Assignment 2

Chapter 4: Input & Output

1. What is the main **limitation of scanf()** function to read strings?
2. Explain about **input/output functions** available in C with the syntax and examples.
3. What is the **role of flags** in formatted output?
4. Why formatted output is required? Write a general format specification of printing different data types? Briefly discuss about each part.
5. What do you mean by **search set**? Explain format specification %[character] and %[^character]. What is the main advantage of using these specifications in reading string? Explain it with a suitable example.
6. Differentiate scanf() & gets().

Chapter 6: User-defined Functions

7. Write a recursive function to obtain the running sum of first 25 natural numbers.
8. A 5-digit positive integer is entered through the keyboard, write a function to calculate sum of digits of the 5-digit number:
    1. Using iteration
    2. Using recursion
9. Differentiate:
    1. Library Functions vs. User-defined Functions
    2. Function Definition vs. Function Prototype
    3. Call by Value vs Call by Reference
    4. Formal Arguments vs. Actual Arguments
    5. Recursion vs. Iteration
    6. Local vs. Global Variables
10. Explain why some problems can be solved either with or without recursion.
11. WAP to calculate $F = (a*b_n)/c!$, where n is integer. Implement two user-defined functions calcFactorial() and calcPower(). And display calculated value from main().
12. WAP to calculate sum & average of entered number, among entered number only calculate sum & avg of those number which is exactly divisible by 9 and not by 6.

Chapter 7: Arrays & Strings

13. WAP to calculate R = X*Y + Z_T where R, X, Y, Z are matrices of valid order and Z_T is the transpose of Z.
14. How can we access the elements of one-dimensional and two-dimensional arrays? Write a function that takes 1D array of 'n' elements and sort them in descending order. Display second largest & second smallest number.
15. WAP to find the frequency of a character in the string entered by user.
16. Write a program to read 7 names of students and sort them alphabetically. The process must be done by the user-defined function.
17. What is the role of string handling functions in program coding? Explain with examples.
18. WAP to read order of square matrix and its elements from keyboard. Find the sum of diagonal elements (trace) of the matrix.
19. WAP to reverse a string using a recursive function & display both strings.
20. WAP to merge two sorted arrays in another array in sorted order.

Chapter 8: Structure

21. Explain structure, structure member, nested structure, array of structure. How does a structure differ from an array? Explain way of accessing members of structure.
22. WAP to create structure named City that has mayor, population, earning, area as member. Assume appropriate types and size of members. Use this structure to read and display records of 5 cities.
23. Create a structure named Currency which has rupees & paisa as member. WAP to read two currencies from user and add them.
24. What are the main differences between structures and arrays?
25. WAP to create a structure named book which has name, page and prices as its members. Read name, page number and price of a book using structure variable. Finally display record of a book using a pointer to structure instead of structure itself to access member variables.
26. WAP to read name, emp_code, gender & salary of 4 employees using array of structure. Display name & emp_code of those employees whose salary is greater than 34,000.
27. Create a structure **Complex** containing real and imaginary as its members. WAP that uses above structure to input two complex numbers and pass to a function, which returns the sum of entered complex numbers in the main function.
28. Create a structure **Time** containing hour, minute & seconds for C-Programming practical as its members. WAP that uses this structure to input start & end times and display working period of lab.

*Note: - Complete your assignment in **separate/new copy**, and submit that **before deadline**.*

Last date of Submission: **Magh 29, 2076**