

## EXERCISES

### Review Questions

1. What are arrays and why are they needed?
2. How is an array represented in the memory?
3. How is a two-dimensional array represented in the memory?
4. What is the use of multi-dimensional arrays?
5. Explain sparse matrix.
6. How are pointers used to access two-dimensional arrays?
7. Why does storing of sparse matrices need extra consideration? How are sparse matrices stored efficiently in the computer's memory?
8. For an array declared as `int arr[50]`, calculate the address of `arr[35]`, if `Base(arr) = 1000` and `w = 2`.
9. Consider a two-dimensional array `Marks[10][5]` having its base address as 2000 and the number of bytes per element of the array is 2. Now, compute the address of the element, `Marks[8][5]`, assuming that the elements are stored in row major order.
10. How are arrays related to pointers?
11. Briefly explain the concept of array of pointers.
12. How can one-dimensional arrays be used for inter-function communication?
13. Consider a two-dimensional array `arr[10][10]` which has base address = 1000 and the number of bytes per element of the array = 2. Now, compute the address of the element `arr[8][5]` assuming that the elements are stored in column major order.
14. Consider the array given below:

Name[0]	Adam
Name[1]	Charles
Name[2]	Dicken
Name[3]	Esha
Name[4]	Georgia
Name[5]	Hillary
Name[6]	Mishaal



- (a) How many elements would be moved if the name Andrew has to be added in it?  
 (i) 7 (ii) 4  
 (iii) 5 (iv) 6
- (b) How many elements would be moved if the name Esha has to be deleted from it?  
 (i) 3 (ii) 4  
 (iii) 5 (iv) 6
15. What happens when an array is initialized with  
 (a) fewer initializers as compared to its size?  
 (b) more initializers as compared to its size?

### Programming Exercises

- Consider an array `MARKS[20][5]` which stores the marks obtained by 20 students in 5 subjects. Now write a program to
  - find the average marks obtained in each subject.
  - find the average marks obtained by every student.
  - find the number of students who have scored below 50 in their average.
  - display the scores obtained by every student.
- Write a program that reads an array of 100 integers. Display all the pairs of elements whose sum is 50.
- Write a program to interchange the second element with the second last element.
- Write a program that calculates the sum of squares of the elements.
- Write a program to compute the sum and mean of the elements of a two-dimensional array.
- Write a program to read and display a square (using functions).
- Write a program that computes the sum of the elements that are stored on the main diagonal of a matrix using pointers.
- Write a program to add two  $3 \times 3$  matrix using pointers.
- Write a program that computes the product of the elements that are stored on the diagonal above the main diagonal.
- Write a program to count the total number of non-zero elements in a two-dimensional array.
- Write a program to input the elements of a two-dimensional array. Then from this array, make two arrays—one that stores all odd elements of the two-dimensional array and the other that stores all even elements of the array.
- Write a program to read two floating point number arrays. Merge the two arrays and display the resultant array in reverse order.
- Write a program using pointers to interchange the second biggest and the second smallest number in the array.
- Write a menu driven program to read and display a  $p \times q \times r$  matrix. Also, find the sum, transpose, and product of the two  $p \times q \times r$  matrices.
- Write a program that reads a matrix and displays the sum of its diagonal elements.
- Write a program that reads a matrix and displays the sum of the elements above the main diagonal. (Hint: Calculate the sum of elements  $A_{ij}$  where  $i < j$ )
- Write a program that reads a matrix and displays the sum of the elements below the main diagonal. (Hint: Calculate the sum of elements  $A_{ij}$  where  $i > j$ )
- Write a program that reads a square matrix of size  $n \times n$ . Write a function `int isUpperTriangular(int a[][], int n)` that returns 1 if the matrix is upper triangular. (Hint: Array  $A$  is upper triangular if  $A_{ij} = 0$  and  $i > j$ )
- Write a program that reads a square matrix of size  $n \times n$ . Write a function `int isLowerTriangular(int a[][], int n)` that returns 1 if the matrix is lower triangular. (Hint: Array  $A$  is lower triangular if  $A_{ij} = 0$  and  $i < j$ )
- Write a program that reads a square matrix of size  $n \times n$ . Write a function `int isSymmetric(int a[][], int n)` that returns 1 if the matrix is symmetric. (Hint: Array  $A$  is symmetric if  $A_{ij} = A_{ji}$  for all values of  $i$  and  $j$ )
- Write a program to calculate  $XA + YB$  where  $A$  and  $B$  are matrices and  $X=2$  and  $Y=3$ .
- Write a program to illustrate the use of a pointer that points to a 2D array.
- Write a program to enter a number and break it into  $n$  number of digits.
- Write a program to delete all the duplicate entries from an array of  $n$  integers.
- Write a program to read a floating point array. Update the array to insert a new number at the specified location.