

Q3 Implement Matrix Multiplication. Analyze their time complexities.

TIME COMPLEXITY: $O(a * b * c)$

SPACE COMPLEXITY: $O(a \times b \times c)$

where $(a \times b)$ and $(b \times c)$ are the dimensions.

```
#include <iostream>
#include <vector>
#include <random>
#include <chrono>

using namespace std;

vector<vector<int>> multiply(vector<vector<int>> & matrix1,
    vector<vector<int>> & matrix2)
{
    int a = int(matrix1.size()), b = int(matrix1[0].size()), c =
        int(matrix2.size()), d = int(matrix2[0].size());
    if(b != c)
        return {};
    vector<vector<int>> result(a, vector<int>(d, 0));
    for(int i = 0; i < a; ++i)
        for(int k = 0; k < d; ++k)
            for(int j = 0; j < b; ++j)
                result[i][k] += matrix1[i][j] * matrix2[j][k];
    return result;
}

int main()
{
    vector<vector<int>> matrix1 = {
        {1, 2, 3},
        {4, 5, 6},
        {7, 8, 9}
    };
    cout << "MATRIX 1: " << endl;
    for (auto & V : matrix1)
    {
        for(int & v : V)
            cout << v << " ";
        cout << endl;
    }
    vector<vector<int>> matrix2 = {
        {1, 0, 0},
        {0, 1, 0},
        {0, 0, 1}
    };
    cout << "MATRIX 2: " << endl;
    for (auto & V : matrix2)
    {
        for(int & v : V)
```

```
        cout << v << " ";
    cout << endl;
}
vector<vector<int>> result = multiply(matrix1, matrix2);
cout << "MATRIX1 * MATRIX2: \n";
for (auto & V : result)
{
    for(int & v : V)
        cout << v << " ";
    cout << endl;
}
}
```

OUTPUT:

MATRIX 1:

1 2 3

4 5 6

7 8 9

MATRIX 2:

1 0 0

0 1 0

0 0 1

MATRIX1 * MATRIX2:

1 2 3

4 5 6

7 8 9

Program ended with exit code: 0