

Arrays 2D - Transpose Matrix

Problem

Submissions

Leaderboard

Discussions

Sheldon Cooper has a square-shaped puzzle made up of small square pieces containing numbers on them. He wants to rearrange the puzzle by changing the elements of a row into a column element and the column element into a row element without altering the shape of the puzzle. Help Sheldon solves this puzzle. Write a program to find the transpose of the given 2D matrix.

Input Format

The first line consists of an integer which represents the number of rows and columns. The next line consists of the elements in the matrix.

Constraints

NA

Output Format

Output prints the transpose of the input matrix.

Sample Input 0

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

Sample Output 0

```
1 2 3
4 5 6
7 8 9
Transpose matrix is:
1 4 7
2 5 8
3 6 9
```

Sample Input 1

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

Sample Output 1

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

```
9 10 11 12
13 14 15 16
Transpose matrix is:
1 5 9 13
2 6 10 14
3 7 11 15
4 8 12 16
```

[f](#) [t](#) [in](#)

Submissions: 574

Max Score: 100

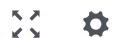
Difficulty: Medium

Rate This Challenge:

☆☆☆☆☆

[More](#)

C



```
1 #include <stdio.h>
2 #include <string.h>
3 #include <math.h>
4 #include <stdlib.h>
5
6 int main ()
7 {
8     int n,i,j;
9     scanf("%d",&n);
10    int arr[n][n];
11    for (i=0;i<n;i++)
12    {
13        for (j=0;j<n;j++)
14        {
15            scanf("%d",&arr[i][j]);
16        }
17    }
18
19    for(i=0;i<n;i++)
20    {
21        for (j=0;j<n;j++)
22        {
23            printf("%d ",arr[i][j]);
24        }
25        printf("\n");
26    }
27
28    printf("Transpose matrix is:\n");
29    for(i=0;i<n;i++)
30    {
31        for (j=0;j<n;j++)
32        {
33            printf("%d ",arr[j][i]);
34        }
35        printf("\n");
36    }
37    return 0;
38 }
39
```

Line: 1 Col: 1

[Upload Code as File](#) ☐ [Test against custom input](#)

[Run Code](#)

[Submit Code](#)

Testcase 0 

Testcase 1 

Congratulations, you passed the sample test case.

Click the **Submit Code** button to run your code against all the test cases.

Input (stdin)

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

Your Output (stdout)

```
1 2 3
4 5 6
7 8 9
Transpose matrix is:
1 4 7
2 5 8
3 6 9
```

Expected Output

```
1 2 3
4 5 6
7 8 9
Transpose matrix is:
1 4 7
2 5 8
3 6 9{-truncated-}
```