

CSE2012

DAA LAB

Name: Preyash

Registration Number: 20BPS1022

Ex3: Matrix Multiplication

1. Matrix Multiplication

Code Window:

```
1  #include <iostream>
2  using namespace std;
3  #include <vector>
4  void read_matrix(vector<vector<int>> &mat, int n)
5  {
6      int ele;
7      for (int i = 0; i < n; i++)
8      {
9          vector<int> row;
10         for (int j = 0; j < n; j++)
11         {
12             cin >> ele;
13             row.push_back(ele);
14         }
15         mat.push_back(row);
16     }
17 }
18 void print_matrix(vector<vector<int>> mat)
19 {
20     int n = mat.size(), i, j;
21     for (i = 0; i < n; i++)
22     {
23         for (j = 0; j < n; j++)
24             cout << mat[i][j] << " ";
25         cout << "\n";
26     }
27 }
28
```

```

29 void matrix_multiply(vector<vector<int>> &mat1, vector<vector<int>> &mat2, vector<vector<int>> &mat3)
30 {
31     int i, j, k, n;
32     n = mat1.size();
33     for (i = 0; i < n; i++)
34     {
35         for (j = 0; j < n; j++)
36         {
37             mat3[i][j] = 0;
38             for (k = 0; k < n; k++)
39             {
40                 mat3[i][j] += mat1[i][k] * mat2[k][j];
41             }
42         }
43     }
44 }
45 int main()
46 {
47     vector<vector<int>> mat1, mat2;
48     int i, j, n;
49     cin >> n;
50     read_matrix(mat1, n);
51     read_matrix(mat2, n);
52     vector<vector<int>> mat3(n, vector<int>(n, 0));
53     matrix_multiply(mat1, mat2, mat3);
54     print_matrix(mat3);
55     return 0;
56 }

```

Code:

```
#include <iostream>
```

```
using namespace std;
```

```
#include <vector>
```

```
void read_matrix(vector<vector<int>> &mat, int n)
```

```
{
```

```
    int ele;
```

```
    for (int i = 0; i < n; i++)
```

```
    {
```

```
        vector<int> row;
```

```
        for (int j = 0; j < n; j++)
```

```
        {
```

```
            cin >> ele;
```

```
            row.push_back(ele);
```

```
        }
```

```
        mat.push_back(row);
```

```
    }
```

```
}
```

```
void print_matrix(vector<vector<int>> mat)
```

```
{  
    int n = mat.size(), i, j;  
    for (i = 0; i < n; i++)  
    {  
        for (j = 0; j < n; j++)  
            cout << mat[i][j] << " ";  
        cout << "\n";  
    }  
}
```

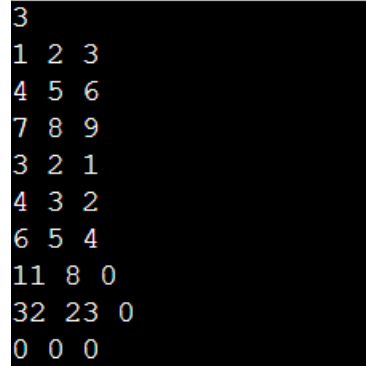
```
void matrix_multiply(vector<vector<int>> &mat1, vector<vector<int>> &mat2, vector<vector<int>>  
&mat3)
```

```
{  
    int i, j, k, n;  
    n = mat1.size();  
    for (i = 0; i < n; i++)  
    {  
        for (j = 0; j < n; j++)  
        {  
            mat3[i][j] = 0;  
            for (k = 0; k < n; k++)  
            {  
                mat3[i][j] += mat1[i][k] * mat2[k][j];  
            }  
        }  
    }  
}
```

```
int main()
```

```
{  
    vector<vector<int>> mat1, mat2;  
    int i, j, n;  
    cin >> n;  
    read_matrix(mat1, n);  
    read_matrix(mat2, n);  
    vector<vector<int>> mat3(n, vector<int>(n, 0));  
    matrix_multiply(mat1, mat2, mat3);  
    print_matrix(mat3);  
    return 0;  
}
```

Output:



```
3  
1 2 3  
4 5 6  
7 8 9  
3 2 1  
4 3 2  
6 5 4  
11 8 0  
32 23 0  
0 0 0
```

2. Iterative Method:

Code window:

```
1  #include <iostream>
2  using namespace std;
3  #include <vector>
4  void read_matrix(vector<vector<int>> &mat, int n)
5  {
6      int ele;
7      for (int i = 0; i < n; i++)
8      {
9          vector<int> row;
10         for (int j = 0; j < n; j++)
11         {
12             cin >> ele;
13             row.push_back(ele);
14         }
15         mat.push_back(row);
16     }
17 }
18 void print_matrix(vector<vector<int>> mat)
19 {
20     int n = mat.size(), i, j;
21     for (i = 0; i < n; i++)
22     {
23         for (j = 0; j < n; j++)
24             cout << mat[i][j] << " ";
25         cout << "\n";
26     }
27 }
28
```

```

29 void matrix_multiply(vector<vector<int>> &mat1, vector<vector<int>> &mat2, vector<vector<int>> &mat3)
30 {
31     int i, j, k, n;
32     n = mat1.size();
33     for (i = 0; i < n; i++)
34     {
35         for (j = 0; j < n; j++)
36         {
37             mat3[i][j] = 0;
38             for (k = 0; k < n; k++)
39             {
40                 mat3[i][j] += mat1[i][k] * mat2[k][j];
41             }
42         }
43     }
44 }
45 int main()
46 {
47     vector<vector<int>> mat1, mat2;
48     int i, j, n;
49     cin >> n;
50     read_matrix(mat1, n);
51     read_matrix(mat2, n);
52     vector<vector<int>> mat3(n, vector<int>(n, 0));
53     matrix_multiply(mat1, mat2, mat3);
54     print_matrix(mat3);
55     return 0;
56 }

```

Code:

```
#include <iostream>
```

```
using namespace std;
```

```
#include <vector>
```

```
void read_matrix(vector<vector<int>> &mat, int n)
```

```
{
```

```
    int ele;
```

```
    for (int i = 0; i < n; i++)
```

```
    {
```

```
        vector<int> row;
```

```
        for (int j = 0; j < n; j++)
```

```
        {
```

```
            cin >> ele;
```

```
            row.push_back(ele);
```

```
        }
```

```
        mat.push_back(row);
```

```

    }
}

void print_matrix(vector<vector<int>> mat)
{
    int n = mat.size(), i, j;
    for (i = 0; i < n; i++)
    {
        for (j = 0; j < n; j++)
            cout << mat[i][j] << " ";
        cout << "\n";
    }
}

```

```

void matrix_multiply(vector<vector<int>> &mat1, vector<vector<int>> &mat2, vector<vector<int>>
&mat3)
{
    int i, j, k, n;
    n = mat1.size();
    for (i = 0; i < n; i++)
    {
        for (j = 0; j < n; j++)
        {
            mat3[i][j] = 0;
            for (k = 0; k < n; k++)
            {
                mat3[i][j] += mat1[i][k] * mat2[k][j];
            }
        }
    }
}

```

```
}  
  
int main()  
{  
    vector<vector<int>> mat1, mat2;  
    int i, j, n;  
    cin >> n;  
    read_matrix(mat1, n);  
    read_matrix(mat2, n);  
    vector<vector<int>> mat3(n, vector<int>(n, 0));  
    matrix_multiply(mat1, mat2, mat3);  
    print_matrix(mat3);  
    return 0;  
}
```

Output:

```
3  
1 2 3  
4 5 6  
7 8 9  
3 2 1  
6 5 4  
9 8 7  
42 36 30  
96 81 66  
150 126 102
```