyElement(arr , modifyRow, modifyCol , newValue);
printArray(arr);

```

Final Code :

2DArrayOperations.

```

import java.util.Scanner;
public class TwoDArrayOperations {
    public static int[][] intitalizeArray(int rows , int cols){
        Scanner scn = new Scanner(System.in);
        int[][] arr = new int[rows][cols];
        System.out.println("Enter the elements of an array:");
        for(int i =0 ; i< rows ; i++){
            for(int i = 0 : i< cols : i++){

```

```

System.out.println("Enter the elements of an array:");
for(int i =0 ; i< rows ; i++){
    for(int j = 0 ; j< cols ; j++){
        arr[i][j] = scn.nextInt();
    }
}
return arr;
}

public static void printArray(int[][] arr){
System.out.println("Array Element");
for(int i = 0 ; i < arr.length ; i++){
    for(int j = 0 ; j< arr[0].length; j++){
        System.out.print(arr[i][j] + " ");
    }
    System.out.println();
}
}

public static void modifyElement(int[][] arr , int row , int col , int newValue){
if(row >=0 && row < arr.length && col >=0 && col< arr[row].length){
    arr[row][col] = newValue;
}else{
    System.out.println("Invalid Index");
}
}

public static int sumOfElement(int[][] arr){
int sum =0;
for(int i = 0; i< arr.length; i++){
    for(int j = 0 ; j< arr[0].length ; j++){
        sum += arr[i][j];
    }
}
return sum;
}

public static int maxElement(int[][] arr){
int max = arr[0][0];
for(int i = 0 ; i < arr.length ; i++){
    for(int j = 0 ; j<arr[0].length ; j++){
        if(arr[i][j] > max){
            max = arr[i][j];
        }
    }
}
return max;
}

public static int minElement(int[][] arr){
int min = arr[0][0];
for(int i = 0 ; i < arr.length ; i++){
    for(int j = 0 ; j<arr[0].length ; j++){
        if(arr[i][j] < min){
            min = arr[i][j];
        }
    }
}
}

```

```

        }
    }
    return min;
}
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    System.out.println("Enter the number of rows");
    int rows = scn.nextInt();
    System.out.println("Enter the number of columns");
    int cols = scn.nextInt();
    int[][] arr = initializeArray(rows , cols);
    printArray(arr);

    // Modify the Element
    System.out.println("Enter the row index to modify: ");
    int modifyRow = scn.nextInt();
    System.out.println("Enter the column index to modify: ");
    int modifyCol = scn.nextInt();
    System.out.println("Enter the new Value");
    int newValue = scn.nextInt();

    modifyElement(arr , modifyRow, modifyCol , newValue);
    printArray(arr);

    int sum = sumOfElement(arr);
    System.out.println("Sum of all Elements : " + sum);

    int max = maxElement(arr);
    System.out.println("Max of an Array :" + max);
    int min = minElement(arr);
    System.out.println("Min of an Array :" + min);
}
}

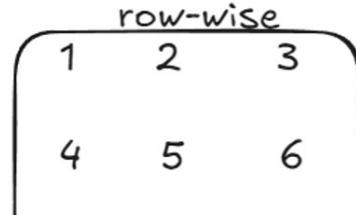
```

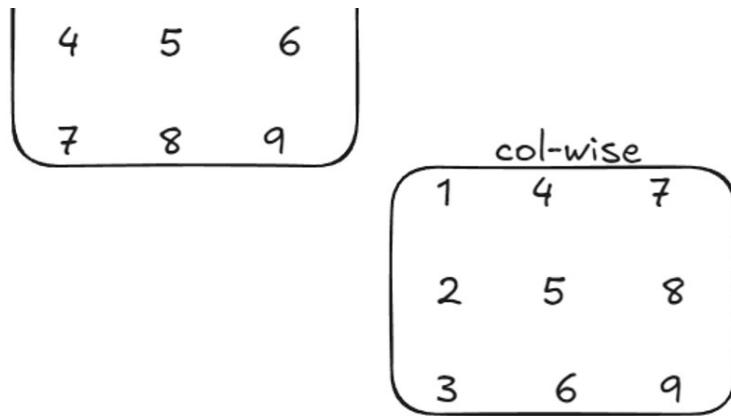
$O(3N^2)$

Write a program to traverse a matrix in different orders

[row-direction , column-direction]

| | | |
|---|---|---|
| 1 | 2 | 3 |
| 4 | 5 | 6 |
| 7 | 8 | 9 |





```

public class MatrixTraversal {
    public static void printRowWise(int[][] matrix){
        System.out.println("Row Wise Traversal");
        for(int i = 0 ; i<matrix.length; i++){
            for(int j = 0; j< matrix[i].length; j++){
                System.out.print(matrix[i][j] + " ");
            }
            System.out.println();
        }
    }

    public static void printColumnWise(int[][] matrix){
        System.out.println("Column Wise Traversal");
        for(int j = 0; j< matrix[0].length; j++){
            for(int i = 0 ; i<matrix.length; i++){
                System.out.print(matrix[i][j] + " ");
            }
            System.out.println();
        }
    }

    public static void main(String[] args) {
        int[][] matrix ={
            {1,2,3},
            {4,5,6},
            {7,8,9}
        };

        printRowWise(matrix);
        printColumnWise(matrix);
    }
}

```

Take the input

Time & Space - $O(N^2)$;

Objective:

Given a 2D array, perform a wave traversal. In wave traversal, we traverse each column, moving

Objective:

Given a 2D array, perform a wave traversal. In wave traversal, we traverse each column, moving downwards in the odd-numbered columns (starting from the left) and upwards in the even-numbered columns.

Example

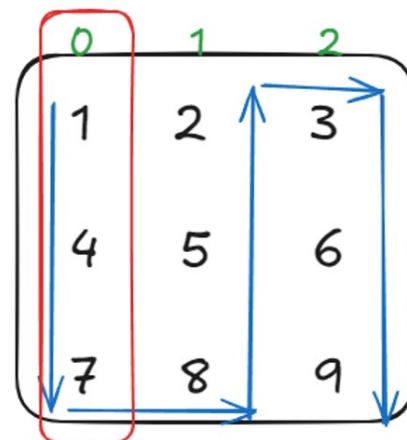
Example 1:

- Input:

```
makefile
Matrix:
1 2 3
4 5 6
7 8 9
```

- Output:

```
yaml
Wave Traversal:
1 4 7 8 5 2 3 6 9
```



$j \% 2 == 0$ [even]
- downward

odd - upward.

1, 4, 7, 8, 5, 2, 3, 6, 9

```
import java.util.Scanner;
public class WaveTraversal {

    public static void waveTraversal(int[][] matrix , int m , int n){
        for(int j = 0 ; j<n ; j++){
            if(j % 2 ==0){
                for(int i =0 ; i<m ; i++){
                    System.out.print(matrix[i][j] + " ");
                }
            }else{
                for(int i = m-1 ; i>=0 ; i--){
                    System.out.print(matrix[i][j] + " ");
                }
            }
        }
    }
}
```

```

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

public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);

    System.out.println("Enter the number of Rows:");
    int m = scn.nextInt();
    System.out.println("Enter the number of Columns:");
    int n = scn.nextInt();

    int[][] matrix = new int[m][n];
    System.out.println("Enter the Element of an matrix");

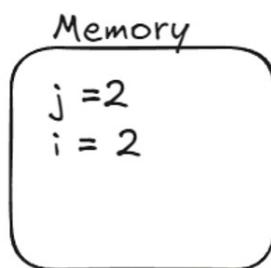
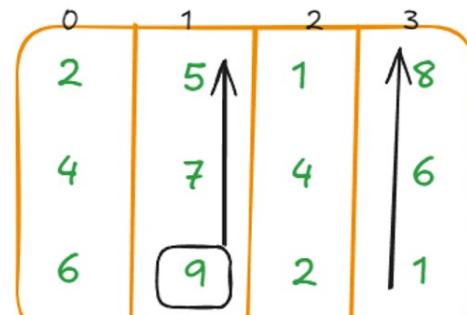
    for(int i =0; i<m ; i++){
        for(int j =0 ; j< n ; j++){
            matrix[i][j] = scn.nextInt();
        }
    }

    System.out.println("Wave Traversal");
    waveTraversal(matrix , m ,n);
}

}

for(int j = 0 ; j<n ; j++){
    if(j % 2 ==0){
        for(int i =0 ; i<m ; i++){
            System.out.print(matrix[i][j] + " ");
        }
    }else{
        for(int i = m-1 ; i>=0 ; i--){
            System.out.print(matrix[i][j] + " ");
        }
    }
}

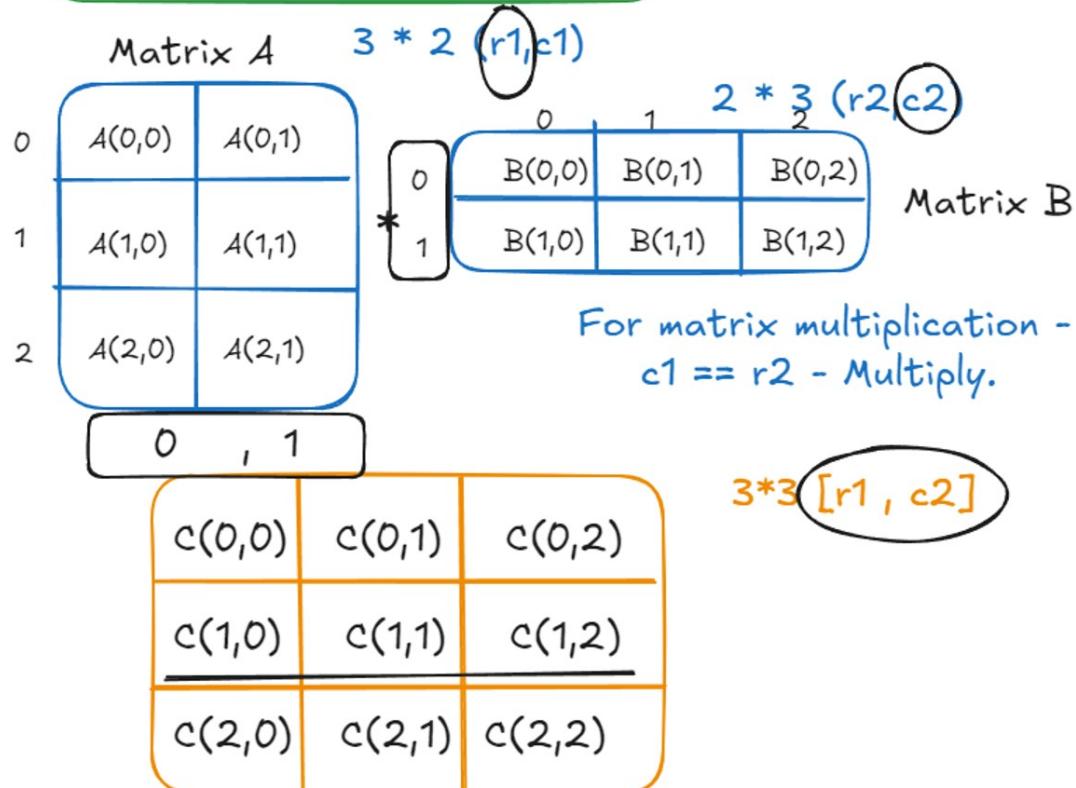
```



2, 4, 6, 9, 7, 5, 1, 6, 8

Matrix Multiplication

Matrix Multiplication



$$c(0,0) = A(0,0) * B(0,0) + A(0,1) * B(1,0)$$

$$c(0,1) = A(0,0) * B(0,1) + A(0,1) * B(1,1)$$

$$c(0,2) = A(0,0) * B(0,2) + A(0,1) * B(1,2)$$

$$c(i,j) = A(i,k) * B(k,j) + A(i,k) * B(k,j)$$

Change Status

Completed

Attendance Code: 189D7328

[Close](#) [Update](#)

```
import java.util.Scanner;
public class MatrixMultiplication {
    public static int[][] multiplyMatrix(int[][] A, int[][] B){
        int m = A.length; // row of A
        int n = A[0].length; // col of A or can be row of B B.length
        int p = B[0].length; // col of B
        int[][] C = new int[m][n];
        for (int i = 0; i < m; i++) {
            for (int j = 0; j < n; j++) {
                for (int k = 0; k < p; k++) {
                    C[i][j] += A[i][k] * B[k][j];
                }
            }
        }
        return C;
    }
}
```

```

int n = A[0].length; // col of A or can be row of B B.length
int p = B[0].length; // col of B
int[][] C = new int[m][p];

for(int i = 0 ; i<m; i++){
    for(int j = 0 ; j<p; j++){
        int sum = 0;
        for(int k = 0 ; k<n ; k++){
            sum += A[i][k] * B[k][j];
        }
        C[i][j] = sum;
    }
}

return C;
}

public static void main(String[] args) {

    Scanner scn = new Scanner(System.in);
    System.out.println("Enter the numbers of rows for Matrix A:");
    int rowsA = scn.nextInt();
    System.out.println("Enter the numbers of Columns for Matrix A:");
    int colsA = scn.nextInt();
    int[][]A = new int[rowsA][colsA];
    System.out.println("Enter the elements of Matrix A");
    for(int i=0 ; i< rowsA; i++){
        for(int j = 0; j< colsA; j++){
            A[i][j] = scn.nextInt();
        }
    }

    System.out.println("Enter the numbers of rows for Matrix B:");
    int rowsB = scn.nextInt();
    System.out.println("Enter the numbers of Columns for Matrix B:");
    int colsB = scn.nextInt();
    int[][]B = new int[rowsB][colsB];
    System.out.println("Enter the elements of Matrix B");
    for(int i=0 ; i< rowsB; i++){
        for(int j = 0; j< colsB; j++){
            B[i][j] = scn.nextInt();
        }
    }

    // Check if matrices can be multiplied;
    if(colsA != rowsB){
        System.out.println("Matrix can't multiply");
        return;
    }

    int[][] C = multiplyMatrix(A,B);

    System.out.println("Matrix C Resultant");
    for(int i = 0 ; i<C.length; i++){
        for(int j =0 ; j<C[i].length ; j++){
            System.out.print(C[i][j] + " ");
        }
    }
}

```

```

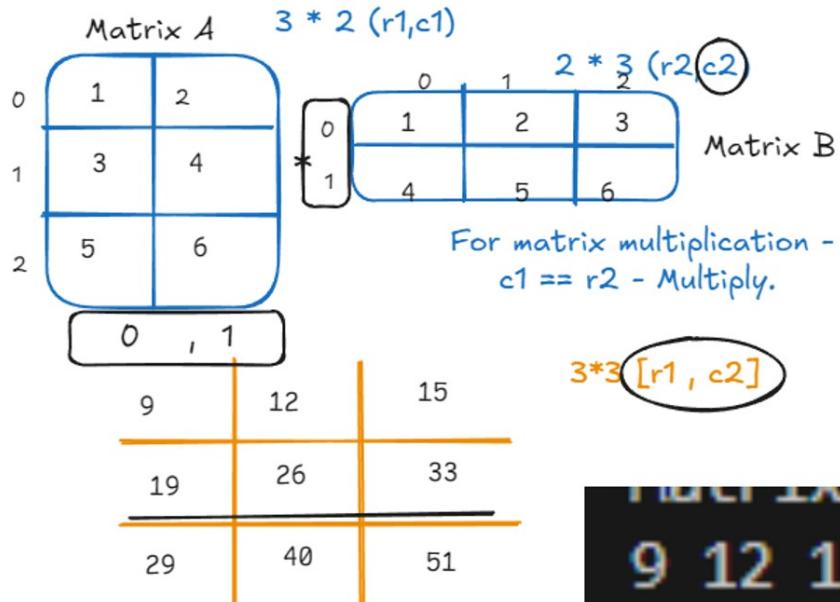
        for(int i = 0 ; i<C.length; i++){
            for(int j = 0 ; j<C[i].length ; j++){
                System.out.print(C[i][j] + " ");
            }
            System.out.println();
        }

    }

int m = A.length; // row of A
int n = A[0].length; // col of A or can be called as r1, c1
int p = B[0].length; // col of B
int[][] C = new int[m][p];

```

3 2 [k] = B.length = 2
3 3
3*3



Matrix C

| | | |
|----|----|----|
| 9 | 12 | 15 |
| 19 | 26 | 33 |
| 29 | 40 | 51 |