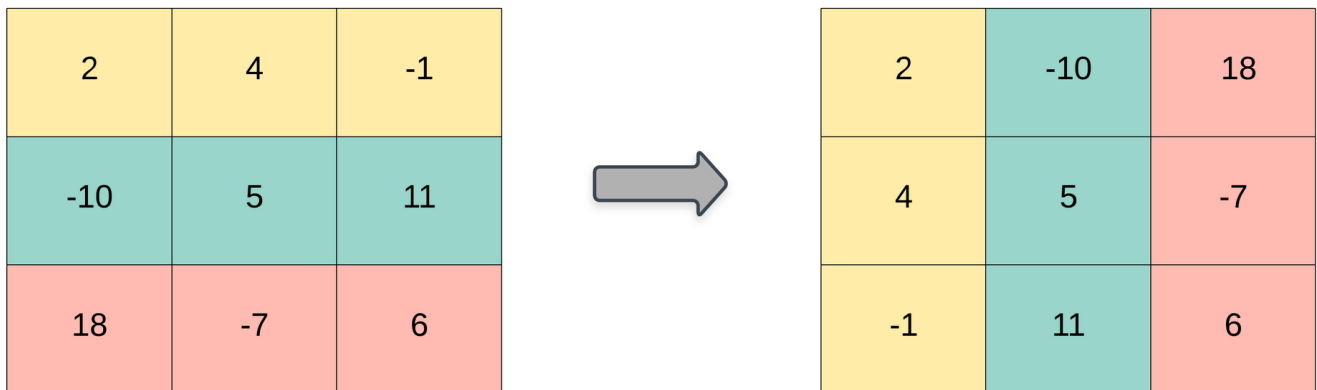


867. Transpose Matrix

Given a 2D integer array `matrix`, return *the transpose of* `matrix`.

The **transpose** of a matrix is the matrix flipped over its main diagonal, switching the matrix's row and column indices.



Example 1:

Input: `matrix = [[1,2,3],[4,5,6],[7,8,9]]`

Output: `[[1,4,7],[2,5,8],[3,6,9]]`

Example 2:

Input: `matrix = [[1,2,3],[4,5,6]]`

Output: `[[1,4],[2,5],[3,6]]`

Constraints:

- `m == matrix.length`
- `n == matrix[i].length`
- `1 <= m, n <= 1000`
- `1 <= m * n <= 105`
- `-109 <= matrix[i][j] <= 109`
- ```
class Solution:
 def transpose(self, matrix: List[List[int]]) -> List[List[int]]:
 ans = [[0]*len(matrix) for _ in range(len(matrix[0]))]
 for i in range(len(matrix)):
 for j in range(len(matrix[0])):
 val = matrix[i][j]
```

```
 ans[j][i] = val
 return ans
```

- For NXN inplace

```
def transposeNXN(matrix):
 for i in range(len(matrix)):
 for j in range(i):
 matrix[i][j], matrix[j][i] = matrix[j][i], matrix[i][j]

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