Write a C program on Matrices using functions

```c
#include<stdio.h>
#include<stdlib.h>

// function to add two 3x3 matrix
void add(int m[3][3], int n[3][3], int sum[3][3])
{
  for(int i=0;i<3;i++)
    for(int j=0;j<3;j++)
      sum[i][j] = m[i][j] + n[i][j];
}

// function to subtract two 3x3 matrix
void subtract(int m[3][3], int n[3][3], int result[3][3])
{
  for(int i=0;i<3;i++)
    for(int j=0;j<3;j++)
      result[i][j] = m[i][j] - n[i][j];
}

// function to multiply two 3x3 matrix
void multiply(int m[3][3], int n[3][3], int result[3][3])
{
  for(int i=0; i < 3; i++)
  {
    for(int j=0; j < 3; j++)
    {
      result[i][j] = 0; // assign 0
      // find product
      for (int k = 0; k < 3; k++)
      result[i][j] += m[i][k] * n[k][j];
    }
  }
}

// function to find transpose of a 3x3 matrix
void transpose(int matrix[3][3], int trans[3][3])
{
  for (int i = 0; i < 3; i++)
    for (int j = 0; j < 3; j++)
      trans[i][j] = matrix[j][i];
}
```

```c
// function to display 3x3 matrix
void display(int matrix[3][3])
{
  for(int i=0; i<3; i++)
  {
    for(int j=0; j<3; j++)
      printf("%d\t",matrix[i][j]);

    printf("\n"); // new line
  }
}

// main function
int main()
{
  // matrix
  int a[][3] = { {1,2,3}, {8,9,10}, {3,1,2} };
  int b[][3] = { {1,2,3}, {4,5,6}, {7,8,9} };
  int c[3][3];

  // print both matrix
  printf("First Matrix:\n");
  display(a);
  printf("Second Matrix:\n");
  display(b);

  // variable to take choice
  int choice;

  // menu-driven
  do
  {
    // menu to choose the operation
    printf("\nChoose the matrix operation,\n");
    printf("--------------------------\n");
    printf("1. Addition\n");
    printf("2. Subtraction\n");
    printf("3. Multiplication\n");
    printf("4. Transpose\n");
    printf("5. Transpose\n");
    printf("6. Exit\n");
    printf("--------------------------\n");
    printf("Enter your choice: ");
    scanf("%d", &choice);
```

```c
    switch (choice) {
      case 1:
        add(a, b, c);
        printf("Sum of matrix: \n");
        display(c);
        break;
      case 2:
        subtract(a, b, c);
        printf("Subtraction of matrix: \n");
        display(c);
        break;
      case 3:
        multiply(a, b, c);
        printf("Multiplication of matrix: \n");
        display(c);
        break;
      case 4:
        printf("Transpose of the first matrix: \n");
        transpose(a, c);
        display(c);
        break;
      case 5:
        printf("Transpose of the second matrix: \n");
        transpose(b, c);
        display(c);
        break;
      case 6:
        printf("Thank You.\n");
        exit(0);
      default:
        printf("Invalid input.\n");
        printf("Please enter the correct input.\n");
    }
  }while(1);

  return 0;
}
```

Output :

First Matrix:

1	2	3
8	9	10
3	1	2

Second Matrix:

1	2	3
4	5	6
7	8	9

Choose the matrix operation,

----

1. Addition
2. Subtraction
3. Multiplication
4. Transpose
5. Transpose
6. Exit

Enter your choice: 1

Sum of matrix:

2	4	6
12	14	16
10	9	11

Choose the matrix operation,

----

1. Addition
2. Subtraction
3. Multiplication
4. Transpose
5. Transpose
6. Exit

Enter your choice: 2

Subtraction of matrix:

0	0	0
4	4	4
-4	-7	-7

Choose the matrix operation,
----
1. Addition
2. Subtraction
3. Multiplication
4. Transpose
5. Transpose
6. Exit

Enter your choice: 3
Multiplication of matrix:

30	36	42
114	141	168
21	27	33

Choose the matrix operation,
----
1. Addition
2. Subtraction
3. Multiplication
4. Transpose
5. Transpose
6. Exit

Enter your choice: 4
Transpose of the first matrix:

1	8	3
2	9	1
3	10	2

Choose the matrix operation,
----
1. Addition
2. Subtraction
3. Multiplication
4. Transpose
5. Transpose
6. Exit

Enter your choice: 5
Transpose of the second matrix:

1	4	7
2	5	8
3	6	9

Choose the matrix operation,
----
1. Addition
2. Subtraction
3. Multiplication
4. Transpose
5. Transpose
6. Exit

Enter your choice: 6
Thank You.