

# Print Alternate Row (14 July)

Problem

Submissions

Leaderboard

Discussions

You are given a 2D matrix, your task is to print the alternate rows of the matrix starting from the 0th row.

## Input Format

First line contains, m and n depicting the size of the matrix. m\*n Integer values, depicting all the elements of the matrix.

## Constraints

1 <= m and n <= 1000 -1000 <= mat[i][j] <= 1000

## Output Format

Print the alternate row.

## Sample Input 0

## Sample Input 0

```
4 6
2 3 8 7 0 4
0 7 6 7 3 5
0 0 8 1 0 8
9 1 9 5 3 0
```

## Sample Output 0

```
2 3 8 7 0 4
0 0 8 1 0 8
```

m = 4

n = 6

for(int i=0; i<mp; i+=2)

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

{ cout << mat[i][j] << " ";

i=0

i=2

i=4

cout << endl;

	0	1	2	3	4	5
0	2	3	8	7	0	4
1	0	7	6	7	3	5
2	0	0	8	1	0	8
3	9	1	9	5	3	0

```
2 3 8 7 0 4
0 0 8 1 0 8
```

```

public class Solution {
    public static void printAlternateRow(int arr[][], int m, int n){
        for(int i=0; i<m; i+=2){
            for(int j=0; j<n; j++){
                System.out.print(arr[i][j] + " ");
            }
            System.out.println();
        }
    }

    public static void main(String[] args) {
        /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your
        Scanner scn = new Scanner(System.in);
        int m = scn.nextInt();
        int n = scn.nextInt();

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

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

        printAlternateRow(arr, m, n);
    }
}

```

→ 5 mins +

5 mins → Alternate  
col wise

Print the matrix column wise such that we print the alternate columns of the matrix starting from the first column.

outer

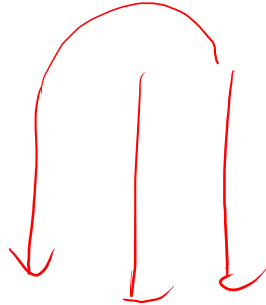
→ for(int j=0; j<n; j+=2)

{

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

cout

,  
cout



	0	1	2
0	49	22	16
1	71	78	23
2	89	24	61

Sample Input 0

```
3
3 4 5
49 22 16
71 78 23
89 24 61
```

Sample Output 0

```
49 71 89
16 23 61
```

```

public class Solution {
    public static void printAlternateColumn(int arr[][], int m, int n){
        for(int j=0; j<n; j+=2){
            for(int i=0; i<m; i++){
                System.out.print(arr[i][j] + " ");
            }
            System.out.println();
        }
    }

    public static void main(String[] args) {
        /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your class sh
        Scanner scn = new Scanner(System.in);
        int m = scn.nextInt();
        int n = scn.nextInt();
        int arr[][] = new int[m][n];

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

        printAlternateColumn(arr, m, n);
    }
}

```

→ 5 min

+ 5 min

↳ Interchange  
arr[i][j]

Interchange elements of the first and last row of a matrix.

#### Input Format

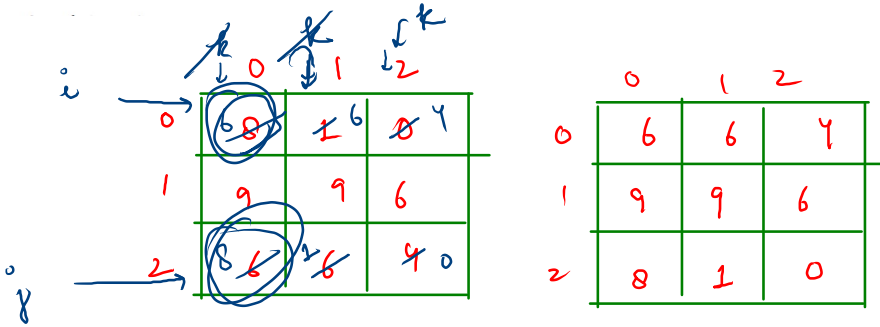
First line contains, m and n depicting the size of the first matrix. m\*n Integer values, depicting all the elements of the matrix.

#### Constraints

$1 \leq m$  and  $n \leq 1000$  -  $1000 \leq \text{mat}[i][j] \leq 1000$

#### Output Format

Print the matrix after interchanging the row



#### Sample Input 0

```
m → 3
n → 3
{
  8 1 0
  9 9 6
  6 6 4
}
```

#### Sample Output 0

```
6 6 4
9 9 6
8 1 0
```

```
int i=0, j=m-1
for(int k=0; k<n; k++)
```

```
{
  int a = arr[i][k];
  int b = arr[j][k];
  int temp = arr[i][k];
  a = b;
  b = temp;
}
```

$m=3$  (0, 1, 2)

→ 0      1      2

6	6	4
9	9	6
8	2	0

→ 2

$k=0 < 3$  (T)

$temp = arr[0][0] = 8$

$arr[0][0] = arr[2][0] = 8$

$arr[2][0] = 8$

$k=1 < 3$  (T)

$temp = arr[0][1] = 1$

$arr[0][1] = arr[2][1] = 6$

$arr[2][1] = 1$

```
public class Solution {
    public static void interchangeRows(int arr[][], int m, int n){
        for(int k=0; k<n; k++){
            int temp = arr[0][k];
            arr[0][k] = arr[m-1][k];
            arr[m-1][k] = temp;
        }
    }

    for(int i=0; i<m; i++){
        for(int j=0; j<n; j++){
            System.out.print(arr[i][j] + " ");
        }
        System.out.println();
    }
}
```

→ 5 min  
+  
5 min  
= Lower  
Triangle  
Matrix

```
public static void main(String[] args) {
    /* Enter your code here. Read input from STDIN. Print output to STDOUT.
    Scanner scn = new Scanner(System.in);
    int m = scn.nextInt();
    int n = scn.nextInt();
    int arr[][] = new int[m][n];
    for(int i=0; i<m; i++){
        for(int j=0; j<n; j++){
            arr[i][j] = scn.nextInt();
        }
    }

    interchangeRows(arr, m, n);
}
```

$k=2 < 3$  (T)

$temp = arr[0][2] = 0$

$arr[0][2] = arr[2][2] = 4$

$arr[2][2] = 0$

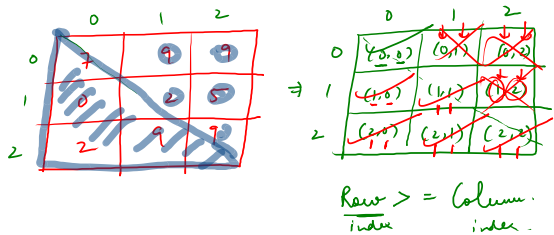
$k=3 < 3$  (F) X

6	6	4
9	9	6
8	1	0

# Print Lower triangular matrix (15 july)

Problem	Submissions	Leaderboard	Discussions
---------	-------------	-------------	-------------

Print the lower triangle of a matrix.



Sample Input 0

```
m -> 3
n -> 3
7 9 9
0 2 5
2 9 9
```

Sample Output 0

```
7 0 0
0 2 0
2 9 9
```

```
for (int i = 0; i < m; i++)
{
    for (int j = 0; j < n; j++)
    {
        if (i >= j)
            sys(arr[i][j]);
        else
            sys(0);
    }
}
```

Lower Triangular Matrix -> Indexing

Row & Col

```
public class Solution {

    public static void printLowerTraingularMatrix(int arr[][],int m,int n){
        for(int i=0;i<m;i++){
            for(int j=0;j<n;j++){
                if(i>=j){
                    System.out.print(arr[i][j] + " ");
                }else{
                    System.out.print(0 + " ");
                }
            }
            System.out.println();
        }
    }

    public static void main(String[] args) {
        /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your cl:
        Scanner scn = new Scanner(System.in);
        int m = scn.nextInt();
        int n = scn.nextInt();

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

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

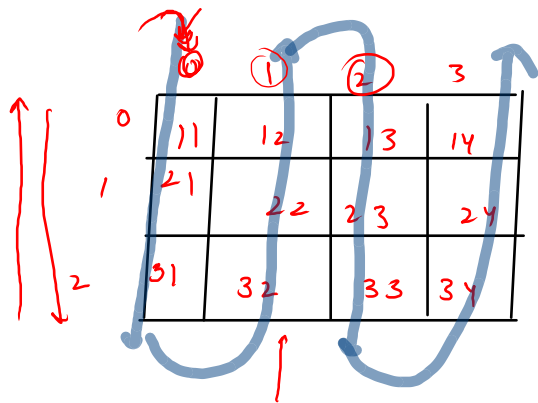
        printLowerTraingularMatrix(arr,m,n);
    }
}
```

---



3  
4  
11  
12  
13  
14  
21  
22  
23  
24  
31  
32  
33  
34

- ~~11~~
- ~~21~~
- ~~31~~
- ~~32~~
- ~~22~~
- ~~12~~
- ~~13~~
- ~~23~~
- ~~33~~
- ~~34~~
- ~~24~~
- ~~14~~



→ columnwise

```

for (int j=0; j<n; j++)
{
    if (j%2 == 0)
        for (int i=0; i<m; i++)
            else
                for (int i=m-1; i>=0; i--)
}

```

when col index  $\rightarrow$  even  $\rightarrow$  print from top to bottom.

if index is odd → print from bottom to top

```

1 public class Solution {
2     public static void printPattern(int arr[][],int m,int n){
3         for(int i=0;i<n;j++){
4             if(j%2==0) → even          0 - m
5             {
6                 for(int i=0;i<m;i++){
7                     System.out.println(arr[i][j]);
8                 }
9             }
10            else{ → odd                } m - 0
11            {
12                for(int i=m-1;i>=0;i--){
13                    System.out.println(arr[i][j]);
14                }
15            }
16        }
17    }
18
19    public static void main(String[] args) {
20        /* Enter your code here. Read input from STDIN. Print output to STDOUT
21        Scanner scn = new Scanner(System.in);
22        int m = scn.nextInt();
23        int n = scn.nextInt();
24
25        int arr[][] = new int[m][n];
26
27        for(int i=0;i<m;i++){
28            for(int j=0;j<n;j++){
29                arr[i][j] = scn.nextInt();
30            }
31        }
32
33        printPattern(arr,m,n);
34    }
35 }

```

