L12: Two Dimensional List Practice Questions

1-Tut : **Predict The Output**

Send Feedback

What will be the output of the following code?

```
li = [[1,2,3],[4,5,6],[7,8,9]]
print(li[2][1])
```

Options

5
8
2
Error

Correct Answer : B

2-Tut : **Predict The Output**

Send Feedback

What will be the output of the following code ?

```
li = [[1,2,3],[4,5,6],[7,8,9]]
print(li[1][3])
```

Options

6
9
3
Error

Correct Answer: D

Jagged Lists

3-Tut : **Predict The Output**

Send Feedback

What will be the output of the following code ?

```
li = [[1,2,3,4],[5,6],[7,8,9]]
print(li[2])
```

Options

[5,6]
Error
[7,8,9]
None of the above

Correct Answer : C

4-Tut : Predict The Output

Send Feedback

What will be the output of the following code?

```
li = [[1,2,3,4],[5,6],[7,8,9]]
print(li[1][3])
```

Options

4
6
Error
None of the above

Correct Answer : C

List Comprehension

5-Tut : Predict The Output

Send Feedback

What will be the output of the following code?

```
li = [ele**2 for ele in range(5)]

print(li)
```

Options

[1,2,3,4,5]
[1,4,9,16,25]
[0,1,4,9,16]
[0,1,4,9,16,25]

Correct Answer : C

6-Tut : Predict The Output

Send Feedback

What will be the output of the following code?

```
li = [ele**2 for ele in range(10) if ele%3 ==0]
print(li)
```

Options

[1,3,9]
[1,9,36,81]
[0,3,6,9]
[0,9,36,81]

Correct Answer : D

7-Tut : **Predict The Output**

Send Feedback

What will be the output of the following code?

```
li = [[ i*j for j in range(4)] for i in range(3)]
print(li)
```

Options

[[0, 0, 0, 0], [0, 1, 2, 3], [0, 2, 4, 6]]
[[0, 0, 0], [0, 1, 2], [0, 2, 4], [0, 3, 6]]
[[1, 2, 3, 4], [2, 4, 6, 8], [3, 6, 9, 12]]
None of above

Correct Answer : A

Input of 2d list

8-Tut : **Row Wise Sum**

Send Feedback

For a given two-dimensional integer array/list of size (N x M), find and print the sum of each of the row elements in a single line, separated by a single space.

Input Format :

The first line contains an Integer 't' which denotes the number of test cases or queries to be run. Then the test cases follow.
First line of each test case or query contains two integer values, 'N' and 'M', separated by a single space.
They represent the 'rows' and 'columns' respectively, for the two-dimensional array/list.
Second line onwards, the next 'N' lines or rows represent the ith row values.
Each of the ith row constitutes 'M' column values separated by a single space.

Output Format :

For each test case, print the sum of every ith row elements in a single line separated by a single space.
Output for every test case will be printed in a seperate line.

Constraints :

1 <= t <= 10^2
0 <= N <= 10^3
0 <= M <= 10^3
Time Limit: 1sec

Sample Input 1:

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

Sample Output 1:

```
3 7 11 15
```

Sample Input 2:

```
2
2 5
4 5 3 2 6
7 5 3 8 9
4 4
1 2 3 4
9 8 7 6
3 4 5 6
-1 1 -10 5
```

Sample Output 2:

```
20 32

10 30 18 -5
```

```
def rowWiseSum(mat, nRows, mCols):
    #Your code goes here
    for i in range(nRows):
        sum = 0
        for j in range(mCols):
            sum = sum + mat[i][j]
        print(sum,end=" ")
```

9-Tut : Predict The Output

Send Feedback

What will be the output of the following code?

```
li=[[1,2,3,4],[5,6,7,8],[9,10,11,12]]
for j in range(4):
    for ele in li:
        print(ele[j],end = " ")
```

Options

1 2 3 4 5 6 7 8 9 10 11 12
1 5 9 2 6 10 3 7 11 4 8 12
Error
1 5 9 10 6 2 3 7 11 12 8 4

Correct Answer : B

10-Tut : Largest Row or Column

Send Feedback

For a given two-dimensional integer array/list of size (N x M), you need to find out which row or column has the largest sum(sum of all the elements in a row/column) amongst all the rows and columns.

Note :

If there are more than one rows/columns with maximum sum, consider the row/column that comes first. And if ith row and the jth column have the same largest sum, consider the ith row as the answer.

Input Format :

The first line contains an Integer 't' which denotes the number of test cases or queries to be run. Then the test cases follow.
First line of each test case or query contains two integer values, 'N' and 'M', separated by a single space.
They represent the 'rows' and 'columns' respectively, for the two-dimensional array/list.
Second line onwards, the next 'N' lines or rows represent the ith row values.
Each of the ith row constitutes 'M' column values separated by a single space.

Output Format :

For each test case, If row sum is maximum, then print: "row" <row_index> <row_sum>
OR
If column sum is maximum, then print: "column" <col_index> <col_sum>
It will be printed in a single line separated by a single space between each piece of information.
Output for every test case will be printed in a seperate line.

Consider :

If there doesn't exist a sum at all then print "row 0 -2147483648", where -2147483648 or -2^31 is the smallest value for the range of Integer.

Constraints :

1 <= t <= 10^2
0 <= N <= 10^3
0 <= M <= 10^3
Time Limit: 1sec

Sample Input 1 :

```
1
2 2
1 1
1 1
```

Sample Output 1 :

```
row 0 2
```

Sample Input 2 :

```
2
3 3
3 6 9
1 4 7
2 8 9
4 2
1 2
90 100
3 40
-10 200
```

Sample Output 2 :

```
column 2 25

column 1 342
```

```
def largestRowCol(arr,nrows,mcols):
    rows = nrows
    cols = mcols
    if(rows == 0 or cols == 0):
        return ['row', 0, -2147483648]

    sumRow = [0] * rows
    sumCol = [0] * cols
    for i in range(rows):
        for j in range(cols):
            sumRow[i] += arr[i][j]
            sumCol[j] += arr[i][j] # Assume row 0 has maximum sum
    l = ['row', 0, sumRow[0]]
    for i in range(rows):
        if sumRow[i] > l[2]:
            l[2] = sumRow[i]
            l[1] = i
    for j in range(cols):
        if sumCol[j] > l[2]:
            l[2] = sumCol[j]
            l[1] = j
            l[0] = 'column'
    return l

def findLargest(arr, nRows, mCols):
    #Your code goes here
    l=largestRowCol(arr,nRows,mCols)
    print(*l)
```

11-Tut : Wave Print

Send Feedback

For a given two-dimensional integer array/list of size (N x M), print the array/list in a sine wave order, i.e, print the first column top to bottom, next column bottom to top and so on.

Input format :

The first line contains an Integer 't' which denotes the number of test cases or queries to be run. Then the test cases follow.
First line of each test case or query contains two integer values, 'N' and 'M', separated by a single space. They represent the 'rows' and 'columns' respectively, for the two-dimensional array/list.
Second line onwards, the next 'N' lines or rows represent the ith row values.
Each of the ith row constitutes 'M' column values separated by a single space.

Output format :

For each test case, print the elements of the two-dimensional array/list in the sine wave order in a single line, separated by a single space.
Output for every test case will be printed in a seperate line.

Constraints :

1 <= t <= 10^2
0 <= N <= 10^3
0 <= M <= 10^3
Time Limit: 1sec

Sample Input 1:

1
3 4
1 2 3 4
5 6 7 8
9 10 11 12

Sample Output 1:

1 5 9 10 6 2 3 7 11 12 8 4

Sample Input 2:

2
5 3
1 2 3
4 5 6
7 8 9
10 11 12
13 14 15
3 3
10 20 30
40 50 60
70 80 90

Sample Output 2:

1 4 7 10 13 14 11 8 5 2 3 6 9 12 15

10 40 70 80 50 20 30 60 90

```
def wavePrint(arr, nRows, mCols):
    #Your code goes here
    for j in range(mCols):
        if j % 2 == 0:
            for i in range(nRows):
                print(arr[i][j], end = " ")
        else:
            for i in range(nRows-1, -1, -1):
                print(arr[i][j], end = " ")
```

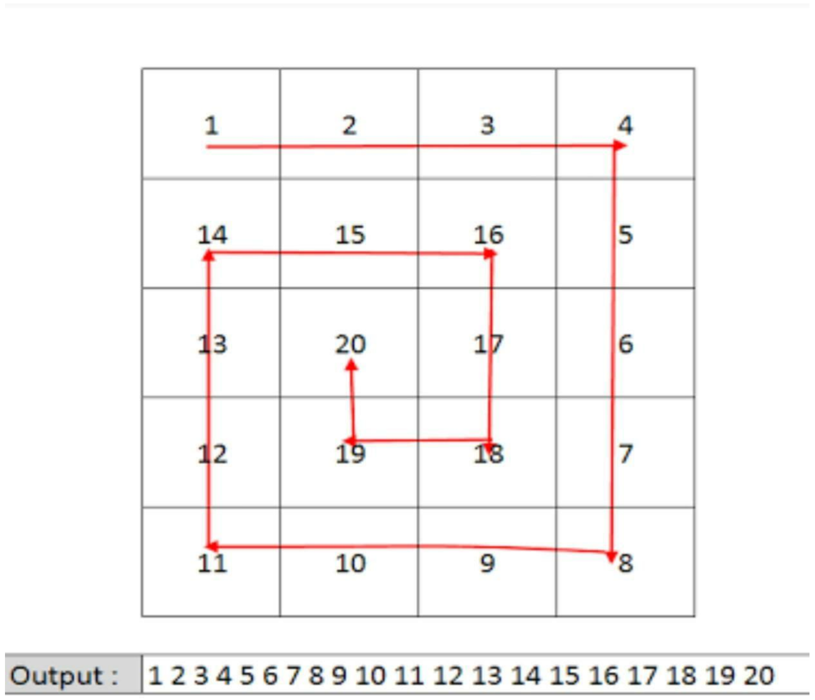
12-Tut : **Spiral Print**

Send Feedback

For a given two-dimensional integer array/list of size (N x M), print it in a spiral form. That is, you need to print in the order followed for every iteration:

a. First row(left to right)
b. Last column(top to bottom)
c. Last row(right to left)
d. First column(bottom to top)
 Mind that every element will be printed only once.

Refer to the Image:



Input format :

The first line contains an Integer 't' which denotes the number of test cases or queries to be run. Then the test cases follow.
First line of each test case or query contains two integer values, 'N' and 'M', separated by a single space. They represent the 'rows' and 'columns' respectively, for the two-dimensional array/list.
Second line onwards, the next 'N' lines or rows represent the ith row values.
Each of the ith row constitutes 'M' column values separated by a single space.

Output format :

For each test case, print the elements of the two-dimensional array/list in the spiral form in a single line, separated by a single space.
Output for every test case will be printed in a seperate line.

Constraints :

1 <= t <= 10^2
0 <= N <= 10^3
0 <= M <= 10^3
Time Limit: 1sec

Sample Input 1:

```
1
4 4
1 2 3 4
5 6 7 8
9 10 11 12
13 14 15 16
```

Sample Output 1:

```
1 2 3 4 8 12 16 15 14 13 9 5 6 7 11 10
```

Sample Input 2:

```
2
3 3
1 2 3
4 5 6
7 8 9
3 1
10
20
30
```

Sample Output 2:

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

10 20 30
```

```python
def spiralPrint(mat, nRows, mCols):
    #Your code goes here
    rs,re = 0,nRows-1
    cs,ce = 0,mCols-1
    total = nRows * mCols
    n = 1
    while(n <= total):
        for i in range(cs,ce+1):
            print(mat[rs][i],end= " ")
            n += 1
        rs += 1

        for i in range(rs,re+1):
            print(mat[i][ce],end = " ")
            n += 1
        ce = ce - 1

        for i in range(ce,cs-1,-1):
            print(mat[re][i],end = " ")
            n += 1
        re = re - 1

        for i in range(re,rs-1,-1):
            print(mat[i][cs],end = " ")
            n += 1
        cs = cs + 1
```