

# WEEK 1

## MATRIX MULTIPLICATION PROGRAM

Matrix

```
#include <stdio.h>
#include <stdlib.h>
void add (int a[3][3], int b[3][3])
{
    int c[3][3], i, j;
    for (i=0; i<3; i++)
    {
        for (j=0; j<3; j++)
            c[i][j] = a[i][j] + b[i][j];
    }
    printf ("Addition:");
    for (i=0; i<3; i++)
    {
        for (j=0; j<3; j++)
            printf ("%d\t", c[i][j]);
        printf ("\n");
    }
}

void sub (int a[3][3], int b[3][3])
{
    int c[3][3], i, j;
    for (i=0; i<3; i++)
    {
        for (j=0; j<3; j++)
            c[i][j] = a[i][j] - b[i][j];
    }
    printf ("Subtraction:");
    for (i=0; i<3; i++)
    {
        for (j=0; j<3; j++)
            printf ("%d\t", c[i][j]);
        printf ("\n");
    }
}
```

```
void mul (int a[3][3], int b[3][3])
```

```
{ int c[3][3], i, j, k;  
  for (i=0; i<3; i++)  
  { for (j=0; j<3; j++)  
    { c[i][j]=0;  
      for (k=0; k<3; k++)  
        c[i][j] += a[i][k] * b[k][j];  
    }  
  }  
  printf ("Multiplication: ");  
  for (i=0; i<3; i++)  
  { for (j=0; j<3; j++)  
    { printf (" %d\t", c[i][j]);  
    }  
    printf ("\n");  
  }  
}
```

```
void rowsum (int a[3][3])
```

```
{ int i, j, s;  
  for (i=0; i<3; i++)  
  { s=0;  
    for (j=0; j<3; j++)  
      s += a[i][j];  
    printf ("Sum of %d row: %d", i, s);  
  }  
}
```

```

void columnsum(int a[3][3])
{
    int i, j, s;
    for (i=0; i<3; i++)
    {
        s=0;
        for (j=0; j<3; j++)
            s+=a[j][i];
        printf("In sum of %d column: %d", j, s);
    }
}

```

Output:

Enter the matrix 1:

1	2	3
4	5	6
7	8	9

Enter the matrix 2:

4	2	6
2	4	7
6	1	2

Addition:

5	4	2
6	9	13
13	9	11

Subtraction:

-3	0	-3
2	1	-1
1	7	7

Multiplication:

26	13	26
62	34	71
98	55	116

Rowsum:

sum of 1 row = 6

sum of 2 row = 15

sum of 3 row = 27

Column sum:

sum of 1 column: 12

sum of 2 column: 15

sum of 3 column: 18

Q  
8/6/23

## C PROGRAM CODE:

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

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

```
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];
```

```
}
```

```
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];
```

```
}
```

```
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;
for (int k = 0; k < 3; k++)
result[i][j] += m[i][k] * n[k][j];
}
}
}

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];
}

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");
}
}

```

```
}  
}  
int main()  
{  
    int a[][3] = { {5,6,7}, {8,9,10}, {3,1,2} };  
    int b[][3] = { {1,2,3}, {4,5,6}, {7,8,9} };  
    int c[3][3];  
    printf("First Matrix:\n");  
    display(a);  
    printf("Second Matrix:\n");  
    display(b);  
    int choice;  
    do  
    {  
        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. Exit\n");  
printf("-----\n");  
printf("Enter your choice: ");  
scanf("%d", &choice);  
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);
```

```
printf("Transpose of the second matrix: \n");
```

```
transpose(b, c);
```

```
display(c);
```

```
break;
```

case 5:

```
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:
5  6  7
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. Exit
-----
Enter your choice: 1
Sum of matrix:
6  8 10
12 14 16
10 9 11
```

```
Choose the matrix operation,
-----
1. Addition
2. Subtraction
3. Multiplication
4. Transpose
5. Exit
-----
Enter your choice: 2
Subtraction of matrix:
4  4  4
4  4  4
-4 -7 -7
```

Choose the matrix operation,

-----

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

-----

Enter your choice: 3

Multiplication of matrix:

78 96 114

114 141 168

21 27 33

Choose the matrix operation,

-----

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

-----

Enter your choice: 4

Transpose of the first matrix:

5 8 3

6 9 1

7 10 2

Transpose of the second matrix:

1 4 7

2 5 8

3 6 9