



Code :-

```
#include <bits/stdc++.h>
using namespace std;

int matrixMultiplication(vector<int>& arr) {
    int n = arr.size();
    vector<vector<int>> dp(n, vector<int>(n, 0));

    for (int len = 2; len < n; len++) {
        for (int i = 0; i < n - len; i++) {
            int j = i + len;
            dp[i][j] = INT_MAX;
            for (int k = i + 1; k < j; k++) {
                int cost = dp[i][k] + dp[k][j] + arr[i] * arr[k] * arr[j];
                dp[i][j] = min(dp[i][j], cost);
            }
        }
    }

    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            cout << setw(4) << dp[i][j] << " ";
        }
        cout << endl;
    }
    return dp[0][n - 1];
}

int main() {
    vector<int> arr;
    int n;
    cout << "Enter the Number of Matrix :- ";
    cin >> n;

    for( int i = 0; i < n; i++ ) {
        int row, column;
        cout << "Enter the Number of Rows and Columns for Matrix " << i + 1 << " :- ";
        cin >> row >> column;
    }
}
```



**Subject: Design and Analysis
of Algorithms (01CT0512)**

Aim: Implementing Matrix Chain Multiplication using Dynamic
Programming Approach

Experiment No: 10

Date: 13\09\2025

Enrollment No: 92301733054

```
if (i == 0) {  
    arr.push_back(row);  
    arr.push_back(column);  
}  
else {  
    arr.push_back(column);  
}  
}  
cout << matrixMultiplication(arr);  
return 0;  
}
```

Output :-

Enter the Number of Matrix :- 3

Enter the Number of Rows and Columns for Matrix 1 :- 2

2

Enter the Number of Rows and Columns for Matrix 2 :- 3

3

Enter the Number of Rows and Columns for Matrix 3 :- 3

6

0	0	12	48
0	0	0	36
0	0	0	0
0	0	0	0

48