```cpp
#include <iostream>
using namespace std;
void transpose(int arr[3][3]) {
    for (int i = 0; i < 3; ++i) {
        for (int j = i + 1; j < 3; ++j) {
            int temp = arr[i][j];
            arr[i][j] = arr[j][i];
            arr[j][i] = temp;
        }
    }
    cout << "The transpose of the matrix is: " << endl;
    for (int i = 0; i < 3; i++) {
        for (int j = 0; j < 3; j++) {
            cout << arr[i][j] << " ";
        }
        cout << endl;
    }
}

void matrixmultiplication(int arr5[3][3], int arr6[3][3]) {
    int input;
    int arr7[3][3] = {};

    for (int y = 0; y < 3; y++) {

        for (int t = 0; t < 3; t++) {
            int sum4 = 0;
            for (int u = 0; u < 3; u++) {
                sum4 = sum4 + (arr5[y][u] * arr6[u][t]);
            }
            arr7[y][t] = sum4;
        }

    }
    cout << "Resultant matrix is: " << endl;
    for (int i = 0; i < 3; i++) {       //Multiplied matrix resultant
        for (int j = 0; j < 3; j++) {
            cout << arr7[i][j] << " ";

        }
        cout << endl;

    }
}
void recursion(int x) {
    if (x > 0) {


        cout << 15 << " x " << x << " = " << 15 * x << endl;



        recursion(x - 1);
    }
}

int main() {
    //Question1
    int sum = 0;
    int sum1 = 0;
    int x[3][3] = {};
    cout << "Enter the values for the matrix: ";
    for (int i = 0; i < 3; i++) {
```

```
    for (int j = 0; j < 3; j++) {
        cin >> x[i][j];

    }
}
for (int i = 0; i < 3; i++) {
    for (int j = 0; j < 3; j++) {
        cout << x[i][j] << " ";
    }
    cout << endl;
}
for (int i = 0; i < 3; i++) {
    for (int j = 2 - i; j >= 2 - i; j--) {
        sum = sum + x[i][j];
    }
}
cout << "The sum of right diagonal is: " << sum << endl;

for (int i = 0; i < 3; i++) {
    for (int j = i; j <= i; j++) {
        sum1 = sum1 + x[i][j];
    }
}
cout << "The sum of left diagonal is: " << sum1;

//Question2
int arr1[3][3] = {};
int arr2[3][3] = {};
int arr3[3][3] = {};
cout << "Enter matrix1: ";
for (int i = 0; i < 3; i++) {
    for (int j = 0; j < 3; j++) {
        cin >> arr1[i][j];
    }
}
cout << "Enter matrix2: ";
for (int i = 0; i < 3; i++) {
    for (int j = 0; j < 3; j++) {
        cin >> arr2[i][j];
    }
}
cout << "Matrix1 is: " << endl;
for (int i = 0; i < 3; i++) {
    for (int j = 0; j < 3; j++) {
        cout << arr1[i][j] << " ";
    }
    cout << endl;
}
cout << "Matrix2 is: " << endl;
for (int i = 0; i < 3; i++) {
    for (int j = 0; j < 3; j++) {
        cout << arr2[i][j] << " ";
    }
    cout << endl;
}

for (int m = 0; m < 3; m++) {

    for (int n = 0; n < 3; n++) {
        arr3[m][n] = arr1[m][n] + arr2[m][n];
    }
```

```
}
cout << "The resultant matrix is: " << endl;
for (int i = 0; i < 3; i++) {
    for (int j = 0; j < 3; j++) {
        cout << arr3[i][j] << " ";
    }
    cout << endl;
}


//Question3
int arr4[3][3] = {};
cout << "Enter a matrix: ";
for (int i = 0; i < 3; i++) {
    for (int j = 0; j < 3; j++) {
        cin >> arr4[i][j];
    }
}
cout << "The matrix is: " << endl;
for (int i = 0; i < 3; i++) {
    for (int j = 0; j < 3; j++) {
        cout << arr4[i][j] << " ";
    }
    cout << endl;
}
transpose(arr4);


//Question4:Matrix Multiplication
int arr10[3][3] = {};
int arr13[3][3] = {};
cout << "Enter matrix 1: ";
for (int i = 0; i < 3; i++) {
    for (int j = 0; j < 3; j++) {
        cin >> arr10[i][j];

    }

}
cout << "Enter matrix 2: ";
for (int i = 0; i < 3; i++) {
    for (int j = 0; j < 3; j++) {
        cin >> arr13[i][j];

    }

}
cout << "Matrix 1 is: " << endl;
for (int i = 0; i < 3; i++) {
    for (int j = 0; j < 3; j++) {
        cout << arr10[i][j] << " ";
    }
    cout << endl;
}
cout << "Matrix 2 is: " << endl;
for (int i = 0; i < 3; i++) {
    for (int j = 0; j < 3; j++) {
        cout << arr13[i][j] << " ";
    }
    cout << endl;

}
```

```
    matrixmultiplication(arr10, arr13);

    //Question5
    int r;
    cout << "How much table do you want ";
    cin >> r;
    recursion(r);


}
```

Question1

```
/tmp/uOH0n1yWqQ.o
Enter the values for the matrix: 1
2
3
4
5
6
7
8
9
1 2 3
4 5 6
7 8 9
The sum of right diagonal is: 15
The sum of left diagonal is: 15
```


Enter matrix1: 1
2
3
4
5
6
7
8
9
Enter matrix2: 10
11
12
13
14
15
16
17
18
Matrix1 is:
1 2 3
4 5 6
7 8 9
Matrix2 is:
10 11 12
13 14 15
16 17 18
The resultant matrix is:
11 13 15
17 19 21
23 25 27

Process exited after 16.52 seconds with return value 0
Press any key to continue . . .

Question2

Output

```
/tmp/YC12p5pILa.o
Enter a matrix: 1
2
3
4
5
6
7
8
9
10
11
The matrix is:
1 2 3
4 5 6
7 10 11
The transpose of the matrix is:
1 4 7
2 5 10
3 6 11
```

Question3

```
Enter matrix 2: 1
2
3
4
5
6
7
8
9
Matrix 1 is:
1 2 3
4 5 6
7 8 9
Matrix 2 is:
1 2 3
4 5 6
7 8 9
Resultant matrix is:
30 36 42
66 81 96
102 126 150
```

Question4

```
How much table do you want 20
15 x 20 = 300
15 x 19 = 285
15 x 18 = 270
15 x 17 = 255
15 x 16 = 240
15 x 15 = 225
15 x 14 = 210
15 x 13 = 195
15 x 12 = 180
15 x 11 = 165
15 x 10 = 150
15 x 9 = 135
15 x 8 = 120
15 x 7 = 105
15 x 6 = 90
15 x 5 = 75
15 x 4 = 60
15 x 3 = 45
15 x 2 = 30
15 x 1 = 15
```

Question5