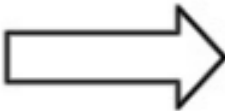


Rotate Matrix

You are given an $n \times n$ 2D matrix representing an image, rotate the image by 90 degrees (clockwise).

You have to rotate the image in-place, which means you have to modify the input 2D matrix directly. **DO NOT** allocate another 2D matrix and do the rotation.

Example 1:

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

1	2
0, 0	0, 1
4	5
1, 0	1, 1
7	8
2, 0	2, 1

3
0, 2
6
1, 2
9
2, 2

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

i, j
 $0, 0 \rightarrow 0, 2$

$0, 1 \rightarrow 1, 2$

$0, 2 \rightarrow 2, 2$

i
 $0, 0 \rightarrow 10, 0$
 $2, 1 \rightarrow 01, 0$
 $2, 2 \rightarrow 2, 0$

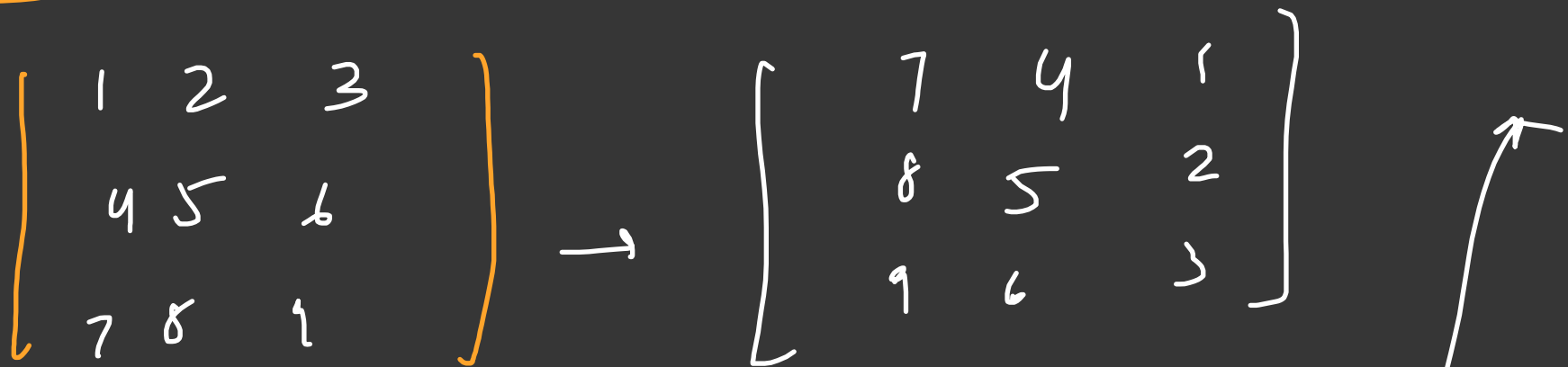
3
 $0 \rightarrow 2$
 $1 \rightarrow 1$
 $2 \rightarrow 0$

2×0
 $3 - 1$
 $(n-1) \times i$
 $n-1$
 $2 - 2 = 0$

Brute force approach:

~~and~~ Create a dummy matrix and map the value of arr to dummy matrix.

Pseudo code:



```
for (i=0; i<n; i++)
```

```
    for (j=0; j<n; j++)
```

```
        arr[i][j] = arr[j][(n-1)-i];
```

Optimal Approach:

- Transpose the matrix

1	2	3
4	5	6
7	8	9

→
Transpose

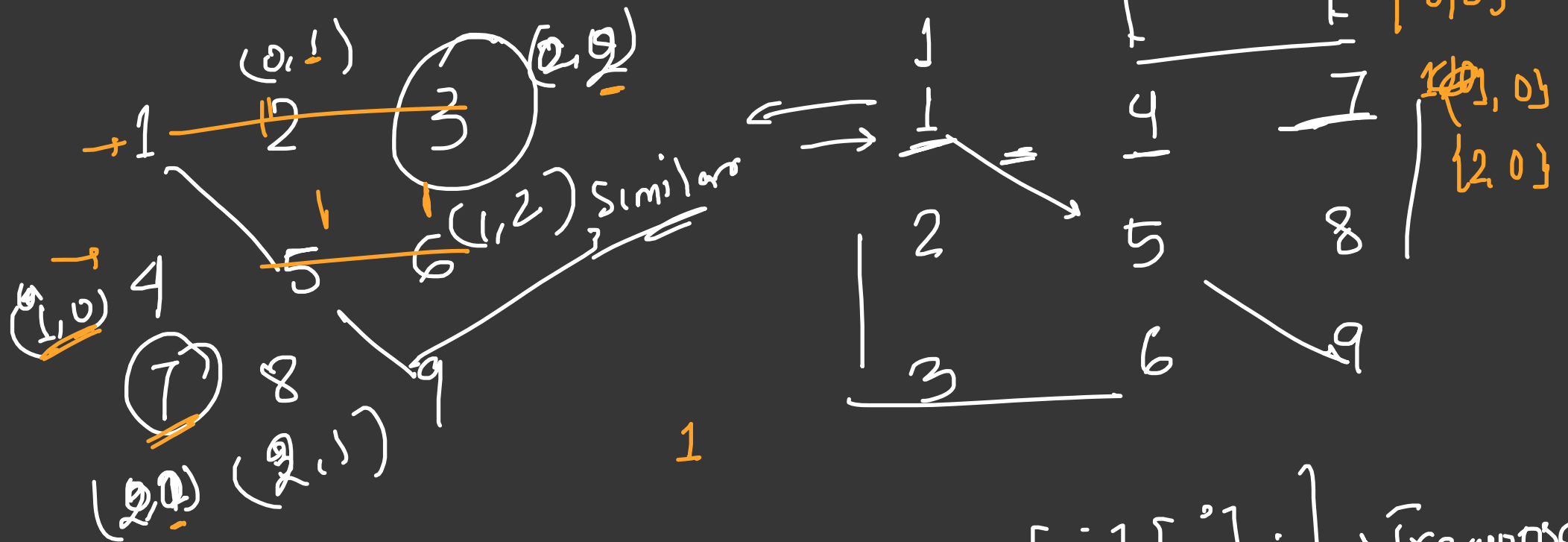
1	4	7
2	5	8
3	6	9

↓ Reverse the every row

ans

7	4	1
8	5	2
9	6	3

How do we Transpose the matrix



$[arr[i][j] = arr[j][i];] \rightarrow \text{Transpose}$

for ($i = 0; i < n-1; i++$)

for ($j = i+1; j < m; j++$)