

Recursion Practice Sheet

1. Implement a **recursive** algorithm to find the sum of N elements.

For Example, N=3

A[]=[4, 7, 8]

Sum=4+7+8=19

2. Given an array of integers, write a **recursive** function to print it in reverse order.
3. Given a string, write a **recursive** function that checks if the given string is a palindrome or not.

Sample Input: "racecar"

Output: palindrome

Sample Input: "hello"

Output: not a palindrome

4. Given an integer number, count the total number of digits in that number using **recursion**.

Sample Input: n = 178

Output: No. of Digits = 3

5. Given two numbers, find the remainder using **recursion** when one number is divided by another without using % operator.

Sample Input: number = 31, divisor = 9

Output: remainder = 4

6. Write a **recursive** function that finds the Greatest Common Divisor of two numbers.

Sample Input: num1 = 10, num2 = 50

Output: GCD = 10

7. Given a string, insert a dash (-) between repeated/identical characters in consecutive positions using **recursion**.

Sample Input: class

Output: clas-s

Sample Input: aaabbbby

Output: a-a-ab-b-by

Sample Input: mmmm

Output: m-m-m-m

8. Given an integer number greater than 0, find the number of occurrences of a digit in the given number using **recursion**.

Sample Input: number = 1000, digit = 0

Output: 3 [Occurrence of digit 0 in 1000 is 3]

Sample Input: number: 17374, digit = 7

Output: 2 [Occurrence of digit 7 in 17374 is 2]