Data Structures & Algorithms Question Paper

Q1: Longest Substring Without Repeating Characters

Problem Statement Given a string `s`, find the length of the **longest substring** without repeating characters.

Constraints

- `0 <= s.length <= 5 * 10^4`
- `s` consists of English letters, digits, symbols, and spaces.

Example 1

Input: s = "abcabcbb"
Output: 3
Explanation: The longest substring without repeating characters is "abc", with a length of 3.

Example 2

Input: s = "bbbbb"
Output: 1
Explanation: The longest substring is "b", with a length of 1.

Q2: Delete Middle Element of a Stack

Problem Statement Given a stack of integers, delete the **middle element** without using any additional data structure (such as arrays, lists, or auxiliary stacks).

- If the stack has an **odd** number of elements, the element at the exact middle index is removed.
- If the stack has an **even** number of elements, remove the element that is closer to the **top** of the stack.

Example 1 (Odd Size)

Input: stack = [1, 2, 3, 4, 5] (Top is 5)
Output: [1, 2, 4, 5]
Explanation: The size is 5, the middle index (0-indexed) is 2, holding the value 3.

Example 2 (Even Size)

Input: stack = [10, 20, 30, 40] (Top is 40)
Output: [10, 30, 40]
Explanation: The size is 4. The middle elements are 20 and 30. The element closer to the top (30) is removed.

Q3: Check if Given Parentheses Expression is Balanced or Not

Problem Statement Given a string `expression` containing only the following parentheses characters: '(', ')', '{', '}', '[', and ']', determine if the input string is **valid** (balanced).

An input string is valid if:

1. Open brackets must be closed by the same type of brackets.
2. Open brackets must be closed in the correct order.
3. Every close bracket has a corresponding open bracket of the same type.

Example 1

Input: expression = "{[()()]}"
Output: True
Explanation: All brackets are correctly opened and closed in the proper sequence.

Example 2

Input: expression = "([)]"
Output: False
Explanation: The square bracket is closed before the inner parenthesis, violating the order rule.

Q4: Merge Two Sorted Lists

Problem Statement You are given the heads of two sorted linked lists, `list1` and `list2`. Merge the two lists into a single sorted linked list. The new list should be made by splicing together the nodes of the first two lists.

Return the head of the merged linked list.

Example 1

Input: list1 = 1 -> 3 -> 5, list2 = 2 -> 4 -> 6
Output: 1 -> 2 -> 3 -> 4 -> 5 -> 6

Explanation: Both sorted lists are merged into a single sorted list.

Example 2

Input: list1 = [], list2 = [0]
Output: [0]
Explanation: Merging an empty list with a list containing a single element.

Q5: Merge K Sorted Lists

Problem Statement You are given an array of `k` linked lists, where each linked list is sorted in ascending order. Merge all the linked lists into one single sorted linked list and return the head of the merged list.

Example 1

Input: lists = [1->4->5, 1->3->4, 2->6]
Output: 1 -> 1 -> 2 -> 3 -> 4 -> 4 -> 5 -> 6
Explanation: All k lists are merged into a single sorted list.

Example 2

Input: lists = [[]]
Output: []
Explanation: Merging a list of one empty list results in an empty list.

Q6: The Celebrity Problem

Problem Statement In a party of N people, a **Celebrity** is defined as someone who satisfies two conditions:

1. The Celebrity does **not** know anyone else at the party.
2. Everyone else at the party **knows** the Celebrity.

You are given an $N \times N$ integer matrix `M`, where `M[i][j] = 1` if person `i` knows person `j`, and `M[i][j] = 0` otherwise.

Your task is to find the index of the celebrity. If the celebrity does not exist, return -1.

Note: You can only use a provided helper function `knows(a, b)` which returns true if person `a` knows person `b`, and false otherwise.

Example 1

Input:
N = 3
matrix M = [[0, 1, 0],
[0, 0, 0],
[0, 1, 0]]

Output: 1
Explanation: Person 1 (index 1) is the celebrity:
1. M[1][0] = 0, M[1][2] = 0 (1 knows no one).
2. M[0][1] = 1, M[2][1] = 1 (Everyone else knows 1).

Example 2

Input:
N = 2
matrix M = [[0, 1],
[1, 0]]
Output: -1
Explanation: Person 0 knows 1, and 1 knows 0. No one satisfies the celebrity criteria.