**Recursion Algorithms**

**Problem 1:** Implementing recursion Algorithms for assignments below:

1. Calculate the Sum of two numbers
2. Calculate the Product of two numbers
3. Calculate the Power of two numbers
4. Finding the GCD (Greatest Common Divisor) of two numbers

**Problem 2:** Implementing recursion Algorithms for assignments below:

1. Reverse an array
2. Apply the recursion Algorithms for Binary search Algorithm
3. Apply the recursion Algorithms for Merge Sort Algorithm
4. Apply the recursion Algorithms for Quick Sort algorithm

**Problem 3:** Given an array of integers arr, write the program to find the minimum and maximum element of that array using recursion.

Example 1:

Input: arr = {1, 4, 3, -5, -4, 8, 6};

Output: min = -5, max = 8

Example 2:

Input: arr = {1, 4, 45, 6, 10, -8};

Output: min = -8, max = 45

**Problem 4:** Given a string, use recursion algorithms to remove adjacent duplicate characters from the string. The output string should not have any adjacent duplicates

Example 1:

Input: azxxzy

Output: ay

Explains:

First “azxxzy” is reduced to “azzy”.

The string “azzy” contains duplicates,

so it is further reduced to “ay”.

Example 2:

Input: geeksforgeeg

Output: gksfor

Example 3:

Input: caaabbbaacdddd

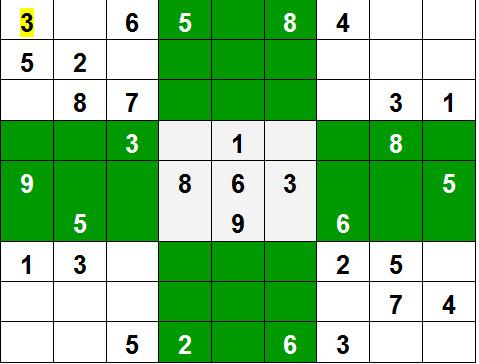
Output: Empty String

Example 4:

Input: acaaabbbacdddd

Output: acac

**Problem 5 (Option):** Given a partially filled 9×9 2D array ‘grid[9][9]’, the goal is to assign digits (from 1 to 9) to the empty cells so that every row, column, and subgrid of size 3×3 contains exactly one instance of the digits from 1 to 9.



Example:

Input: grid

{ {3, 0, 6, 5, 0, 8, 4, 0, 0},  
{5, 2, 0, 0, 0, 0, 0, 0, 0},  
{0, 8, 7, 0, 0, 0, 0, 3, 1},  
{0, 0, 3, 0, 1, 0, 0, 8, 0},  
{9, 0, 0, 8, 6, 3, 0, 0, 5},  
{0, 5, 0, 0, 9, 0, 6, 0, 0},   
{1, 3, 0, 0, 0, 0, 2, 5, 0},  
{0, 0, 0, 0, 0, 0, 0, 7, 4},  
{0, 0, 5, 2, 0, 6, 3, 0, 0} }

Output:

3 1 6 5 7 8 4 9 2  
5 2 9 1 3 4 7 6 8  
4 8 7 6 2 9 5 3 1  
2 6 3 4 1 5 9 8 7  
9 7 4 8 6 3 1 2 5  
8 5 1 7 9 2 6 4 3  
1 3 8 9 4 7 2 5 6  
6 9 2 3 5 1 8 7 4  
7 4 5 2 8 6 3 1 9