**Exercise**

**Theory Questions**

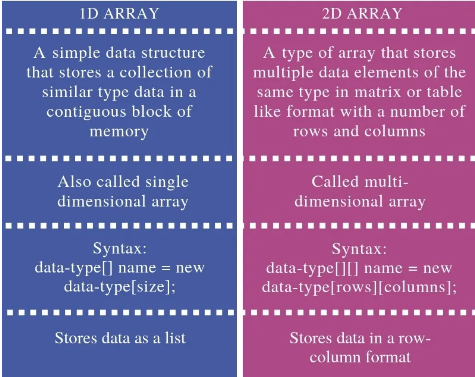
1. What is array?

Answer::

An array stores multiple values in a single variable. Arrays are a kind of data structure that can store a fixed-size sequential collection of elements of the same type (homogeneous) values. An array is used to store a collection of data, but it is often more useful to think of an array as a collection of variables.

1. What is the difference between a single dimensional array and a multidimensional array?

Answer::



1. Describe the two (2D) dimension array.

Answer::

**Two (2D) dimensional** **Array**

The two dimensional arrays are such type of arrays which stores an multiple column at each row and column index number instead of single element. It can be created using nested array. These type of arrays can be used to store similar type of elements, but the index is always a number.

**Practical Questions**

1. What would be the output of the following Program?

void main()**{**

**int** salary**[ ]** = **{**50000,40000,20000,30000,60000**}**;

**for** (**int** i=4; i>=0; i--)

**{**

***printf***( " Employee %d , and Salary is %d \n ",i, salary**[** i **]**);

**}**

**}**

**Answer::**

**OUTPUT**

Employee 4, and Salary is 60000

Employee 3, and Salary is 30000

Employee 2, and Salary is 20000

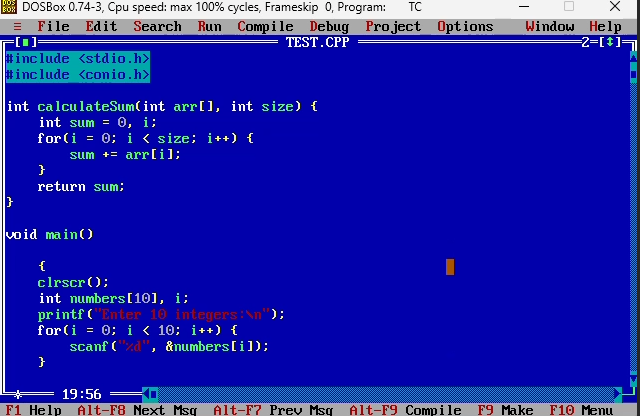
Employee 1, and Salary is 40000

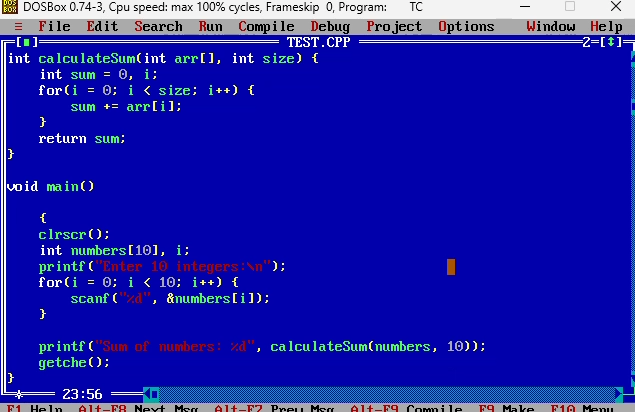
Employee 0, and Salary is 50000

1. Write a program that takes 10 integers as input and prints their sum by using array.

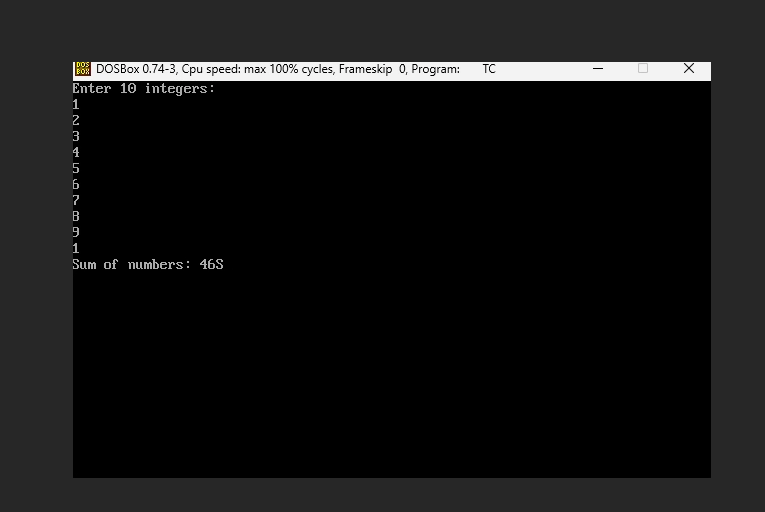
Answer::

INPUT





OUTPUT



1. Create a Mark-Sheet for 10 students using arrays which contains the following;

Roll Number

Name

Marks of 3 subjects ( Maths, English and Urdu)

and display following form

Student List Of Mark sheet

----------------------------------------------------------------------------------------------------------------------

Roll # | Student Name | Math’s | English | Urdu | Obtain Marks | Percentage | Grade |

99 | xxxxxxxxxxxx | 99 | 99 | 99 | 999 | 99.99 | xx |

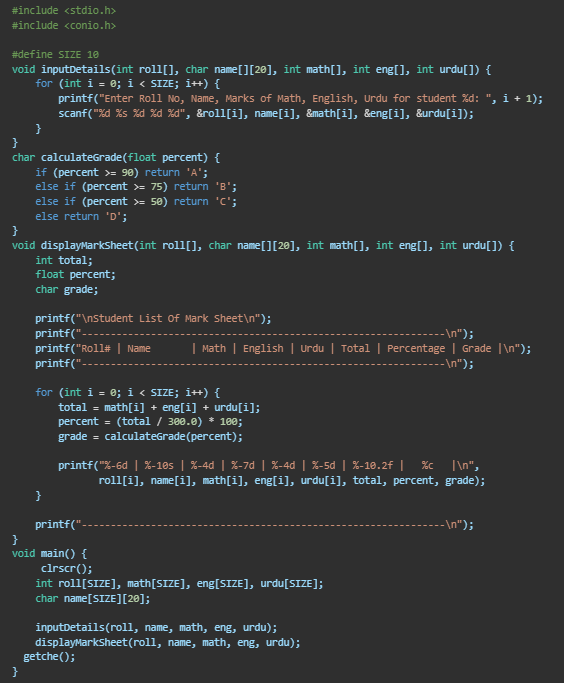
99 | xxxxxxxxxxxx | 99 | 99 | 99 | 999 | 99.99 | xx |

99 | xxxxxxxxxxxx | 99 | 99 | 99 | 999 | 99.99 | xx |

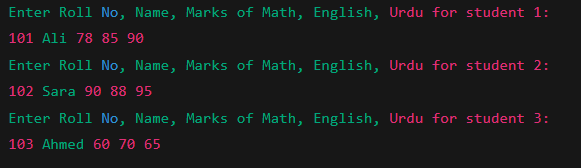
:

Answer::

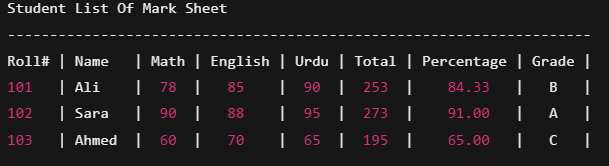
PROGRAM



INPUT



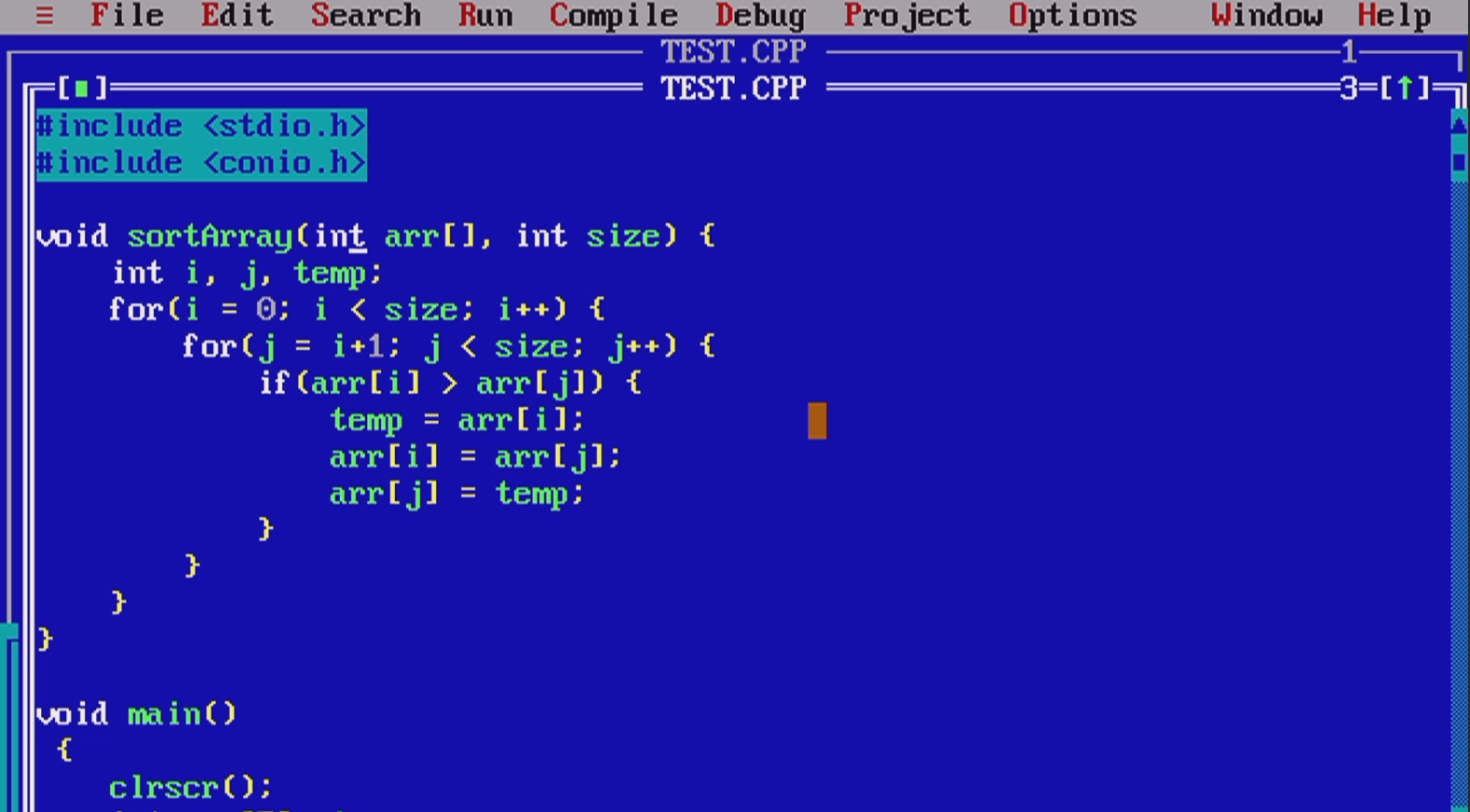
OUTPUT



1. Write a program to sort an integer array in ascending order.

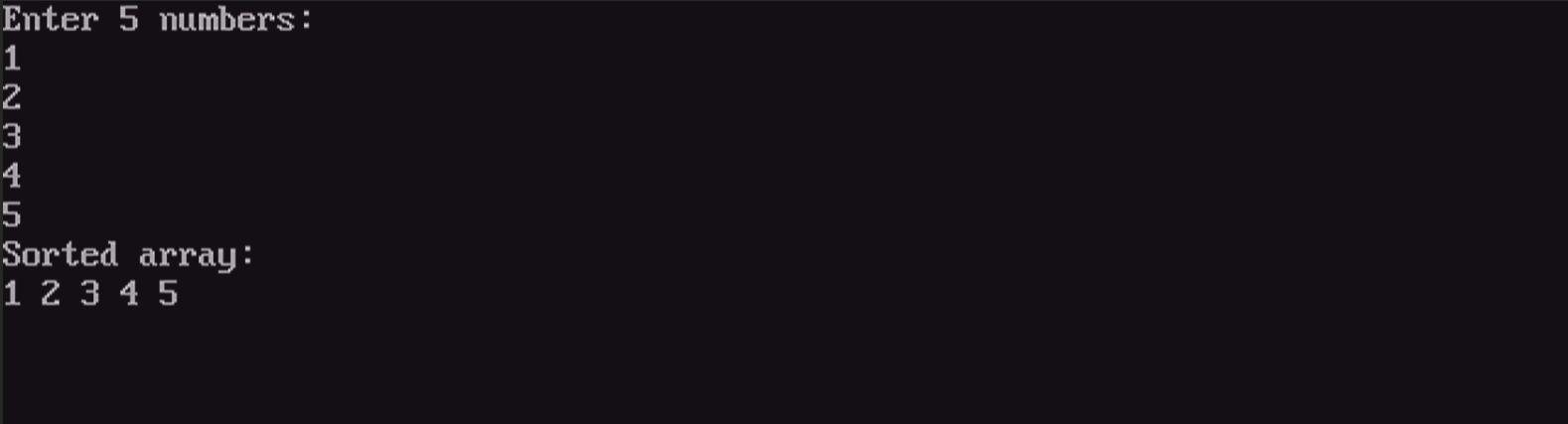
ANSWER::

PROGRAM:



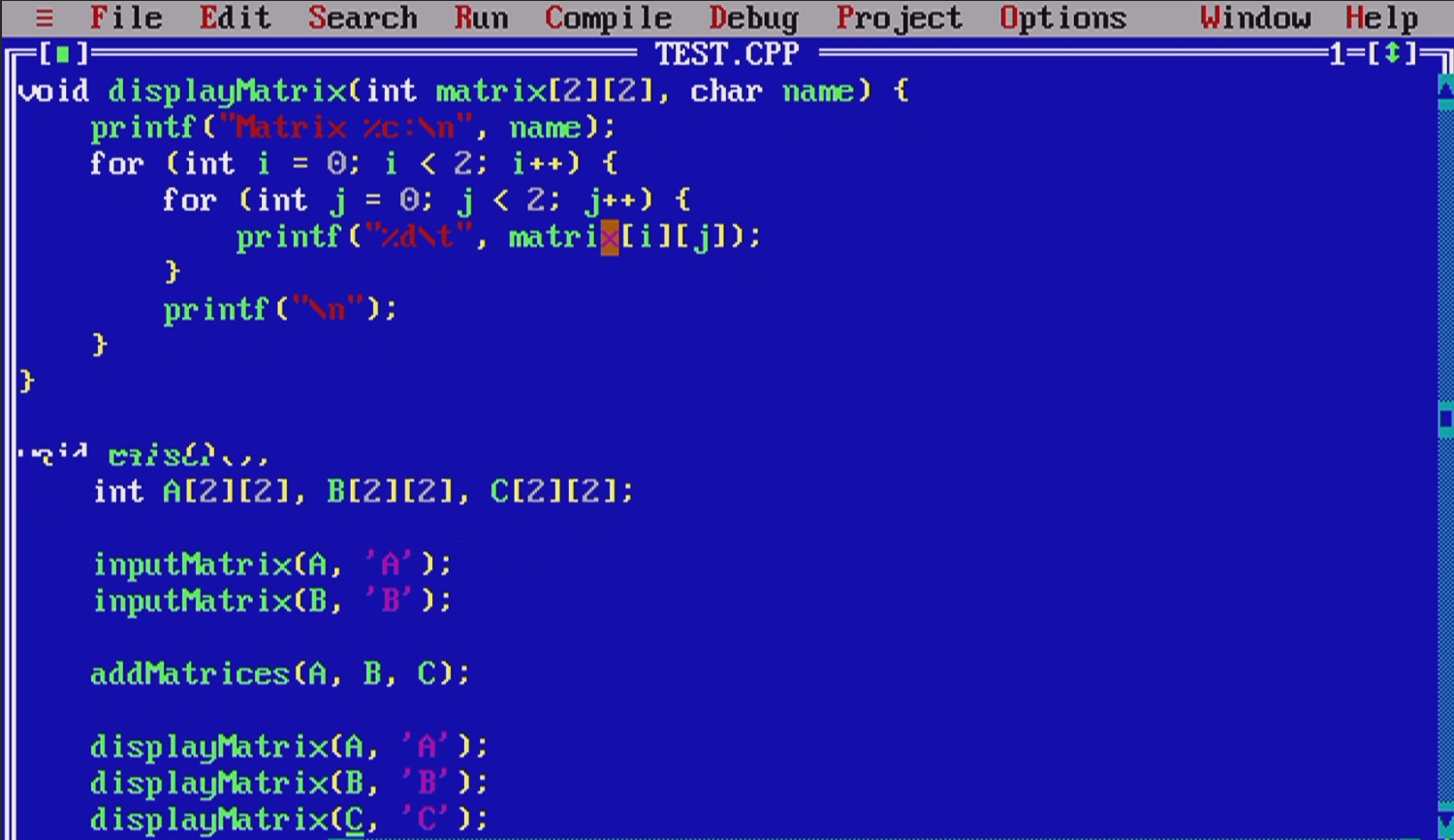
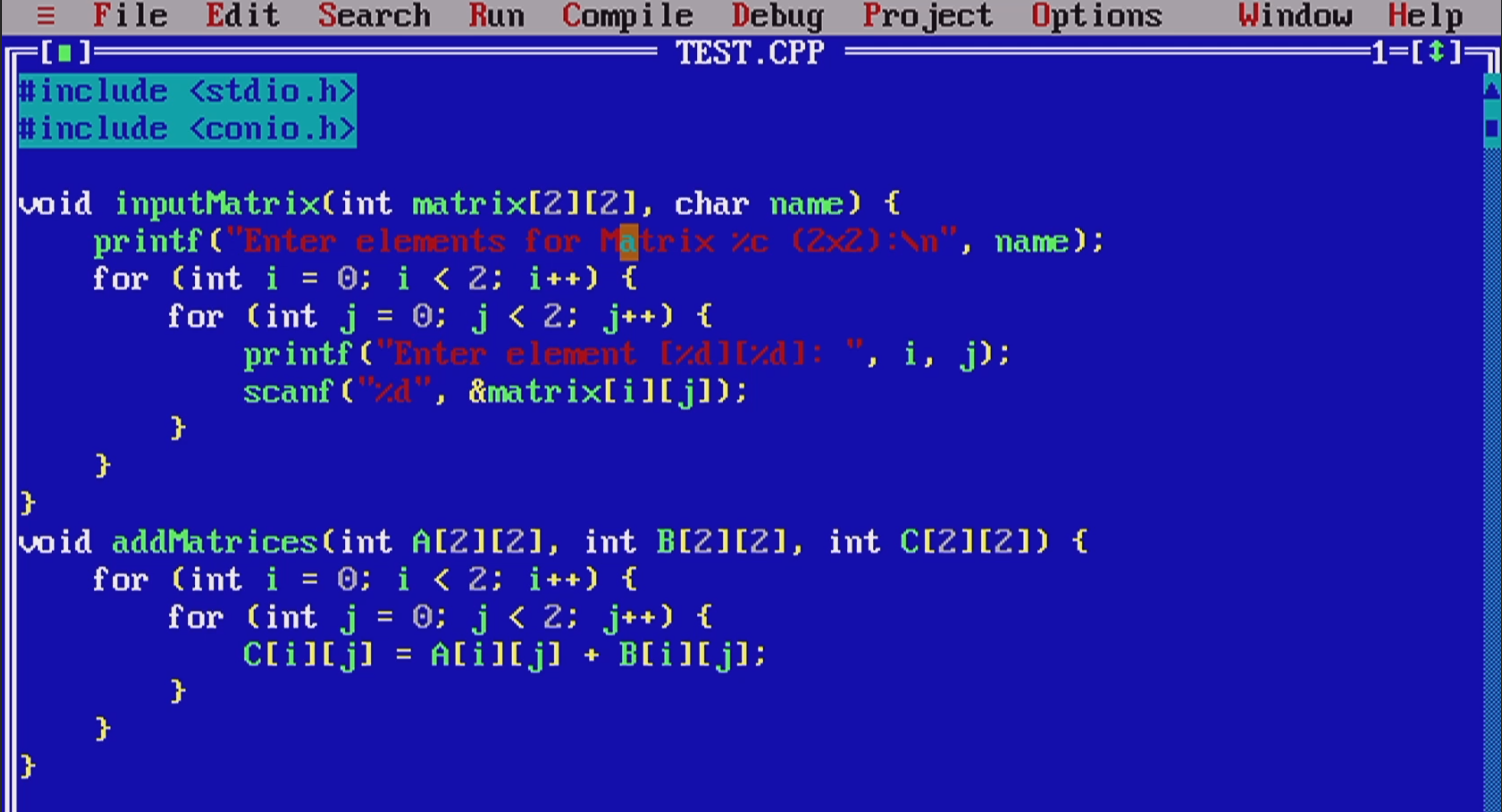


OUTPUT:

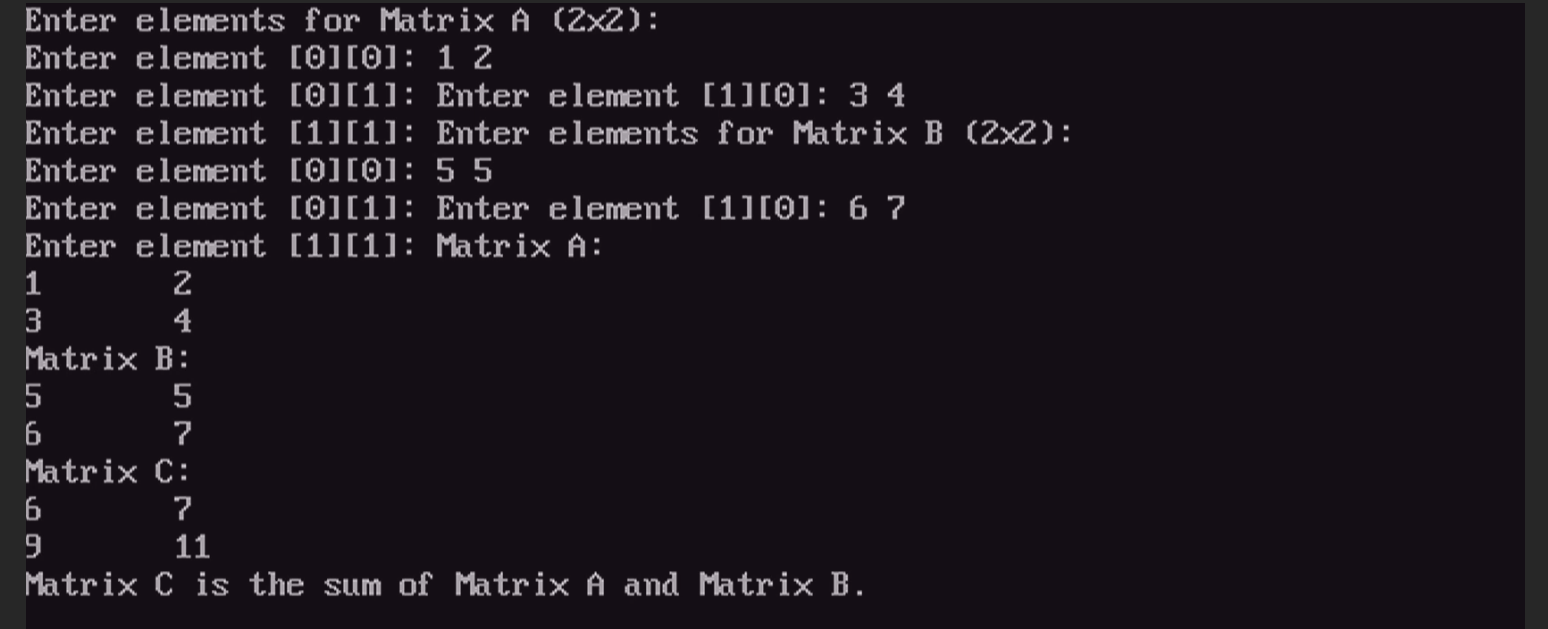


1. Write a program that adds up two 2x2 arrays A and B stores the sum in third array C.

ANSWER::

PROGRAM::  


OUTPUT::

****

**Objective MCQ’s**

1. Which declaration of array is correct?
2. **int** Number**[** **]**;
3. Number **int[** **]**;
4. **int Number[ 5 ];**
5. **int** Number(5);
6. The first element number start always from \_\_\_ index number.
7. 1
8. **0**
9. 2
10. NULL
11. What is the correct syntax for declaring and initializing an associative array?
12. **char** Names**[** **][** **]** = array**{**"Asif" - "Muhammad" - "Kamran" - "Umer Ahmed"**}**;
13. **char** Names**[**4**][**10 **]** = **{**Asif , Muhammad , Kamran , Umer Ahmed**}**;
14. **char** Names**[**4**][**10 **]** = ("Asif" , "Muhammad" , "Kamran" , "Umer Ahmed");
15. **char Names[4][10 ] = {"Asif", "Muhammad", "Kamran", "Umer Ahmed"};**
16. When you pass an array as an argument to a function, what is actually passed?
17. The values of all elements in array
18. The address of the array
19. **The address of the first element in the array**
20. The value of first element in the array
21. The following declares a two dimensional float array name ***No*** with 5 rows and 3 column
22. **float** No**[**3,5**]**;
23. **float** No**[**5,3**]**;
24. **float** No**[**3**][**5**]**;
25. **float No[5][3];**
26. An array is a collection of variable’s in C/C++
27. Different data types scattered throughout memory.
28. Similar data types scattered throughout memory.
29. **Similar data types place continuously in memory.**
30. Different data types place continuously in memory.
31. Which is last element of array, if we declared integer array **int** Salary **[**10**]**; ?
32. Salary**[**0**]**;
33. Salary**[**1**]**;
34. Salary**[**10 **]**;
35. **Salary[9];**