A = (4×5 matrix grid)

A[0][4]

A[1][3]

TC = O(1)

A[3][1]

(0,0)

A = (N×M matrix grid)

(0, M-1)

(N-1, M-1)

(N-1, 0)

N * M

SC = O(N*M)

Q → Print row wise sum.

A = (3×4 matrix)

	0	1	2	3
0	3	8	9	2
1	1	2	3	6
2	4	10	11	17

o/p
22
12
42
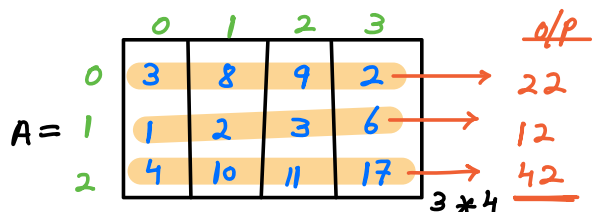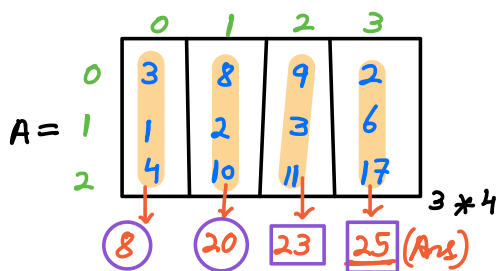
3 * 4

```
for i → 0 to (N-1)
    sum = 0
    for j → 0 to (M-1)
        sum += A[i][j]
    print (sum)
```

TC = O(N*M)
SC = O(1)

Q → Find max column sum.

A = (3×4 matrix)

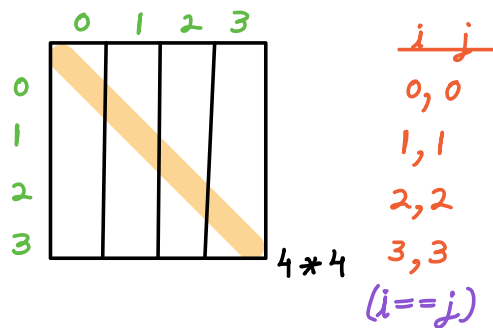|	0	1	2	3
0	3	8	9	2
1	1	2	3	6
2	4	10	11	17

8   20   23   25 (Ans)

3 * 4

```
ans = INT_MIN
for j → 0 to (M-1)
    sum = 0
    for i → 0 to (N-1)
        sum += A[i][j]
    ans = max(ans, sum)
return ans
```

N → Rows
M → Columns

j	i	
0	{0, 1, 2}	(A[0][0] + A[1][0] + A[2][0])
1	{0, 1, 2}	(A[0][1] + A[1][1] + A[2][1])

TC = O(N*M)
SC = O(1)

Q → Given a square matrix A[N][N].
Print main diagonal from → top left to bottom right

Matrix 4*4 with green column headers 0 1 2 3 and row indices 0 1 2 3, orange diagonal highlighted.

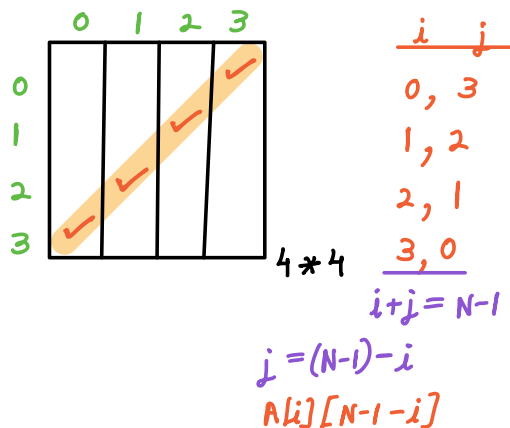| i	j
0, 0
1, 1
2, 2
3, 3

4 * 4

$(i == j)$

```
for i → 0 to (N-1)
    print (A[i][i])
```

$TC = O(N)$

$SC = \underline{O(1)}$

Q → Given a **square matrix** $A[N][N]$.
Print main diagonal from → top right to bottom left

i	j
0, 3
1, 2
2, 1
3, 0

4 * 4

$i + j = N-1$

$j = (N-1) - i$

$A[i][N-1-i]$

```
j = N-1
for i → 0 to (N-1)
    print (A[i][j])
    j -= 1
```

$TC = O(N)$

$SC = \underline{O(1)}$

$A[4][4]$

i	j
0	3
1	2
2	1
3	0
4	-1	→ stop

Q → Print all diagonals from right to left.

$A =$ matrix 3*4

	0	1	2	3
0	3	8	9	2
1	1	2	3	6
2	4	10	11	17

3 * 4

o/p

```
3
8   1
9   2   4
2   3   10
6   11
17
```

diagonals = $\underline{N + M - 1}$

$A =$ (3*4 matrix with diagonal arrows), ④ M, ③ N
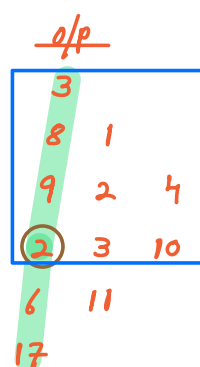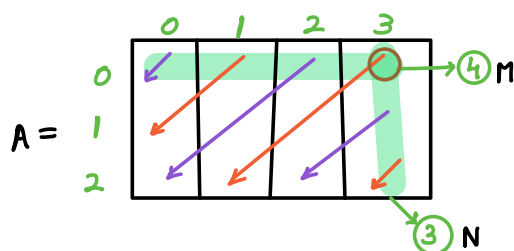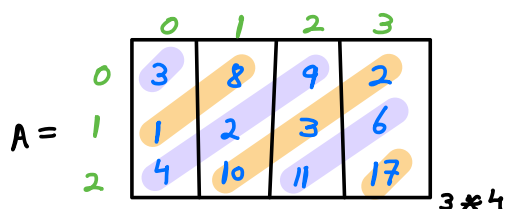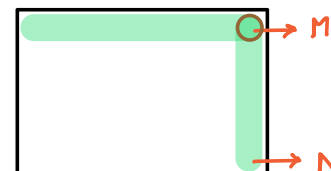
for k → 0 to (M-1)
    i = 0   j = k
    while (i < N && j >= 0)
        print (A[i][j])
        i += 1
        j -= 1
    print ("\n")

for k → 1 to (N-1)
    i = k   j = M-1
    while (i < N && j >= 0)
        print (A[i][j])
        i += 1
        j -= 1
    print ("\n")

$TC = O(N*M)$

$SC = O(1)$

10:30 PM

A =

	0	1	2	3
0	3	8	9	2
1	1	2	3	6
2	4	10	11	17

3 * 4

k	i	j
0	0	0
	1	-1
1	0	1
	1	0
	2	-1
2	0	2
	1	1
	2	0
	3	-1

k	i	j
3	0	3
	1	2
	2	1
	3	0
4		→ stop

k	i	j
1	1	3
	2	2
	3	1
2	2	3
	3	2
3		→ stop

Q → Given a __square matrix__ A[N][N]. Update the matrix to its __transpose__ without extra space.

A =

	0	1	2	3
0	1	2	3	4
1	5	6	7	8
2	9	10	11	12
3	13	14	15	16

→

	0	1	2	3
0	1	5	9	13
1	2	6	10	14
2	3	7	11	15
3	4	8	12	16

(0,2) → (2,0)
(2,0) → (0,2)

(3,2) → (2,3)
(2,3) → (3,2)

$(i,j) \longleftrightarrow (j,i)$

X

$i = 2$
$j = 3$ ✓
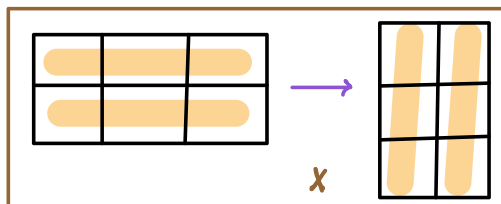
$(2,3) \longleftrightarrow (3,2)$ $i<j$ for $i \rightarrow 0$ to $(N-1)$
for $j \rightarrow (i+1)$ to $(N-1)$
$t = A[i][j]$
$A[i][j] = A[j][i]$ } swap$(A[i][j],$
$A[j][i] = t$ $A[j][i])$

x { $i = 3$
$j = 2$ } $(3,2) \longleftrightarrow (2,3)$ $i>j$
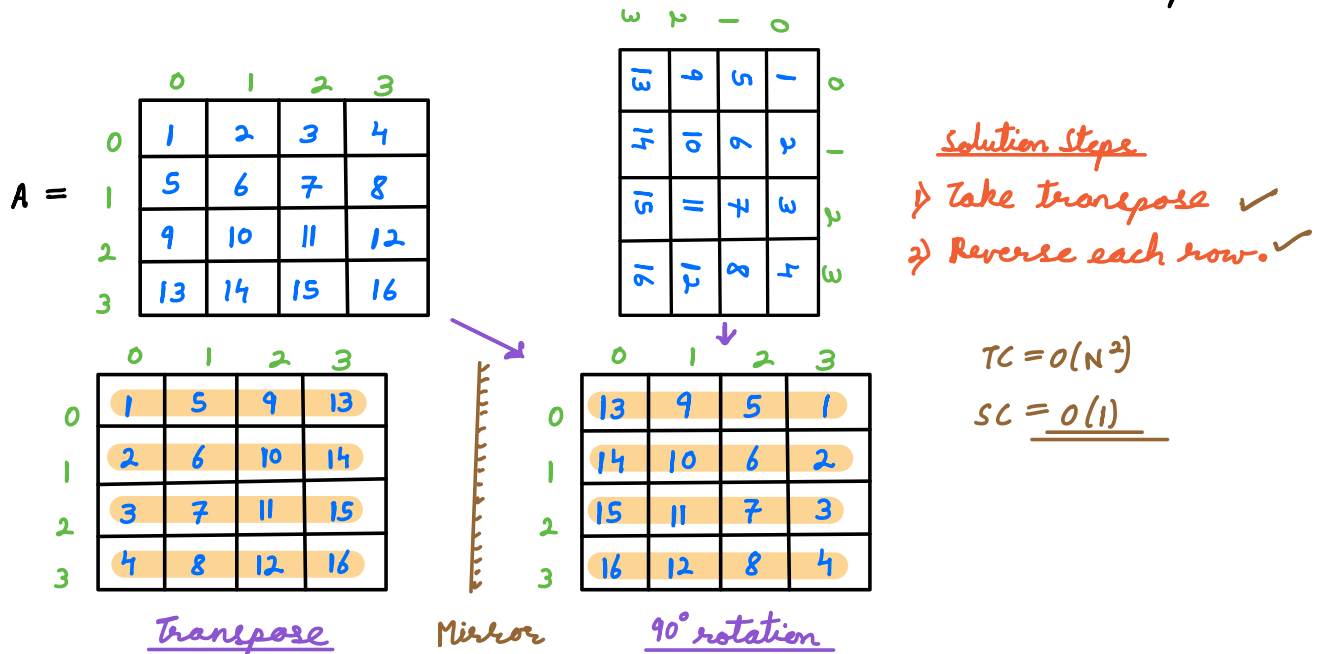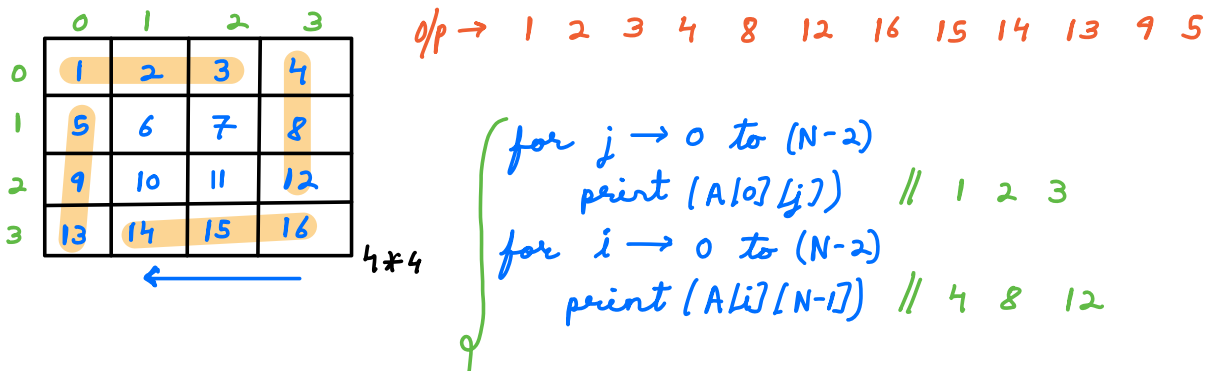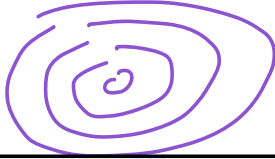
$TC = O(N^2)$
$SC = \underline{O(1)}$

Q→ Given a __square matrix__ A[N][N]. ==Update the matrix== to its __90° clockwise rotation__ without extra space.

A =

0	1	2	3
1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

__Solution Steps__
1) Take transpose ✓
2) Reverse each row. ✓

$TC = O(N^2)$
$SC = \underline{O(1)}$

Transpose

0	1	2	3
1	5	9	13
2	6	10	14
3	7	11	15
4	8	12	16

Mirror

90° rotation

0	1	2	3
13	9	5	1
14	10	6	2
15	11	7	3
16	12	8	4

Q→ Given a square matrix. Print __boundary elements__ in clockwise order starting from (0,0).

0	1	2	3
1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

4*4

o/p → 1 2 3 4 8 12 16 15 14 13 9 5

for $j \rightarrow 0$ to $(N-2)$
  print $(A[0][j])$ // 1 2 3
for $i \rightarrow 0$ to $(N-2)$
  print $(A[i][N-1])$ // 4 8 12

$TC = O(4*(N-1))$

$= \underline{O(N)}$

$SC = \underline{O(1)}$

for $j \rightarrow (N-1)$ to 1
    print $(A[N-1][j])$ // 16   15   14

for $i \rightarrow (N-1)$ to 1
    print $(A[i][0])$ // 13   9   5

$3 * 5$

$N = 3$

$M = 5$

$\underline{N + M - 1}$