Assignment - 16 (Multi-Dimensional Array in C Language)

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

#include<stdio.h>

int main()

{

int i, j, a[3][3], b[3][3];

printf("Enter elements of first matrix of order 3X3: ");

for(i = 0; i < 3; i++)

{

for(j = 0; j < 3; j++)

scanf("%d", &a[i][j]);

}

printf("Enter elements of second matrix of order 3X3: ");

for(i = 0; i < 3; i++)

{

for(j = 0; j < 3; j++)

scanf("%d", &b[i][j]);

}

printf("Sum of two matrices is:\n");

for(i = 0; i < 3; i++)

{

for(j = 0; j < 3; j++)

printf("%d\t", a[i][j] + b[i][j]);

printf("\n");

}

return 0;

}

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

#include<stdio.h>

int main()

{

int a[3][3], b[3][3], i, j, k, pro[3][3] = {{0}};

printf("Enter elements of first matrix of order 3X3: ");

for(i = 0; i < 3; i++)

{

for(j = 0; j < 3; j++)

scanf("%d", &a[i][j]);

}

printf("Enter elements of second matrix of order 3X3: ");

for(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++)

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

printf("Product of 2 matrices each of order 3X3 is: ");

for(i = 0; i < 3; i++)

{

for(j = 0; j < 3; j++)

printf("%d\t", pro[i][j]);

printf("\n");

}

return 0;

}

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

#include<stdio.h>

int main()

{

int i, j, arr[3][3], transpose[3][3];

printf("Enter elements of matrix of order 3X3: ");

for(i = 0; i < 3; i++)

{

for(j = 0; j < 3; j++)

scanf("%d", &arr[i][j]);

}

for(i = 0; i < 3; i++)

{

for(j = 0; j < 3; j++)

{

transpose[i][j] = arr[j][i];

}

}

printf("Transpose of matrix is:\n");

for(i = 0; i < 3; i++)

{

for(j = 0; j < 3; j++)

{

printf("%d\t", transpose[i][j]);

}

printf("\n");

}

return 0;

}

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

#include<stdio.h>

int main()

{

int arr[3][3], i, j, rightDiagonalElementsSum = 0;

printf("Enter elements of matrix of order 3X3: ");

for(i = 0; i < 3; i++)

{

for(j = 0; j < 3; j++)

scanf("%d", &arr[i][j]);

}

for(i = 0; i < 3; i++)

{

for(j = 0; j < 3; j++)

{

if(i + j == 2)

{

rightDiagonalElementsSum = rightDiagonalElementsSum + arr[i][j];

}

}

}

printf("Sum of right diagonal elements of a matrix is %d.", rightDiagonalElementsSum);

return 0;

}

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

#include<stdio.h>

int main()

{

int arr[3][3], i, j, leftDiagonalElementsSum = 0;

printf("Enter elements of matrix of order 3X3: ");

for(i = 0; i < 3; i++)

{

for(j = 0; j < 3; j++)

scanf("%d", &arr[i][j]);

}

for(i = 0; i < 3; i++)

{

for(j = 0; j < 3; j++)

{

if(i == j)

{

leftDiagonalElementsSum = leftDiagonalElementsSum + arr[i][j];

}

}

}

printf("Sum of left diagonal elements of a matrix is %d.", leftDiagonalElementsSum);

return 0;

}

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

#include<stdio.h>

int main()

{

int matrix[3][3], i, j, rowSum, columnSum;

printf("Enter elements of matrix of order 3X3: ");

for(i = 0; i < 3; i++)

{

for(j = 0; j < 3; j++)

scanf("%d", &matrix[i][j]);

}

for(i = 0; i < 3; i++)

{

rowSum = 0;

columnSum = 0;

for(j = 0; j < 3; j++)

{

rowSum += matrix[i][j];

columnSum += matrix[j][i];

}

printf("Sum of elements of row %d is %d.\n", i + 1, rowSum);

printf("Sum of elements of column %d is %d.\n", i + 1, columnSum);

}

return 0;

}

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

#include<stdio.h>

int main()

{

int matrix[3][3], i, j;

printf("Enter the elements of a matrix of order 3X3: ");

for(i = 0; i < 3; i++)

{

for(j = 0; j < 3; j++)

scanf("%d", &matrix[i][j]);

}

printf("The lower triangular of this matrix is:\n");

for(i = 0; i < 3; i++)

{

for(j = 0; j < 3; j++)

{

if(j <= i)

printf("%d\t", matrix[i][j]);

}

printf("\n");

}

return 0;

}

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

#include<stdio.h>

int main()

{

int matrix[3][3], i, j;

printf("Enter the elements of a matrix of order 3X3: ");

for(i = 0; i < 3; i++)

{

for(j = 0; j < 3; j++)

scanf("%d", &matrix[i][j]);

}

printf("The upper triangular matrix of this matrix is:\n");

for(i = 0; i < 3; i++)

{

for(j = 0; j < 3; j++)

{

if(j >= i)

printf("%d\t", matrix[i][j]);

else

printf("0\t");

}

printf("\n");

}

return 0;

}

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

#include<stdio.h>

int main()

{

int matrix[3][3], i, j, zeroesCount = 0;

printf("Enter the elements of a matrix of order 3X3: ");

for(i = 0; i < 3; i++)

{

for(j = 0; j < 3; j++)

scanf("%d", &matrix[i][j]);

}

for(i = 0; i < 3; i++)

{

for(j = 0; j < 3; j++)

{

if(matrix[i][j] == 0)

zeroesCount++;

}

}

if(zeroesCount > 4) // if more than half of the matrix elements are zero, then matrix is considered as sparse matrix.

printf("Matrix is a sparse matrix.");

else

printf("Matrix is not sparse matrix.");

return 0;

}

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

#include<stdio.h>

int main()

{

int matrix[3][3], i, j, maximumOnes = 0, onesCount, maximumOnesRow = -1;

printf("Enter the elements of a matrix of order 3X3: ");

for(i = 0; i < 3; i++)

{

for(j = 0; j < 3; j++)

scanf("%d", &matrix[i][j]);

}

for(i = 0; i < 3; i++)

{

onesCount = 0;

for(j = 0; j < 3; j++)

{

if(matrix[i][j] == 1)

onesCount++;

}

if(onesCount > maximumOnes)

{

maximumOnes = onesCount;

maximumOnesRow = i + 1;

}

}

if(maximumOnesRow == -1)

printf("No element in the matrix has value 1.\n");

else

printf("Row with maximum number of 1s is Row %d.", maximumOnesRow);

return 0;

}