

[www.codechef.com/ide](http://www.codechef.com/ide)

C (gcc 6.2)



Code gets auto-saved every

**CODECHEF**

An Educational Initiative

Have a 40%

## Code, Compile & Run

Add × +

Context Code/Name (e.g., JUDGE/PROBNAME)

Select

Open File

Custom Input

3 3

11 12 13 14 15 16 17 18 19

Status Successfully executed Date 2020-05-08 07

Input

3 3

11 12 13 14 15 16 17 18 19

Output

10

19

Transpose of the matrix

11 14 17

12 15 18

13 16 19

```
1 /*program to obtain the transpose of the matrix*/
2 #include <stdio.h>
3 int main ()
4 {
5     int arr1[10][10], arr2[10][10], i,j,n, m;
6     printf ("enter number of rows in matrix\n");
7     scanf ("%d", &n) ;
8     printf ("enter number of col umns in matrix\n");
9     scanf ("%d", &m);
10    printf ("enter the elements of the matrix: \n");
11    for (i=0; i<m; i++)
12        for (j=0; j<n; j++)
13            scanf ("%d", &arr1[i][j]);
14    printf ("elements of matrix are: \n");
15    for (i=0; i<m; i++)
16    {
17        for (j=0;j<n; j++)
18        {
19            printf ("%d\t", arr1[i][j]);
20            printf ("\n");
21        }
22    }
23    printf ("Transpose of the matrix \n");
24    for (i=0;i<n; i++)
25    {
26        for (j=0;j<m; j++)
27        {
28            arr2 [i][j]=arr1[j][i];
29            printf ("%d\t",arr2[i][j]);
30        }
31        printf ("\n");
32    }
33 }
```

# C program to implement transpose of a matrix:

## Algorithm:

- Step 1: Start
- Step 2: input  $r, c$
- Step 3: Display enter matrix elements.  
for ( $i=0; i < r; i++$ )  
for ( $j=0; j < c; j++$ )  
o/p  $a[i][j]$   
End for  
End for
- Step 4: Print entered matrix
- Step 5: o/p  $a[i][j]$   
if ( $j == c-1$ )  
output "\n"
- Step 6: for ( $i=0; i < r; i++$ )  
for ( $j=0; j < c; j++$ )  
transpose  $[i][j] = a[i][j]$
- Step 7: Display "Transpose of matrix"  
Repeat for ( $i=0; i < c; i++$ )  
Repeat for ( $j=0; j < r; j++$ )  
o/p transpose  $[i][j]$   
if ( $j == r-1$ )  
o/p ~~transpose~~ "." "\n"
- Step 8: Stop.

## Flowchart:



