

ICT Application- Transpose of Matrix using ‘C’ Language Tool

P.R. Kolhe, P P Kolhe, S D Desai, A Bowalekar
Officer Incharge, AKMU, Dr. BSKKV, Dapoli, India.

Abstract – In mathematics to solve matrix problems, manually it is very difficult task to get the required output need to perform. Matrix transpose means, write the columns of A as the rows of A^T or write the columns of A as the rows of A^T .

In the era of Information Communication Technology (ICT) .The ICT programming technique, it is easier task. One of the very popular programs in C programming is Matrix Transpose. Transposes mean change row to column. This paper discusses Matrix Transpose in C language, source code and methods with outputs. The source codes of program for Matrix Transpose in C programming are to be compiled. Running them on Turbo C or available version and other platforms might require a few modifications to the code. You probably know how to multiply two matrices.

Index Terms – Matrix ,ICT , C lang. turboc , Matrix transpose ,T .

1. INTRODUCTION

1.1. Introduction to Matrix transpose

A matrix is defined as a rectangular arrangement of numbers in the form of rows and columns. When the elements in rows and columns of a matrix are exchanged, a new matrix is formed which is known as transpose of the matrix. It carries a great significance in structural dynamics calculation, electrical engineering, mathematics and physics.

The transpose of a matrix was introduced in 1858 by the British mathematician Arthur Cayley.

In linear algebra, the of a matrix A is another matrix A^T (also written A' , A^r , tA or A^t) created by any one of the following equivalent actions:

- I. Reflect A over its main diagonal (which runs from top-left to bottom-right) to obtain A^T
- II. write the rows of A as the columns of A^T
- III. write the columns of A as the rows of A^T

Formally, the i th row, j th column element of A^T is the j th row, i th column element of A:

If A is an $m \times n$ matrix then A^T is an $n \times m$ matrix.

Transpose of a matrix is a new matrix whose rows are the columns of the original. (This makes the columns of the new matrix the rows of the original). Here is a matrix and its transpose

$$\begin{pmatrix} 5 & 4 & 3 \\ 4 & 0 & 4 \\ 7 & 10 & 3 \end{pmatrix}^T = \begin{pmatrix} 5 & 4 & 7 \\ 4 & 0 & 10 \\ 3 & 4 & 3 \end{pmatrix}$$

A matrix is defined as a rectangular arrangement of numbers in the form of rows and columns. When the elements in rows and columns of a matrix are exchanged, a new matrix is formed which is known as transpose of the matrix. It carries a great significance in structural dynamics calculation, electrical engineering, mathematics and physics.

Finding the transpose of a matrix in C is a popular tutorial under “array”. It basically gives the idea of matrix inputting, manipulating and outputting using the standard input/output functions of the C language.

The basic logic behind matrix transposition is swapping the elements of row and respective column. After entering all the elements of the matrix, the number of rows and columns are swapped and a loop is started to exchange the elements. The exchanged or newly formed elements are stored as the elements of transposed matrix.

2. ALGORITHM TO TRANSPOSE A MATRIX

- Start
- Declare all the necessary variables
- Enter the order of matrix
- Enter the elements of matrix row-wise using loop
- Display the entered matrix in standard format (it is not a compulsory step)
- Assign number of rows with number of column
- Swap $(i, j)^{\text{th}}$ element with $(j, i)^{\text{th}}$
- Store the new elements as element of transposed matrix
- Print the elements of transpose matrix in format using loop
- Stop

3. SOURCE CODE FOR MATRIX TRANSPORT

```
#include <stdio.h>
```

```

#include<math.h>
#include<conio.h>
int main()
{
    int a, b, c, d, mat[10][10], trans[10][10]; //declaration of
variable
    printf(" Enter the number of rows and columns of matrix: ");
    scanf("%d%d",&a,&b); // inputting order of matrix
    printf(" Enter the elements of matrix \n");
    for( c = 0 ; c < a ; c++ )// loop to input the matrix
    {
        for( d = 0 ; d < b ; d++ )
        {
            scanf("%d",&mat[c][d]);
        }
    }
    for( c = 0 ; c < a ; c++ ) // loop to transpose the matrix
    {
        for( d = 0 ; d < b ; d++ )
        {
            trans[d][c] = mat[c][d];
        }
    }
    printf(" The traspose of entered matrix is:-\n");
    for( c = 0 ; c < b ; c++ ) // loop for printing transposed matrix
    {
        for( d = 0 ; d < a ; d++ )

```

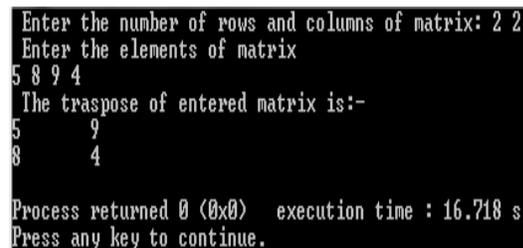
```

    {
        printf("%d\t",trans[c][d]);
    }
    printf("\n");
}
return 0;
}

```

4. OUTPUT FOR MATRIX TRANSPORT

When C program is executed, it asks for the number of rows and columns of the matrix to be transposed. Then, the elements of matrix need to be entered row wise. Finally, the transposed matrix is displayed as console output in standard matrix format fig-1.



```

Enter the number of rows and columns of matrix: 2 2
Enter the elements of matrix
5 8 9 4
The traspose of entered matrix is:-
5      9
8      4

Process returned 0 (0x0)   execution time : 16.718 s
Press any key to continue.

```

Fig-1

REFERENCES

- [1] E Balgruhaswami "Programming in C"
- [2] Y .Kanetkar " Let's C
- [3] Programming with C Schaum's Outlines Series
- [4] C: the Complete Reference Paperback – 2000 by Herbert Schildt (Author)
- [5] Programming and Problem Solving Through "C" Language Paperback – 2016 by Harsha Priya (Author), R. Ranjeet (Author)
- [6] <http://www.cquestions.com>
- [7] <http://www.programmingsimplified.com/>