1. Write a program to calculate the sum of two matrices each of order 3x3.

```c
#include<stdio.h>
int main()
{
    int a[3][3], b[3][3], c[3][3];
    printf("Enter matrix 1 elements\n");
    for (int i = 0; i < 3; i++)
    {
        for (int j = 0; j < 3; j++)
        {
            scanf("%d", &a[i][j]);
        }
    }
    printf("Enter matrix 2 elements\n");
    for (int i = 0; i < 3; i++)
    {
        for (int j = 0; j < 3; j++)
        {
            scanf("%d", &b[i][j]);
        }
    }

    // below for-loop is, for addition of 2 matrix
    for (int i = 0; i < 3; i++)
    {
        for (int j = 0; j < 3; j++)
        {
            c[i][j] =  a[i][j] +  b[i][j];
        }
    }
    printf("Sum of 2 matrices\n");
    for (int i = 0; i < 3; i++)
    {
        for (int j = 0; j < 3; j++)
        {
            printf("%3d ", c[i][j]);
        }
        printf("\n");
    }
    return 0;
}
```

```
================================================================================
Output:
Enter matrix 1 elements
-5 2 0
7 -3 4
-1 3 2
Enter matrix 2 elements
0 -1 8
6 -14 2
9 5 1
Sum of 2 matrices
 -5   1   8
 13 -17   6
  8   8   3
```

2. Write a program to calculate the product of two matrices each of order 3x3.

```c
#include<stdio.h>
int main()
```

```
{
    int a[3][3], b[3][3], c[3][3], sum;
    printf("Enter matrix 1 elements\n");
    for (int i = 0; i < 3; i++)
    {
        for (int j = 0; j < 3; j++)
        {
            scanf("%d", &a[i][j]);
        }
    }
    printf("Enter matrix 2 elements\n");
    for (int i = 0; i < 3; i++)
    {
        for (int j = 0; j < 3; j++)
        {
            scanf("%d", &b[i][j]);
        }
    }

    // below for-loop is, for product of 2 matrix
    for (int i = 0; i < 3; i++)
    {
        for (int j = 0; j < 3; j++)
        {
            sum = 0;
            for (int k = 0; k < 3; k++)
            {
                sum = sum + a[i][k] * b[k][j];
            }
            c[i][j] = sum;
        }
    }
    printf("Product of 2 matrix\n");
    for (int i = 0; i < 3; i++)
    {
        for (int j = 0; j < 3; j++)
        {
            printf("%2d ", c[i][j]);
        }
        printf("\n");
    }
    return 0;
}
================================================================================
Output:
Enter matrix 1 elements
1 2 3
4 5 6
7 0 4
Enter matrix 2 elements
4 7 8
0 2 3
5 6 7
Product of 2 matrix
19 29 35
46 74 89
48 73 84
```

3. Write a program in C to find the transpose of a given matrix.

```
#include <stdio.h>
int main()
{
    int a[10][10], transpose[10][10], r, c;
```

```
     printf("Enter how many rows and columns to have in a matrix (Max 10 x
10): ");
    scanf("%d%d", &r, &c);
    printf("Enter %d x %d matrix elements\n", r, c);
    for (int i = 0; i < r; i++)
    {
        for (int j = 0; j < c; j++)
        {
            scanf("%d", &a[i][j]);
            transpose[j][i] = a[i][j];
        }
    }

    printf("The transpose of a given matrix\n");
    for (int i = 0; i < c; i++)
    {
        for (int j = 0; j < r; j++)
        {
            printf("%2d ", transpose[i][j]);
        }
        printf("\n");
    }
    return 0;
}
```

```
Output:
Enter how many rows and columns to have in a matrix (Max 10 x 10): 2 4
Enter 2 x 4 matrix elements
8 4 6 2
5 9 1 3
The transpose of a given matrix
 8  5
 4  9
 6  1
 2  3
```

4. Write a program in C to find the sum of right diagonals of a matrix.

```
#include <stdio.h>
int main()
{
    int a[10][10], i, j, sum = 0, r;
    printf("Enter number of rows for square matrix (max 10 rows) : ");
    scanf("%d", &r);
    printf("Enter %d x %d element of matrix\n", r, r);
    for (i = 0; i < r; i++)
    {
        for (j = 0; j < r; j++)
        {
            scanf("%d", &a[i][j]);
        }
    }
    for (i = 0; i < r; i++)
    {
        sum += a[i][i];
    }
    printf("The sum of right diagonals of a matrix : %d", sum);
    return 0;
}
```

```
Output:
Enter number of rows for square matrix (max 10 rows) : 3
Enter 3 x 3 element of matrix
6 5 4
2 5 8
```

```
3 5 7
The sum of right diagonals of a matrix : 18
```

5. Write a program in C to find the sum of left diagonals of a matrix.

```c
#include <stdio.h>
int main()
{
    int a[10][10], i, j, sum = 0, r;
    printf("Enter number of rows for square matrix (max 10 rows) : ");
    scanf("%d", &r);
    printf("Enter %d x %d element of matrix\n", r, r);
    for (i = 0; i < r; i++)
    {
        for (j = 0; j < r; j++)
        {
            scanf("%d", &a[i][j]);
        }
    }
    for (i = 0; i < r; i++)
    {
        sum += a[i][(r - 1) - i];
    }
    printf("The sum of left diagonals of a matrix : %d", sum);
    return 0;
}
```

```
================================================================================
Output:
Enter number of rows for square matrix (max 10 rows) : 4
Enter 4 x 4 element of matrix
1 2 3 4
3 2 1 4
7 8 9 6
9 5 3 4
The sum of left diagonals of a matrix : 22
```

6. Write a program in C to find the sum of rows and columns of a Matrix.

```c
#include <stdio.h>
int main()
{
    int a[10][10], i, j, sum, r, c;
    printf("Enter number of rows and column of a matrix (max 10 x 10) : ");
    scanf("%d%d", &r, &c);
    printf("Enter %d x %d element of matrix\n", r, c);
    for (i = 0; i < r; i++)
    {
        for (j = 0; j < c; j++)
        {
            scanf("%d", &a[i][j]);
        }
    }

    // below for loop is for sum of rows
    for (i = 0; i < r; i++)
    {
        sum = 0;
        for (j = 0; j < c; j++)
        {
            sum += a[i][j];
        }
        printf("Sum of row %d is %d\n", i + 1, sum);
    }

    // below for loop is for sum of column
```

```
    for (i = 0; i < c; i++)
    {
        sum = 0;
        for (j = 0; j < r; j++)
        {
            sum += a[j][i];
        }
        printf("Sum of column %d is %d\n", i + 1, sum);
    }
    return 0;
}
```

```
Output:
Enter number of rows and column of a matrix (max 10 x 10) : 3 3
Enter 3 x 3 element of matrix
4 5 6
2 5 8
3 5 7
Sum of row 1 is 15
Sum of row 2 is 15
Sum of row 3 is 15
Sum of column 1 is 9
Sum of column 2 is 15
Sum of column 3 is 21
```

7. Write a program in C to print or display the lower triangular of a given matrix.

```
#include <stdio.h>
int main()
{
    int a[10][10], i, j, sum = 0, r;
    printf("Enter number of rows for square matrix (max 10 rows) : ");
    scanf("%d", &r);
    printf("Enter %d x %d element of matrix\n", r, r);
    for (i = 0; i < r; i++)
    {
        for (j = 0; j < r; j++)
        {
            scanf("%d", &a[i][j]);
        }
    }
    printf("Given Matrix\n");
    for (i = 0; i < r; i++)
    {
        for (j = 0; j < r; j++)
        {
            printf("%d ", a[i][j]);
        }
        printf("\n");
    }
    printf("The lower triangular of a given matrix\n");
    for (i = 0; i < r; i++)
    {
        for (j = 0; j < r; j++)
        {
            if (j <= i)
            {
                printf("%d ", a[i][j]);
            }
            else
            {
                printf("0 ");
            }
        }
        printf("\n");
```

```
    }
    return 0;
}
```

```
Output:
Enter number of rows for square matrix (max 10 rows) : 5
Enter 5 x 5 element of matrix
5 6 3 2 4
1 2 3 5 4
7 8 9 6 5
3 6 9 8 7
1 4 7 8 9
Given Matrix
5 6 3 2 4
1 2 3 5 4
7 8 9 6 5
3 6 9 8 7
1 4 7 8 9
The lower triangular of a given matrix
5 0 0 0 0
1 2 0 0 0
7 8 9 0 0
3 6 9 8 0
1 4 7 8 9
```

8. Write a program in C to print or display an upper triangular matrix.

```c
#include <stdio.h>
int main()
{
    int a[10][10], i, j, sum = 0, r;
    printf("Enter number of rows for square matrix (max 10 rows) : ");
    scanf("%d", &r);
    printf("Enter %d x %d element of matrix\n", r, r);
    for (i = 0; i < r; i++)
    {
        for (j = 0; j < r; j++)
        {
            scanf("%d", &a[i][j]);
        }
    }
    printf("Given Matrix\n");
    for (i = 0; i < r; i++)
    {
        for (j = 0; j < r; j++)
        {
            printf("%d ", a[i][j]);
        }
        printf("\n");
    }
    printf("The upper triangular of a given matrix\n");
    for (i = 0; i < r; i++)
    {
        for (j = 0; j < r; j++)
        {
            if (j >= i)
            {
                printf("%d ", a[i][j]);
            }
            else
            {
                printf("0 ");
            }
        }
        printf("\n");
```

```
    }
    return 0;
}
```

```
Output:
Enter number of rows for square matrix (max 10 rows) : 5
Enter 5 x 5 element of matrix
1 4 7 8 9
3 6 9 8 7
1 2 3 6 9
3 2 1 4 7
7 8 9 6 3
Given Matrix
1 4 7 8 9
3 6 9 8 7
1 2 3 6 9
3 2 1 4 7
7 8 9 6 3
The upper triangular of a given matrix
1 4 7 8 9
0 6 9 8 7
0 0 3 6 9
0 0 0 4 7
0 0 0 0 3
```

9. Write a program in C to accept a matrix and determine whether it is a sparse matrix.

```
#include <stdio.h>
int main()
{
    int a[10][10], i, j, count = 0, r, c;
    printf("Enter number of rows and column of a matrix (max 10 x 10) : ");
    scanf("%d%d", &r, &c);
    printf("Enter %d x %d element of matrix\n", r, c);
    for (i = 0; i < r; i++)
    {
        for (j = 0; j < c; j++)
        {
            scanf("%d", &a[i][j]);
        }
    }
    printf("Given Matrix\n");
    for (i = 0; i < r; i++)
    {
        for (j = 0; j < c; j++)
        {
            printf("%d ", a[i][j]);
        }
        printf("\n");
    }
    for (i = 0; i < r; i++)
    {
        for (j = 0; j < c; j++)
        {
            if (a[i][j] == 0)
            {
                count++;
            }
        }
    }
    count > (r * c) / 2 ? printf("It is a sparse matrix") : printf("It is not
a sparse matrix");
    return 0;
}
```

```
Output:
Enter number of rows and column of a matrix (max 10 x 10) : 5 5
Enter 5 x 5 element of matrix
1 0 1 0 1
0 2 5 0 0
0 0 0 7 1
0 0 0 0 1
9 0 0 0 0
Given Matrix
1 0 1 0 1
0 2 5 0 0
0 0 0 7 1
0 0 0 0 1
9 0 0 0 0
It is a sparse matrix
```

10. Write a program in C to find the row with maximum number of 1s.

```c
#include <stdio.h>
int main()
{
    int a[10][10], i, j, count, r, c, ones = 0, next_i = 0;
    printf("Enter number of rows and column of a matrix (max 10 x 10) : ");
    scanf("%d%d", &r, &c);
    printf("Enter %d x %d element of matrix\n", r, c);
    for (i = 0; i < r; i++)
    {
        for (j = 0; j < c; j++)
        {
            scanf("%d", &a[i][j]);
        }
    }
    printf("Given Matrix\n");
    for (i = 0; i < r; i++)
    {
        for (j = 0; j < c; j++)
        {
            printf("%d ", a[i][j]);
        }
        printf("\n");
    }
    for (i = 0; i < r; i++)
    {
        count = 0;
        for (j = 0; j < c; j++)
        {
            if (a[i][j] == 1)
            {
                count++;
            }
        }
        if (count > ones)
        {
            ones = count;
            next_i = i;
        }
    }
    printf("The row with maximum number of 1s is %d", next_i + 1);
    return 0;
}
```

```
================================================================================
Output:
Enter number of rows and column of a matrix (max 10 x 10) : 4 4
Enter 4 x 4 element of matrix
1 1 1 1
```

```
1 0 0 1
0 0 0 1
1 0 1 0
Given Matrix
1 1 1 1
1 0 0 1
0 0 0 1
1 0 1 0
The row with maximum number of 1s is 1
```