

- Given a 2D array of type T, return the transpose of the matrix.
- The transposition is the matrix flipped over its main diagonal, switching row and column indices.

$\begin{bmatrix} 2 & 4 & -1 \\ -10 & 5 & 11 \\ 18 & -7 & 6 \end{bmatrix} \rightarrow \begin{bmatrix} 2 & -10 & 18 \\ 4 & 5 & -7 \\ -1 & 11 & 6 \end{bmatrix}$

$\begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 5 \\ 2 & 6 \\ 3 & 7 \\ 4 & 8 \end{bmatrix}$

```

for (i = m.size() - 1; i > 0; --i)
    for (j = m.size() - 1; j < i; ++j)
        swap(m[i][j], m[j][i])

```

Only if square?
 No. Because $j < i$
 Assumed square.

↓ →

nm: list[list[T]] (m[0].size(), m.size())

```

for i = 0 → m.size()
    for j = i → m[i].size()
        nm[j][i] = m[i][j]

```

Steps:

1. Create 2D list in which #rows = #columns and #cols = #rows of input matrix.
2. Enumerate over all values in input matrix.
Value @ [row][col] becomes value @ [col][row] for transposed matrix.
3. Return the matrix.