**Rotate by 90 degree**

[matrix](http://www.practice.geeksforgeeks.org/tag-page.php?tag=matrix&isCmp=0)

[Microsoft](http://www.practice.geeksforgeeks.org/tag-page.php?tag=%20Microsoft&isCmp=1)

Given an square matrix, turn it by 90 degrees in anti-clockwise direction without using any extra space.

**Input:**

The first line of input contains a single integer T denoting the number of test cases. ThenT test cases follow. Each test case consist of two lines. The first line of each test case consists of an integer N, where N is the size of the square matrix.The second line of each test case contains NxN space separated values of the matrix M.

**Output:**

Corresponding to each test case, in a new line, print the rotated array.

**Constraints:**

1 ≤ T ≤ 50  
1 ≤ N ≤ 50  
1 ≤ A[i] ≤ 100  
  
**Example:**

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

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

\*\*For More Examples Use Expected Output\*\*

<http://www.practice.geeksforgeeks.org/problem-page.php?pid=1311>

#include <stdio.h>

#include <math.h>

#include <iostream>

#include <cmath>

#include <vector>

#define ll long long int

using namespace std;

int main()

{

int t;

scanf("%d",&t);

while(t--) {

int n;

scanf("%d",&n);

int arr[n\*n];

for(int i =0; i<n\*n; i++) {

scanf("%d", &arr[i]);

}

int index=0;

int m[n][n];

for(int i =0; i<n; i++) {

for(int j=0; j<n; j++) {

m[i][j] = arr[index++];

}

}

/\*for(int i =0; i<n; i++) {

for(int j=0; j<n; j++) {

printf("%d ", m[i][j]);

}

printf("\n");

} \*/

for(int col=n-1; col >=0; col--) {

for(int fila =0; fila < n; fila++) {

printf("%d ", m[fila][col]);

}

}

printf("\n");

}

return 0;

}