

Name: Deepanshu Gupta

Section: E

Roll No.: 51

Stream: B. Tech. ML+AI

Sem: 1st

Assignment-4

Q1: Write a program to find the sum of each row and each column of a matrix. Also find the sum of diagonal elements.

SOURCE CODE

```
#include<stdio.h>
int main() {
    printf("-----DETAILS-----\nName: Deepanshu Gupta\nRoll No.: 51\n");
    printf("-----OUTPUT-----\n");
    int i, j, r, c;
    printf("Enter number of rows in a matrix: ");
    scanf("%d", &r);
    printf("Enter number of columns in a matrix: ");
    scanf("%d", &c);
    int arr[r][c];
    for (i=0; i<r; i++) {
        for (j=0; j<c; j++) {
            printf("Enter element of row %d and column %d: ", i+1, j+1);
            scanf("%d", &arr[i][j]);
        } for(i=0; i<r; i++) {
            int sum=0;
            for (j=0; j<c; j++)
                sum+=arr[i][j];
            printf("Sum of row %d = %d\n", i+1, sum);
        } for(i=0; i<c; i++) {
            int sum=0;
            for (j=0; j<r; j++)
                sum+=arr[j][i];
            printf("Sum of column %d = %d\n", i+1, sum);
        } int sum=0;
        for(i=0; i<r; i++) {
            for (j=0; j<c; j++){
                if (i==j)
                    sum+=arr[i][j];
            }
        } printf("Sum of diagonal elements = %d\n", sum);
    return 0;
}
```

OUTPUT

```
PS C:\Users\Deepanshu\Desktop\WarIsOn> ./a.exe
```

-----DETAILS-----

Name: Deepanshu Gupta

Roll No.: 51

-----OUTPUT-----

Enter number of rows in a matrix: 3

```
Enter number of columns in a matrix: 3
```

```
Enter element of row 1 and column 1: 1
```

Enter element of row 1 and column 2: 2

Enter element of row 1 and column 3: 3

Enter element of row 2 and column 1: 4

Enter element of row 2 and column 2: 5

```
Enter element of row 2 and column 3: 6
```

```
Enter element of row 3 and column 1: 7
```

Enter element of row 3 and column 2: 8

Enter element of row 3 and column 3: 9

Sum of row 1 = 6

Sum of row 2 = 15

Sum of row 3 = 24

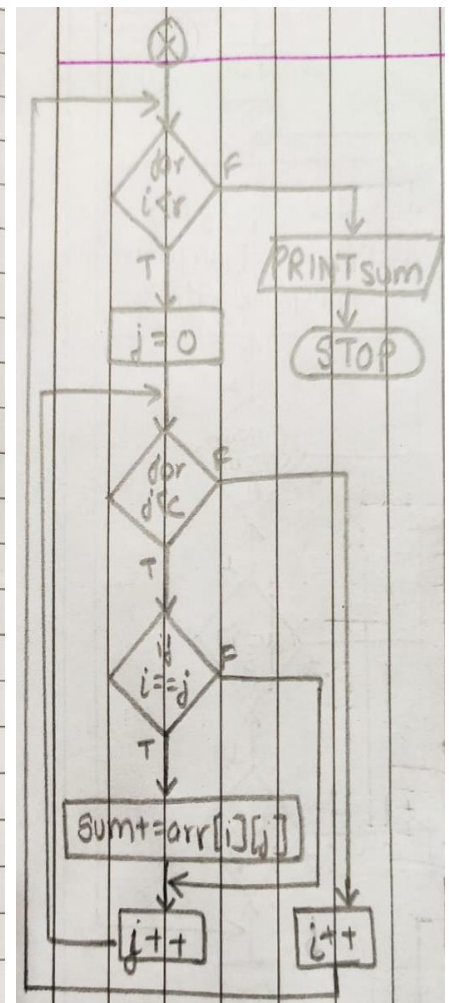
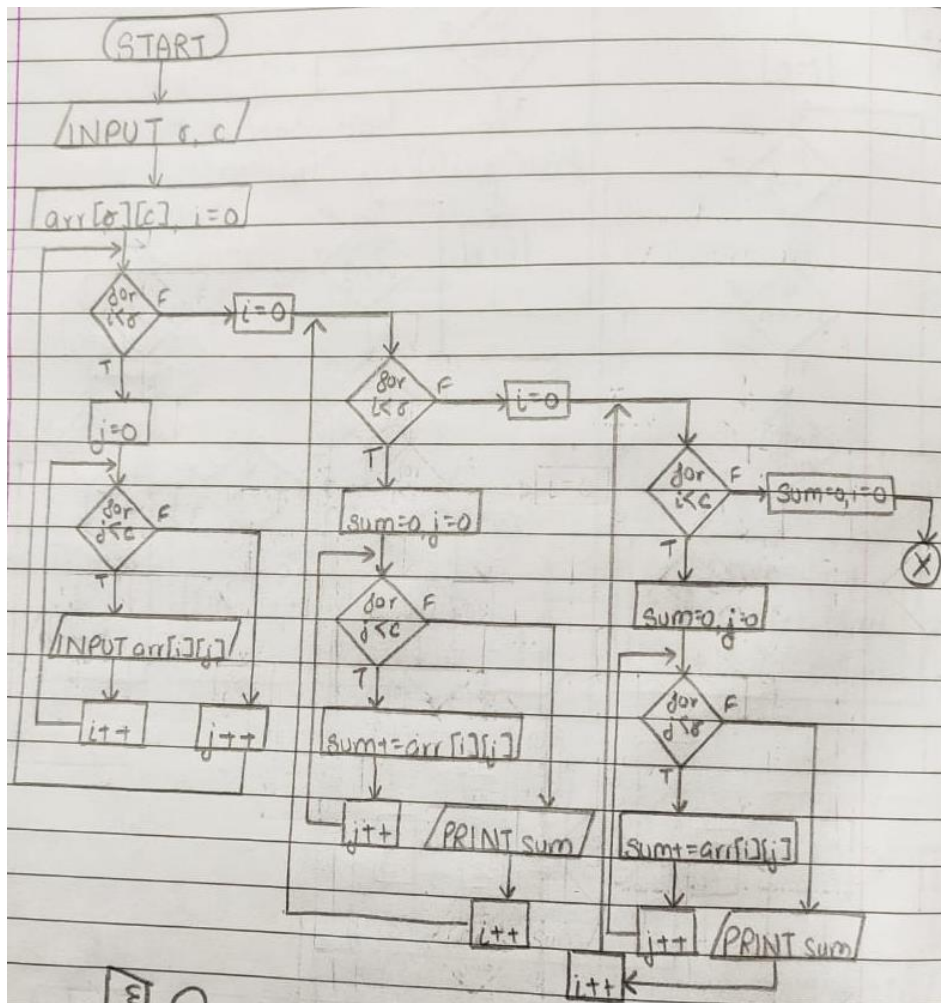
Sum of column 1 = 12

Sum of column 2 = 15

Sum of column 3 = 18

Sum of diagonal elements = 15

FLOWCHART



Q2: Write a program to test whether a given matrix is upper triangular or not. (An upper triangular matrix is one in which all the elements below its principal diagonal are zero).

SOURCE CODE

```
#include<stdio.h>
int main() {
    printf("-----DETAILS-----\n");
    printf("Name: Deepanshu Gupta\nRoll No.: 51\n");
    printf("-----OUTPUT-----\n");
    int i, j, n, loop=0, count=0;
    printf("Enter the order of a matrix: ");
    scanf("%d", &n);
    int arr[n][n];
    for (i=0; i<n; i++) {
        for (j=0; j<n; j++) {
            printf("Enter element of row %d and column %d: ", i+1, j+1);
            scanf("%d", &arr[i][j]);
        }
    }
    for (i=1; i<n; i++) {
        for (j=0; j<i; j++) {
            loop++;
            if (arr[i][j]==0)
                count++;
        }
    }
    for (i=0; i<n; i++) {
        for (j=0; j<n; j++)
            printf("%d\t", arr[i][j]);
        printf("\n");
    }
    if (count==loop)
        printf("Yes, given matrix is upper triangular\n");
    else
        printf("No, given matrix is not upper triangular\n");
    return 0;
}
```

OUTPUT

```
PS C:\Users\Deepanshu\Desktop\WarIsOn> ./a.exe
```

```
-----DETAILS-----
```

```
Name: Deepanshu Gupta
```

```
Roll No.: 51
```

```
-----OUTPUT-----
```

```
Enter the order of a matrix: 3
```

```
Enter element of row 1 and column 1: 1
```

```
Enter element of row 1 and column 2: 2
```

```
Enter element of row 1 and column 3: 3
```

```
Enter element of row 2 and column 1: 0
```

```
Enter element of row 2 and column 2: 5
```

```
Enter element of row 2 and column 3: 6
```

```
Enter element of row 3 and column 1: 0
```

```
Enter element of row 3 and column 2: 0
```

```
Enter element of row 3 and column 3: 9
```

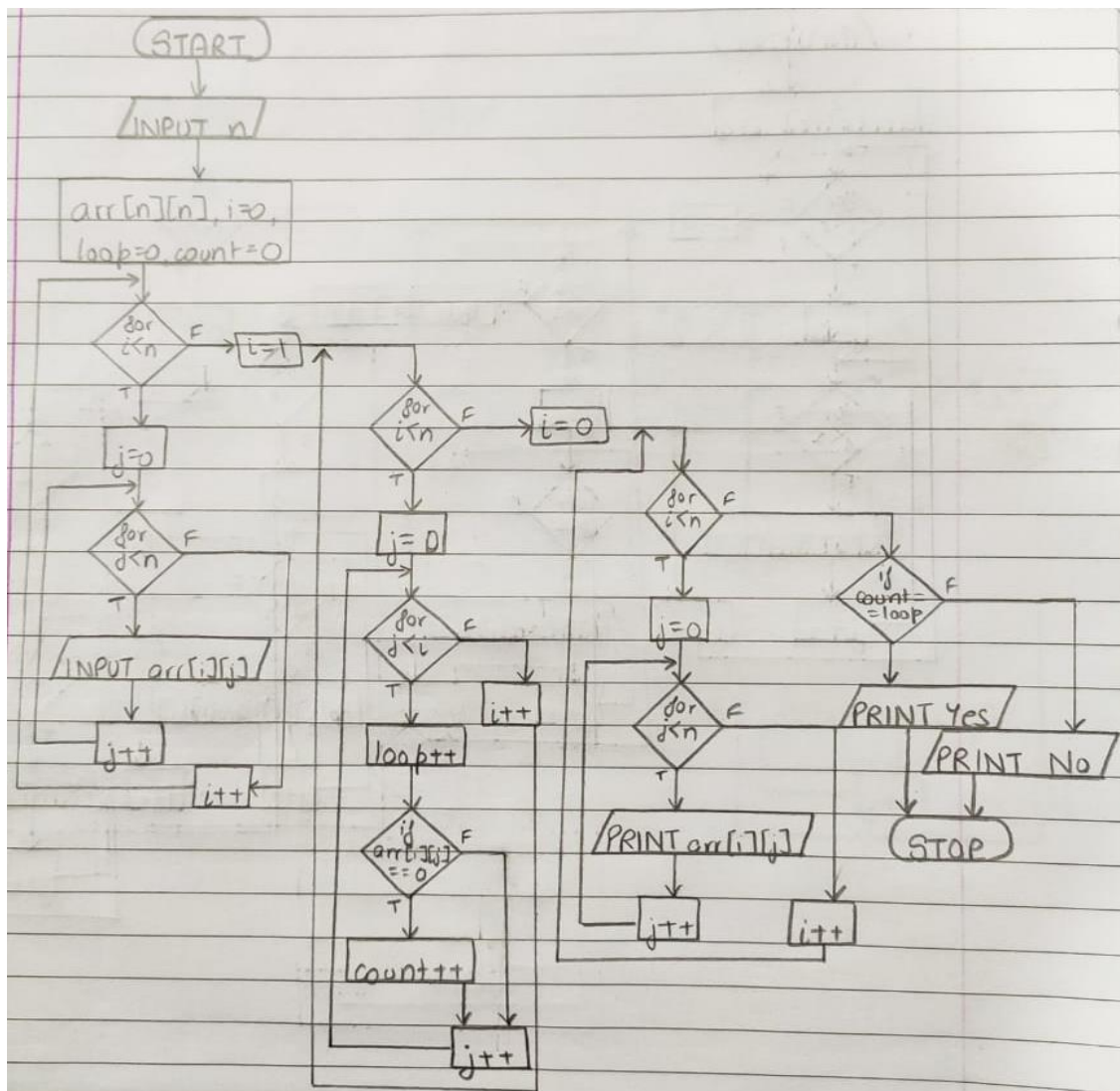
```
1      2      3
```

```
0      5      6
```

```
0      0      9
```

```
Yes, given matrix is upper triangular
```

FLOWCHART



Q3: Write a program to find the transpose of a matrix.

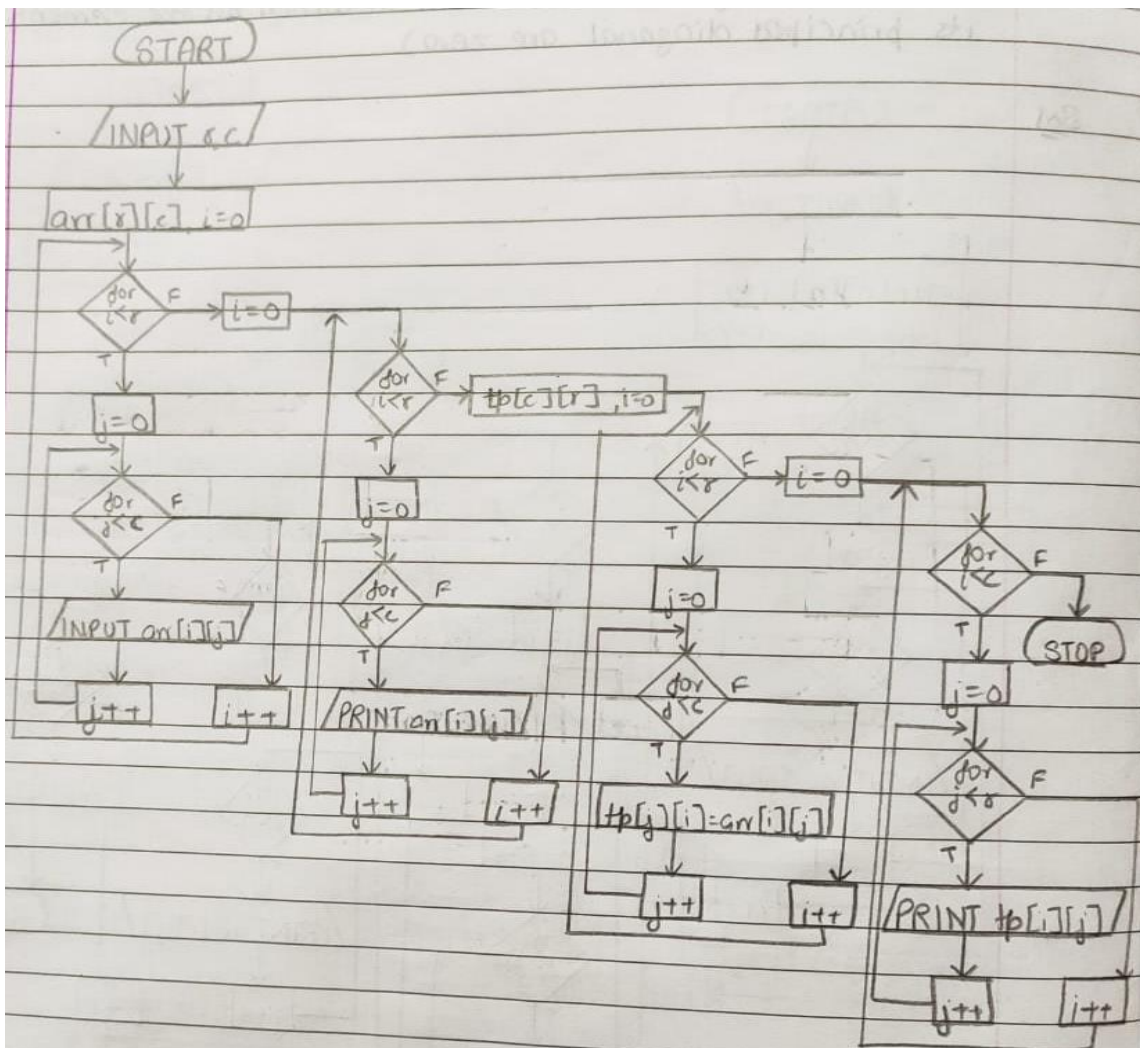
SOURCE CODE

```
#include<stdio.h>
int main() {
    printf("-----DETAILS-----\n");
    printf("Name: Deepanshu Gupta\nRoll No.: 51\n");
    printf("-----OUTPUT-----\n");
    int i, j, r, c;
    printf("Enter number of rows in a matrix: ");
    scanf("%d", &r);
    printf("Enter number of columns in a matrix: ");
    scanf("%d", &c);
    int arr[r][c];
    for (i=0; i<r; i++) {
        for (j=0; j<c; j++) {
            printf("Enter element of row %d and column %d: ", i+1, j+1);
            scanf("%d", &arr[i][j]);
        }
    }
    printf("Given matrix: \n");
    for (i=0; i<r; i++) {
        for (j=0; j<c; j++)
            printf("%d\t", arr[i][j]);
        printf("\n");
    }
    int tp[c][r];
    for (i=0; i<r; i++) {
        for (j=0; j<c; j++) {
            tp[j][i]=arr[i][j];
        }
    }
    printf("Transpose of given matrix: \n");
    for (i=0; i<c; i++) {
        for (j=0; j<r; j++)
            printf("%d\t", tp[i][j]);
        printf("\n");
    }
    return 0;
}
```


OUTPUT

```
PS C:\Users\Deepanshu\Desktop\WarIsOn> ./a.exe
-----DETAILS-----
Name: Deepanshu Gupta
Roll No.: 51
-----OUTPUT-----
Enter number of rows in a matrix: 3
Enter number of columns in a matrix: 2
Enter element of row 1 and column 1: 1
Enter element of row 1 and column 2: 2
Enter element of row 2 and column 1: 3
Enter element of row 2 and column 2: 4
Enter element of row 3 and column 1: 5
Enter element of row 3 and column 2: 6
Given matrix:
1      2
3      4
5      6
Transpose of given matrix:
1      3      5
2      4      6
```

FLOWCHART



Q4: Write a program to find the product of two matrix.

SOURCE CODE

```
#include<stdio.h>
int main() {
    printf("-----DETAILS-----\n");
    printf("Name: Deepanshu Gupta\nRoll No.: 51\n");
    printf("-----OUTPUT-----\n");
    int i, j, k, r1, c1, r2, c2;
    printf("Enter number of rows in a matrix 1: ");
    scanf("%d", &r1);
    printf("Enter number of columns in a matrix 1: ");
    scanf("%d", &c1);
    printf("Enter number of rows in a matrix 2: ");
    scanf("%d", &r2);
    printf("Enter number of columns in a matrix 2: ");
    scanf("%d", &c2);
    printf("For Matrix 1:\n");
    int arr1[r1][c1];
    for (i=0; i<r1; i++) {
        for (j=0; j<c1; j++) {
            printf("Enter element of row %d and column %d: ", i+1, j+1);
            scanf("%d", &arr1[i][j]);
        }
    }
    printf("For Matrix 2:\n");
    int arr2[r2][c2];
    for (i=0; i<r2; i++) {
        for (j=0; j<c2; j++) {
            printf("Enter element of row %d and column %d: ", i+1, j+1);
            scanf("%d", &arr2[i][j]);
        }
    }
    printf("Given matrices: \n");
    printf("Matrix 1:\n");
    for (i=0; i<r1; i++) {
        for (j=0; j<c1; j++)
            printf("%d\t", arr1[i][j]);
        printf("\n");
    }
    printf("Matrix 2:\n");
    for (i=0; i<r2; i++) {
        for (j=0; j<c2; j++)
            printf("%d\t", arr2[i][j]);
        printf("\n");
    }
    if (c1==r2) {
        int multi[r1][c2];
        for(i=0; i<r1; i++) {
```

```

for(j=0; j<c2; j++){
    int sum=0;
    for (k=0; k<r2; k++)
        sum+=(arr1[i][k]*arr2[k][j]);
    multi[i][j]=sum;
}
} printf("Resultant Matrix:\n");
for (i=0; i<r1; i++) {
    for (j=0; j<c2; j++)
        printf("%d\t", multi[i][j]);
    printf("\n");
}
} else printf("Multiplication is not possible\n");
return 0;
}

```

OUTPUT

```

PS C:\Users\Deepanshu\Desktop\WarIsOn> ./a.exe
-----DETAILS-----
Name: Deepanshu Gupta
Roll No.: 51
-----OUTPUT-----
Enter number of rows in a matrix 1: 2
Enter number of columns in a matrix 1: 2
Enter number of rows in a matrix 2: 2
Enter number of columns in a matrix 2: 3
For Matrix 1:
Enter element of row 1 and column 1: 1
Enter element of row 1 and column 2: 2
Enter element of row 2 and column 1: 3
Enter element of row 2 and column 2: 4
For Matrix 2:
Enter element of row 1 and column 1: 1
Enter element of row 1 and column 2: 2
Enter element of row 1 and column 3: 3
Enter element of row 2 and column 1: 4
Enter element of row 2 and column 2: 5
Enter element of row 2 and column 3: 6#include<stdio.h>
Given matrices:
Matrix 1:
1      2
3      4
Matrix 2:
1      2      3
4      5      6
Resultant Matrix:
9      12     15
19     26     33

```


FLOWCHART

