A - Product of 2 Matrices

Given 2 matrices, find the product.

**Input Format**

First line of input contains T - number of test cases. First line of each test case contains N1, M1 - size of the 1st matrix. Its followed by N1 lines each containing M1 intergers - elements of the 1st matrix. The next line contains N2, M2 - size of the 2nd matrix. Its followed by N2 lines each containing M2 intergers - elements of the 2nd matrix. Note that M1 = N2.

**Constraints**

1 <= T <= 100  
1 <= N1,M1,N2,M2 <= 50  
-100 <= mat[i][j] <= 100

**Output Format**

For each test case, print the resultant product matrix, separated by newline.

**Sample Input 0**

2

2 2

1 2

3 -1

2 3

1 -2 3

2 3 -1

2 3

27 29 53

-28 49 -24

3 4

23 52 -38 72

-64 15 -59 -10

-75 43 10 25

**Sample Output 0**

5 4 1

1 -9 10

-5210 4118 -2207 2979

-1980 -1753 -2067 -3106

#include <iostream>

#include <vector>

using namespace *std*;

void fill2DMatrix(*vector*<*vector*<int>>& matrix)

{

int ele;

int rows = matrix.*size*(), cols = matrix[0].*size*();

for (auto i = 0; i < rows; i++)

{

for (auto j = 0; j < cols; j++)

{

*cin* >> ele;

matrix[i][j] = ele;

}

}

}

void display2DMatrix(*vector*<*vector*<int>>& matrix)

{

for (auto i : matrix)

{

for (auto j : i)

{

*cout* << j << " ";

}

*cout* << *endl*;

}

}

*vector*<*vector*<int>> productOfMatrix(*vector*<*vector*<int>>&m1, *vector*<*vector*<int>>&m2)

{

int rows1 = m1.*size*();

int cols1 = m1[0].*size*();

int rows2 = m2.*size*();

int cols2 = m2[0].*size*();

*vector*<*vector*<int>> res(rows1, *vector*<int>(cols2));

for (auto i = 0; i < rows1; i++)

{

for (auto j = 0; j < cols2; j++)

{

res[i][j] = 0;

for (auto k = 0; k < rows2; k++)

{

res[i][j] += m1[i][k] \* m2[k][j];

}

}

}

return res;

}

int main()

{

int t; *cin* >> t;

while (t--)

{

int n1, m1;

*cin* >> n1 >> m1;

*vector*<*vector*<int>> matrix1(n1, *vector*<int>(m1));

fill2DMatrix(matrix1);

int n2, m2;

*cin* >> n2 >> m2;

*vector*<*vector*<int>> matrix2(n2, *vector*<int>(m2));

fill2DMatrix(matrix2);

auto res = productOfMatrix(matrix1, matrix2);

display2DMatrix(res);

}

return 0;

}