Array

1. Explain Array with real time example.
2. Write down the advantages of array.
3. Explain disadvantages of array over dynamic memory allocation.
4. Explain types of arrays.
5. Explain one dimensional array with declaration, compile time initialization and run time initialization.
6. Explain two-dimensional array with initialization.
7. Explain row-major and column-major arrangement with example.
8. Explain multidimensional array with example.
9. Write a program to create an array as per given range by the user and display it.
10. Write a program to find maximum and minimum value from one dimensional array.
11. Write a program to find the sum and average of all the element of an array.
12. Write a program to sort an array in ascending order.
13. Write a program to search an element in given array.
14. What happens when an array with a specified size is assigned?
    a. With values fewer than the specified size, and
    b. With values more than specified size.
15. Write a program to count the even and odd numbers in an array.
16. Write a program to count negative, positive and zero in an array.
17. Explain how to pass individual array elements to an array.
18. Explain how to pass 1-D array to a function.
19. Write a program to take input from user in 2-D array and display it in matrix form.
20. Write a program for addition of two matrix and display.
21. Write a program for subtraction of two matrix and display
22. Write a program for multiplication of two matrix and display.
23. Write a program to find the transpose of a given matrix.
24. Explain arrays with more than two dimensions.
25. Write a program to count the occurrence of a number in a matrix.
26. Write a program to store all the element of a 2-D array in a 1-D array row-wise.

Function

27. Define function and concept of reusability.
28. How many types of function and explain with example?
29. What are the advantages of using function?
30. Explain function definition, function declaration and function call.
31. Distinguish local variable and global variable.
32. Distinguish local scope and global scope.
33. What is the role of actual argument and formal argument?
34. Categorized the function on the basis of arguments and return values.
    a. Functions with no arguments and no return values
    b. Functions with no arguments and a return value.
    c. Functions with arguments and no return value.
    d. Functions with arguments and a return value.
    e. Function returns multiple value.
35. Differentiate between call by value and call by reference using an example.
36. Define recursion with proper example.
37. Write a program to find the factorial of a given number using function.
38. Write a program to find given number is prime or not using function.
39. Write a program to find the reverse of a given number using function.
40. Write a program to print all the prime number less than 500 using function.
41. Write a program to find the factorial of a number by recursive method.
42. Write a program to print Fibonacci series by recursive method.
43. Explain advantages and disadvantages of recursion.
44. What is prototyping? Why is it necessary?
45. State the problems we are likely to encounter when we pass global variables as parameter to functions.
46. Enumerate the rules that apply to a function call.
47. Develop a modular interactive program using functions that reads the values of three sides of a triangle and displays either its area or its perimeter as per the request of the user. Given the three sides a, b and c
    Perimeter = a + b + c
    Area=$\sqrt{(s-a)(s-b)(s-c)}$
    Where s= (a + b + c)/2
48. Write a function that can be called to find the largest element of an m by n matrix.
49. Explain what is likely to happen when the following situations are encountered in a program.
    a. Actual arguments are less than the formal arguments in a function
    b. Data type of one of the actual arguments does not match with the type of the corresponding formal argument.

c. Data type of one of the arguments in a prototype does not match with the type of the corresponding formal parameter in the header line.
d. The order of actual parameters in the function call is different from the order of formal parameter in a function where all the parameter are of the same type.
e. The type of expression used in return statement does not match with the type of the functions.

50. Write a program to print pascal's triangle

```
1
1       1
1       2       1
1       3       3       1
1       4       6       4       1
```