```c
/*

Que : 1.Write C Program to read and print a Matrix , R and C must be input by User.
(Using Static Memory Allocation).
Owner: Rushikesh Sanjay Pokharkar
Batch: PPA9

*/

//                                      *********  Solution  *********


#include<stdio.h> //Include Necessary Header Files.

void main() {
	int a[20][20], row, col; // Initialize required variables

	printf("Enter Number Of rows:");
	scanf_s("%d", &row); // Take input - Number of array elements in row.

	printf("Enter Number of columns:");
	scanf_s("%d", &col); // Take input - Number of array elements in columns.

	printf("Enter array Elements...\n");

	for (int i = 0; i < row; i++) // For loop to take input array elements.
	{
		for (int j = 0; j < col; j++)
		{
			scanf_s("%d", &a[i][j]);
		}
	}

	printf("Array Elements are: \n");
	for (int i = 0; i < row; i++) // For loop to print array elements.
	{
		for (int j = 0; j < col; j++)
		{
			printf("%d\t", a[i][j]);
		}
		printf("\n");
	}
}




/*

Que : 2.Write a C Program to Search Element in a 2D Array (Using Static Memory
Allocation).
Owner: Rushikesh Sanjay Pokharkar
Batch: PPA9

*/

//                                      *********  Solution  *********


#include<stdio.h> //Include Necessary Header Files.

void main() {
```

```
	int arr[100][100], row, col; // Initialize required variables

	printf("Enter Number Of rows:");
	scanf_s("%d", &row); // Take input - Number of array elements in row.

	printf("Enter Number of columns:");
	scanf_s("%d", &col); // Take input - Number of array elements in columns.

	printf("Enter array Elements...\n");

	for (int i = 0; i < row; i++) // For loop to take input array elements.
	{
		for (int j = 0; j < col; j++)
		{
			scanf_s("%d", &arr[i][j]);
		}
	}

	printf("Array Elements are: \n");
	for (int i = 0; i < row; i++) // For loop to print array elements.
	{
		for (int j = 0; j < col; j++)
		{
			printf("%d\t", arr[i][j]);
		}
		printf("\n");
	}

	// Logic to search Element in 2D array using SMA.
	int search, flag = 0;
	printf("Enter the Search Element: ");
	scanf_s("%d", &search);

	for (int i = 0; i < row; i++) // For loop to search element in array.
	{
		for (int j = 0; j < col; j++)
		{
			if (arr[i][j] == search) // If element found then break the
loop..
			{
				flag = 1;
				break;
			}
		}
		if (flag == 1) // If element found then break the loop..
		{
			break;
		}
	}
	if (flag == 1) {
		printf("The given element %d is Present in array..", search);
	}
	else {
		printf("The given element %d is Not Present in array..", search);
	}
}



/*
```

```
Que : 3.Write a C Program to find the transpose of a given matrix (Using Static Memory
Allocation).
Owner: Rushikesh Sanjay Pokharkar
Batch: PPA9

*/

//                                        ********  Solution  ********


#include<stdio.h> //Include Necessary Header Files.

void main() {
	int arr[100][100], row, col; // Initialize required variables

	printf("Enter Number Of rows:");
	scanf_s("%d", &row); // Take input - Number of array elements in row.

	printf("Enter Number of columns:");
	scanf_s("%d", &col); // Take input - Number of array elements in columns.

	printf("Enter array Elements...\n");

	for (int i = 0; i < row; i++) // For loop to take input array elements.
	{
		for (int j = 0; j < col; j++)
		{
			scanf_s("%d", &arr[i][j]);
		}
	}

	printf("Array Elements are: \n");
	for (int i = 0; i < row; i++) // For loop to print array elements.
	{
		for (int j = 0; j < col; j++)
		{
			printf("%d\t", arr[i][j]);
		}
		printf("\n");
	}

	// Logic for transpose of a matrix.
	printf("The Transpose of given matrix is:\n");
	for (int i = 0; i < row; i++) // For loop to print transpose of matrix..
	{
		for (int j = 0; j < col; j++)
		{
			printf("%d\t", arr[j][i]);
		}
		printf("\n");
	}

}




/*

Que : 4. Write a C program to add two matrices in third Matrix(Using Static Memory
Allocation).
Owner: Rushikesh Sanjay Pokharkar
```

```
Batch: PPA9

*/

//                                        *********  Solution  *********


#include<stdio.h> //Include Necessary Header Files.

void main() {
	int arr1[20][20], arr2[20][20], add[20][20], row, col; // Initialize required
variables

	printf("Enter Number Of Rows:");
	scanf_s("%d", &row); // Take input - Number of array elements in row.

	printf("Enter Number Of Columns:");
	scanf_s("%d", &col); // Take input - Number of array elements in columns.

	printf("Enter array Elements of first matrix...\n");
	for (int i = 0; i < row; i++) // For loop to take input array elements.
	{
		for (int j = 0; j < col; j++)
		{
			scanf_s("%d", &arr1[i][j]);
		}
	}

	printf("Enter array Elements of second matrix...\n");
	for (int i = 0; i < row; i++) // For loop to take input array elements.
	{
		for (int j = 0; j < col; j++)
		{
			scanf_s("%d", &arr2[i][j]);
		}
	}

	printf("First array is: \n");
	for (int i = 0; i < row; i++) // For loop to print array elements.
	{
		for (int j = 0; j < col; j++)
		{
			printf("%d\t", arr1[i][j]);
		}
		printf("\n");
	}

	printf("Second array is: \n");
	for (int i = 0; i < row; i++) // For loop to print array elements.
	{
		for (int j = 0; j < col; j++)
		{
			printf("%d\t", arr2[i][j]);
		}
		printf("\n");
	}

	// Logic for addition of two matrices...
	printf("The addition of given two matrices is:\n");
	for (int i = 0; i < row; i++) // For loop to add two matrices..
	{
		for (int j = 0; j < col; j++)
```

```
			{
				add[i][j] = arr1[i][j] + arr2[i][j];
			}
		}

		for (int i = 0; i < row; i++) // For loop to print addition matrix.
		{
			for (int j = 0; j < col; j++)
			{
				printf("%d\t", add[i][j]);
			}
			printf("\n");
		}

}




/*

Que : 5. Write a C program to subtract two matrices in third matrix (Using Static
memory Allocation).
Owner: Rushikesh Sanjay Pokharkar
Batch: PPA9

*/

//                                        *********  Solution  *********


#include<stdio.h> //Include Necessary Header Files.

void main() {
	int arr1[20][20], arr2[20][20], sub[20][20], row, col; // Initialize required
variables

	printf("Enter Number Of Rows:");
	scanf_s("%d", &row); // Take input - Number of array elements in row.

	printf("Enter Number Of Columns:");
	scanf_s("%d", &col); // Take input - Number of array elements in columns.

	printf("Enter array Elements of first matrix...\n");
	for (int i = 0; i < row; i++) // For loop to take input array elements.
	{
		for (int j = 0; j < col; j++)
		{
			scanf_s("%d", &arr1[i][j]);
		}
	}

	printf("Enter array Elements of second matrix...\n");
	for (int i = 0; i < row; i++) // For loop to take input array elements.
	{
		for (int j = 0; j < col; j++)
		{
			scanf_s("%d", &arr2[i][j]);
		}
	}

	printf("First array is: \n");
```

```
	for (int i = 0; i < row; i++) // For loop to print array elements.
	{
		for (int j = 0; j < col; j++)
		{
			printf("%d\t", arr1[i][j]);
		}
		printf("\n");
	}

	printf("Second array is: \n");
	for (int i = 0; i < row; i++) // For loop to print array elements.
	{
		for (int j = 0; j < col; j++)
		{
			printf("%d\t", arr2[i][j]);
		}
		printf("\n");
	}

	// Logic for Substraction of two matrices...
	printf("The addition of given two matrices is:\n");
	for (int i = 0; i < row; i++) // For loop to substract two matrices..
	{
		for (int j = 0; j < col; j++)
		{
			sub[i][j] = arr1[i][j] - arr2[i][j];
		}
	}

	for (int i = 0; i < row; i++) // For loop to print substraction matrix.
	{
		for (int j = 0; j < col; j++)
		{
			printf("%d\t", sub[i][j]);
		}
		printf("\n");
	}

}




/*

Que : 6. Write a c program to check whether given matrix is upper triangular or not
(Using Static Memory Allocation).
Owner: Rushikesh Sanjay Pokharkar
Batch: PPA9

*/

//                                        *********  Solution  *********


#include<stdio.h> //Include Necessary Header Files.

void main() {
	int arr[20][20], row, col; // Initialize required variables

	printf("Enter Number Of rows:");
	scanf_s("%d", &row); // Take input - Number of array elements in row.
```

```
	printf("Enter Number of columns:");
	scanf_s("%d", &col); // Take input - Number of array elements in columns.

	printf("Enter array Elements...\n");

	for (int i = 0; i < row; i++) // For loop to take input array elements.
	{
		for (int j = 0; j < col; j++)
		{
			scanf_s("%d", &arr[i][j]);
		}
	}

	printf("Array Elements are: \n");
	for (int i = 0; i < row; i++) // For loop to print array elements.
	{
		for (int j = 0; j < col; j++)
		{
			printf("%d\t", arr[i][j]);
		}
		printf("\n");
	}


	// Logic to check given matrix is upper triangular or not..
	int flag = 0;
	for (int i = 1; i < row; i++) {
		for (int j = 0; j < i; j++) {
			if (arr[i][j] != 0) {
				flag = 1;
				break;
			}
		}
		if (flag == 1) {
			break;
		}
	}

	if (flag == 1) {
		printf("The given matrix is not an upper triangular matrix.");
	}
	else {
		printf("The given matrix is an upper triangular matrix.");
	}
}




/*

Que : 7. Write a C program to check whether given matrix is lower triangular or not
(Using Static Memory Allocation).
Owner: Rushikesh Sanjay Pokharkar
Batch: PPA9

*/

//                                      *********  Solution  *********
```

```c
#include<stdio.h> //Include Necessary Header Files.

void main() {
	int arr[20][20], row, col; // Initialize required variables

	printf("Enter Number Of rows:");
	scanf_s("%d", &row); // Take input - Number of array elements in row.

	printf("Enter Number of columns:");
	scanf_s("%d", &col); // Take input - Number of array elements in columns.

	printf("Enter array Elements...\n");

	for (int i = 0; i < row; i++) // For loop to take input array elements.
	{
		for (int j = 0; j < col; j++)
		{
			scanf_s("%d", &arr[i][j]);
		}
	}

	printf("Array Elements are: \n");
	for (int i = 0; i < row; i++) // For loop to print array elements.
	{
		for (int j = 0; j < col; j++)
		{
			printf("%d\t", arr[i][j]);
		}
		printf("\n");
	}


	// Logic to check given matrix is lower triangular or not..
	int flag = 0;
	for (int i = 1; i < row; i++) {
		for (int j = 0; j < i; j++) {
			if (arr[j][i] != 0) {
				flag = 1;
				break;
			}
		}
		if (flag == 1) {
			break;
		}
	}

	if (flag == 1) {
		printf("The given matrix is not an lower triangular matrix.");
	}
	else {
		printf("The given matrix is an lower triangular matrix.");
	}
}




/*

Que : 8. Write C Program to Check if a given Matrix is an Unit Matrix. (Using Static
Memory Allocation).
Owner: Rushikesh Sanjay Pokharkar
```

```
Batch: PPA9

*/

//                                        *********  Solution  *********


#include<stdio.h> //Include Necessary Header Files.

void main() {
	int arr[20][20], row, col; // Initialize required variables

	printf("Enter Number Of rows:");
	scanf_s("%d", &row); // Take input - Number of array elements in row.

	printf("Enter Number of columns:");
	scanf_s("%d", &col); // Take input - Number of array elements in columns.

	printf("Enter array Elements...\n");

	for (int i = 0; i < row; i++) // For loop to take input array elements.
	{
		for (int j = 0; j < col; j++)
		{
			scanf_s("%d", &arr[i][j]);
		}
	}

	printf("Array Elements are: \n");
	for (int i = 0; i < row; i++) // For loop to print array elements.
	{
		for (int j = 0; j < col; j++)
		{
			printf("%d\t", arr[i][j]);
		}
		printf("\n");
	}


	// Logic to check given matrix is unit matrix or not..
	int flag = 0;
	for (int i = 0; i < row; i++)
	{
		for (int j = 0; j < col; j++)
		{
			if (i == j) {
				if (arr[i][j] != 1) {
					flag = 1;
					break;
				}
			}
			else {
				if (arr[i][j] != 0) {
					flag = 1;
					break;
				}
			}
		}
		if (flag == 1) {
			break;
		}
	}
```

```
	if (flag == 1) {
		printf("The given matrix is not an unit matrix.");
	}
	else {
		printf("The given matrix is an unit matrix.");
	}

}




/*

Que : 9. Write a C Program to check whether a given matrix is an identity matrix or
not (Using Static Memory Allocation).
Owner: Rushikesh Sanjay Pokharkar
Batch: PPA9

*/

//                                        *********  Solution  *********


#include<stdio.h> //Include Necessary Header Files.

void main() {
	int arr[20][20], row, col; // Initialize required variables

	printf("Enter Number Of rows:");
	scanf_s("%d", &row); // Take input - Number of array elements in row.

	printf("Enter Number of columns:");
	scanf_s("%d", &col); // Take input - Number of array elements in columns.

	printf("Enter array Elements...\n");

	for (int i = 0; i < row; i++) // For loop to take input array elements.
	{
		for (int j = 0; j < col; j++)
		{
			scanf_s("%d", &arr[i][j]);
		}
	}

	printf("Array Elements are: \n");
	for (int i = 0; i < row; i++) // For loop to print array elements.
	{
		for (int j = 0; j < col; j++)
		{
			printf("%d\t", arr[i][j]);
		}
		printf("\n");
	}


	// Logic to check given matrix is Identity matrix or not..
	int flag = 0;
	for (int i = 0; i < row; i++)
	{
		for (int j = 0; j < col; j++)
```

```
			{
				if (i == j) {
					if (arr[i][j] != 1) {
						flag = 1;
						break;
					}
				}
				else {
					if (arr[i][j] != 0) {
						flag = 1;
						break;
					}
				}
			}
			if (flag == 1) {
				break;
			}
		}

		if (flag == 1) {
			printf("The given matrix is not an Identity matrix.");
		}
		else {
			printf("The given matrix is an Identity matrix.");
		}
}




/*

Que : 10. Write C program to check if the matrix is symmetric or not (Using Static
Memory Allocation).
Owner: Rushikesh Sanjay Pokharkar
Batch: PPA9

*/

//                                        *********  Solution  *********


#include<stdio.h> //Include Necessary Header Files.

void main() {
		int arr[20][20], row, col; // Initialize required variables

		printf("Enter Number Of rows:");
		scanf_s("%d", &row); // Take input - Number of array elements in row.

		printf("Enter Number of columns:");
		scanf_s("%d", &col); // Take input - Number of array elements in columns.

		printf("Enter array Elements...\n");

		for (int i = 0; i < row; i++) // For loop to take input array elements.
		{
			for (int j = 0; j < col; j++)
			{
				scanf_s("%d", &arr[i][j]);
			}
		}
```

```
	printf("Array Elements are: \n");
	for (int i = 0; i < row; i++) // For loop to print array elements.
	{
		for (int j = 0; j < col; j++)
		{
			printf("%d\t", arr[i][j]);
		}
		printf("\n");
	}


	// Logic to check given matrix is Symmitric matrix or not..
	int flag = 0;
	for (int i = 0; i < row; i++)
	{
		for (int j = 0; j < col; j++)
		{
			if (arr[i][j] != arr[j][i]) {
				flag = 1;
				break;
			}
		}
		if (flag == 1) {
			break;
		}
	}
	if (flag == 1) {
		printf("The given matrix is not an symmitric matrix");
	}
	else {
		printf("The given matrix is an symmitric matrix");
	}
}
```