***Transpose of a Matrix***

To find the transpose of a matrix in C++, you can use nested loops. Here's one way to do it:

C++Java

// Java Program to find

// transpose of a matrix

class GFG

{

static final int N = 4;

// This function stores transpose

// of A[][] in B[][]

static void transpose(int A[][], int B[][])

{

int i, j;

for (i = 0; i < N; i++)

for (j = 0; j < N; j++)

B[i][j] = A[j][i];

}

// Driver code

public static void main (String[] args)

{

int A[][] = { {1, 1, 1, 1},

{2, 2, 2, 2},

{3, 3, 3, 3},

{4, 4, 4, 4}};

int B[][] = new int[N][N], i, j;

transpose(A, B);

System.out.print("Result matrix is \n");

for (i = 0; i < N; i++)

{

for (j = 0; j < N; j++)

System.out.print(B[i][j] + " ");

System.out.print("\n");

}

}

}

// This code is contributed by Anant Agarwal.

Here, the outer loop iterates through the columns of the matrix, and the inner loop iterates through the rows. For each element in the matrix, we swap the row and column indices to get the corresponding element in the transpose.

Here's a step-by-step explanation of the code:

* The program declares two integer variables, **rows** and **cols**, to store the number of rows and columns in the matrix, respectively.
* The program then prompts the user to enter the number of rows and columns of the matrix, and uses the **cin** function to read the values into the **rows** and **cols** variables.
* The program then creates an array of **int** with **rows** number of rows and **cols** number of columns, named **matrix**.
* Next, the program uses a nested for loop to prompt the user to enter the elements of the matrix. The outer loop iterates through the rows and the inner loop iterates through the columns. At each iteration, the program uses the **cin** function to read the current element into the **matrix** array.
* The program then prints the original matrix using another nested for loop. The outer loop iterates through the rows and the inner loop iterates through the columns. At each iteration, the program uses the **cout** function to print the current element of the **matrix** array.
* Finally, the program prints the transpose of the matrix using another nested for loop. This time the outer loop iterates through the columns, and the inner loop iterates through the rows. The program uses **matrix[j][i]** to get the corresponding element of the transpose because, in a transpose, the rows become columns and vice-versa.