

TASK2 – 2D Array

Use functions for all the below questions (Deadline May 2nd)

1. Problem1 : Wiggly Path1

Given an input matrix , access the array elements in this wiggly path-I and print the array in the pattern given below

Input :

```
1  2  3  4  5
6  7  8  9 10
11 12 13 14 15
16 17 18 19 20
```

Output :

1 2 3 4 5 10 9 8 7 6 11 12 13 14 15 20 19 18 17 16

Input :

```
10 24 32
50  6 17
99 10 11
```

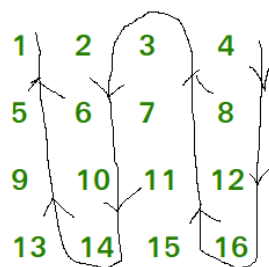
Output :

10 24 32 17 6 50 99 10 11

2. Problem 2: Wiggly Path II

Given an input matrix , access the array elements in this wiggly path-I and print the array in the pattern given below

Input



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

Input : 1 2 3 4

5 6 7 8

9 10 11 12

13 14 15 16

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

Input : 1 9 4 10

3 6 90 11

2 30 85 72

6 31 99 15

Output : 10 11 72 15 99 85 90 4 9 6 30 31 6 2 3 1

3. Problem3: Anti Diagonal Sorting

Given a matrix, sort the antidiagonal (top right to bottom left) elements of the matrix. The matrix has to be updated. Print the updated matrix.

Input:

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

Output:

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

4. Problem 4: Binary Matrix Transformation

You are given a binary matrix.. Your task is to predict the status of the matrix after 'K' iterations . Each iteration brings about a set of transformation to the matrix . Transformations are as follows.

- ☐ An element 1 in current matrix remains 1 in the next iteration only if it is surrounded by just A number of 1s (not more not less) in the previous iteration; where $0 \leq \text{limit1a} \leq A \leq \text{limit1b}$, otherwise reset to 0.
- ☐ An element 0 in current matrix becomes 1 in the next iteration only if it is surrounded by just B number of 1s (not more not less) in the previous iteration; where $0 \leq \text{limit0a} \leq B \leq \text{limit0b}$, otherwise remain 0..

Note that second transformation(k=2) is applied on the output of first transformation.(k=1).

.{For hints refer the video lecture}

Input:

$$\begin{bmatrix} 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

limit1a =2,limit1b=2,limit0a=2,limit0b=3,K=2.

Output after first iteration

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}$$

Output after second iteration

$$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

5. Problem 5: Task Assignment Problem

Let there be N workers and N jobs. Any worker can be assigned to perform any job, incurring some cost that may vary depending on the work-job assignment. It is required to perform all jobs by assigning exactly one worker to each job and exactly one job to each agent in such a way that the total cost of the assignment is minimized. {For hints refer the video lecture}

	Job 1	Job 2	Job 3	Job 4
A	9	2	7	8
B	6	4	3	7
C	5	8	1	8
D	7	6	9	4

Worker A takes 8 units of time to finish job 4.

An example job assignment problem. Green values show optimal job assignment that is A-Job2, B-Job1, C-Job3 and D-Job4