Exercise 10: To construct a cpp program for matrix multiplication using two dimensional arrays.

Source Code:

```cpp
#include <iostream>

using namespace std;

int main()

{

   int a[10][10], b[10][10], mult[10][10], r1, c1, r2, c2, i, j, k;

   cout << "Enter rows and columns for first matrix: ";

   cin >> r1 >> c1;

   cout << "Enter rows and columns for second matrix: ";

   cin >> r2 >> c2;

   // If column of first matrix in not equal to row of second matrix,

   // ask the user to enter the size of matrix again.

   while (c1!=r2)

   {

      cout << "Error! column of first matrix not equal to row of second.";

      cout << "Enter rows and columns for first matrix: ";

      cin >> r1 >> c1;

      cout << "Enter rows and columns for second matrix: ";

      cin >> r2 >> c2;

   }

   // Storing elements of first matrix.

   cout << endl << "Enter elements of matrix 1:" << endl;

   for(i = 0; i < r1; ++i)

      for(j = 0; j < c1; ++j)

      {

         cout << "Enter element a" << i + 1 << j + 1 << " : ";

         cin >> a[i][j];

      }
```

```cpp
// Storing elements of second matrix.

cout << endl << "Enter elements of matrix 2:" << endl;

for(i = 0; i < r2; ++i)

    for(j = 0; j < c2; ++j)

    {

        cout << "Enter element b" << i + 1 << j + 1 << " : ";

        cin >> b[i][j];

    }

// Initializing elements of matrix mult to 0.

for(i = 0; i < r1; ++i)

    for(j = 0; j < c2; ++j)

    {

        mult[i][j]=0;

    }

// Multiplying matrix a and b and storing in array mult.

for(i = 0; i < r1; ++i)

    for(j = 0; j < c2; ++j)

        for(k = 0; k < c1; ++k)

        {

            mult[i][j] += a[i][k] * b[k][j];

        }

// Displaying the multiplication of two matrix.

cout << endl << "Output Matrix: " << endl;

for(i = 0; i < r1; ++i)

for(j = 0; j < c2; ++j)

{

    cout << " " << mult[i][j];

    if(j == c2-1)

        cout << endl;

}

return 0;   }
```

OUTPUT:

Enter rows and columns for first matrix: 2 2

Enter rows and columns for second matrix: 2 2

Enter elements of matrix 1:

Enter element a11 : 13

Enter element a12 : 13

Enter element a21 : 33

Enter element a22 : 33

Enter elements of matrix 2:

Enter element b11 : 33

Enter element b12 : 33

Enter element b21 : 13

Enter element b22 : 13

Output Matrix:

 598 598

 1518 1518