Assignment – 16 A Job Ready Bootcamp in C++, DSA and IOT MySirG

Multi-Dimensional Array in C Language

1. Write a program to calculate the sum of two matrices each of order 3x3.
2. #include<stdio.h>
3. int main()
4. {
5. int a[3][3], b[3][3], c[3][3], i,j;
6. printf("Enter First Marix of the Elements :\n");
7. for ( i = 0; i < 3; i++)
8. {
9. for(j=0; j<3; j++)
10. {
11. scanf("%d", &a[i][j]);
12. }
13. }
14. printf("Enter Second Marix of the Elements :\n");
15. for ( i = 0; i < 3; i++)
16. {
17. for(j=0; j<3; j++)
18. {
19. scanf("%d", &b[i][j]);
20. }
21. }
22. for ( i = 0; i < 3; i++)
23. {
24. for(j=0; j<3; j++)
25. {
26. c[i][j]=a[i][j]+b[i][j];
27. }
28. }
29. printf("Sum of the Two Matrix is :\n");
30. for ( i = 0; i < 3; i++)
31. {
32. for(j=0; j<3; j++)
33. {
34. printf("%d ", c[i][j]);
35. }
36. printf("\n");
37. }
38. }

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

#include<stdio.h>

int main()

{

    int a[3][3];

    int b[3][3];

    int c[3][3];

    int i,j,k,sum=0;

    printf("Enter elements of first matrix: \n");

    for (int i = 0; i < 3; i++)

    {

        for ( j = 0; j < 3; j++)

        {

            scanf("%d", &a[i][j]);

        }

    }

    printf("Enter elements of second matrix: \n");

    for (int i = 0; i < 3; i++)

    {

        for ( j = 0; j < 3; j++)

        {

            scanf("%d", &b[i][j]);

        }

    }

    for ( i = 0; i < 3; i++)

    {

        for ( j = 0; j < 3; j++)

        {

            for ( k = 0; k < 3; k++)

            {

                sum=sum+a[i][k]\*b[k][j];

                           }

            c[i][j]=sum;

            sum=0;

        }

    }

    for ( i = 0; i < 3; i++)

    {

        for ( j = 0; j < 3; j++)

        {

            printf("|%d ", c[i][j]);

        }

        printf("\n");

    }

}

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

#include<stdio.h>

int main()

{

    int a[3][3];

    int b[3][3];

    int c[3][3];

    int i,j,k,sum=0;

    printf("Enter elements of first matrix: \n");

    for (int i = 0; i < 3; i++)

    {

        for ( j = 0; j < 3; j++)

        {

            scanf("%d", &a[i][j]);

        }

    }

    for ( i = 0; i < 3; i++)

    {

        for ( j = 0; j < 3; j++)

        {

            b[i][j]=a[j][i];

        }

    }

    for ( i = 0; i < 3; i++)

    {

        for ( j = 0; j < 3; j++)

        {

            printf("|%d ", b[i][j]);

        }

        printf("\n");

    }

}

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

#include<stdio.h>

int main()

{

    int a[3][3];

    int i,j,k,s;

    printf("Enter elements of first matrix: \n");

    for (int i = 0; i < 3; i++)

    {

        for ( j = 0; j < 3; j++)

        {

            scanf("%d", &a[i][j]);

        }

    }

    for ( i = 0; i < 3; i++)

    {

        for ( j = 0; j < 3; j++)

        {

            k=a[0][0]+a[1][1]+a[2][2];

        }

    }

    printf("%d", k);

}

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

#include<stdio.h>

int main()

{

    int a[3][3];

    int i,j,k,s;

    printf("Enter elements of first matrix: \n");

    for (int i = 0; i < 3; i++)

    {

        for ( j = 0; j < 3; j++)

        {

            scanf("%d", &a[i][j]);

        }

    }

    for ( i = 0; i < 3; i++)

    {

        for ( j = 0; j < 3; j++)

        {

            k=a[0][2]+a[1][1]+a[2][0];

        }

    }

    printf("%d", k);

}

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

#include<stdio.h>

int main()

{

    int a[3][3];

    int i,j,k=0,s=0;

    printf("Enter elements of first matrix: \n");

    for (int i = 0; i < 3; i++)

    {

        for ( j = 0; j < 3; j++)

        {

            scanf("%d", &a[i][j]);

        }

    }

    printf("rows sum \n");

    for ( i = 0; i < 3; i++)

    {

        k=0;

        for ( j = 0; j < 3; j++)

        {

            k=k+a[i][j];

        }

        printf("%d ", k);

    }

     printf("\ncolumn sum\n");

    for ( i = 0; i < 3; i++)

    {

        s=0;

        for ( j = 0; j < 3; j++)

        {

            s=s+a[j][i];

        }

        printf("%d ", s);

    }

    }

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

#include<stdio.h>

int main()

{

    int a[3][3];

    int i,j,k=0,s=0;

    printf("Enter elements of first matrix: \n");

    for (int i = 0; i < 3; i++)

    {

        for ( j = 0; j < 3; j++)

        {

            scanf("%d", &a[i][j]);

        }

    }

    for ( i = 0; i < 3; i++)

    {

        for ( j = 0; j < 3; j++)

        {

            if (i>j)

            {

                a[i][j]=0;

            }

            printf("%d ", a[i][j]);

        }

        printf("\n");

    }

}

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

#include<stdio.h>

int main()

{

    int a[3][3];

    int i,j,k=0,s=0;

    printf("Enter elements of first matrix: \n");

    for (int i = 0; i < 3; i++)

    {

        for ( j = 0; j < 3; j++)

        {

            scanf("%d", &a[i][j]);

        }

    }

    for ( i = 0; i < 3; i++)

    {

        for ( j = 0; j < 3; j++)

        {

            if (i<j)

            {

                a[i][j]=0;

            }

            printf("%d ", a[i][j]);

        }

        printf("\n")

    }

}

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[3][3];

    int i,j,m,n,ctr=0;

    printf("Enter row number\n");

    scanf("%d", &m);

    printf("Enter column number\n");

    scanf("%d", &n);

    printf("Enter elements of the matrix\n");

    for ( i = 0; i < m; i++)

    {

        for ( j = 0; j < n; j++)

        {

            scanf("%d", &a[i][j]);

            if (a[i][j]==0)

            {

                ++ctr;

            }

        }

    }

    if (ctr>(m\*n)/2)

    {

        printf("Given matrix is spares matrix\n");

    }

    else

    {

        printf("Not spares matrix");

    }

}

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

#include<stdio.h>

int main()

{

    int a[3][3];

    int i,j,m,n,ctr=0,max=0,index;

    printf("Enter row number\n");

    scanf("%d", &m);

    printf("Enter column number\n");

    scanf("%d", &n);

    printf("Enter elements of the matrix\n");

    for ( i = 0; i < m; i++)

    {

        for ( j = 0; j < n; j++)

        {

            scanf("%d", &a[i][j]);

        }

    }

     for ( i = 0; i < m; i++)

     {

        for ( j = 0; j < n; j++)

        {

            if (a[i][j]==1)

            {

                ctr=a[i][j]+ctr;

            }

            if (ctr>max)

        {

            max=ctr;

            index=i;

        }

        }

     }

      printf("Highest 1s row is :%d with number :%d", index, max);

}