

## COL106 Lab Week 7 Questions

These problems are from all the topics you have covered till now in the course. This will be good practice for the lab test. You may have seen some similar questions in the older lab sheets. Please make sure you complete these questions and discuss your solutions with your Lab TA. *Good luck!*

### Easy

#### Problem 1: Baseball Game

Your task is to keep score in a baseball game, given a record of what has happened so far. You are given a list of strings `operation[]`, where `operations[i]` is the  $i$ -th operation that you must apply to the record and is one of the following:

- An integer  $x$ : Record a new score of  $x$ .
- '+': Record a new score that is the sum of the previous two scores.
- 'D': Record a new score that is the double of the previous score.
- 'C': Invalidate the previous score, removing it from the record.

Your task is to return the sum of all the scores on the record after applying all the operations.

You can submit and check your solution [here](#).

#### Problem 2: Circular Queues

Your task is to implement circular queues using arrays, as discussed in class. As usual, your implementation should support the typical queue operations (like enqueue, dequeue, etc).

You can submit and check your implementation [here](#).

### Medium

#### Problem 3: Palindromic Linked List

Given a singly linked list of single digit numbers (0-9), return whether it is a palindrome or not, using  $O(n)$  time and *only*  $O(1)$  extra space.

You can submit and check your solution [here](#).

Note: Even if your submission passes, you should verify that your implementation uses  $O(1)$  extra space and runs in  $O(n)$  time, with your lab group TA.

#### Problem 4: Delete Node from BST

Given a Binary Search Tree and an input key, your task is to delete the key from the BST. You need to ensure that the tree remains a BST after deletion.

Submit and check your code [here](#).

Hard

#### Problem 5: Visible People in Queue

There are  $n$  people standing in a queue, and they are numbered from 0 to  $n - 1$  in left to right. You are given an array `heights[]` of distinct integers where `heights[i]` represents the height of the  $i$ -th person.

A person can see another person to their right in the queue if everybody in between them is shorter than both of them.

Return an array *answer* of length  $n$  where `answer[i]` is the number of people the  $i$ -th person can see to their right in the queue.

Submit and check your code [here](#).

#### Problem 6: Number of Special Inversions

Given an integer array `nums[]`, return the number of special inversions in the array.

A special inversions is a pair  $(i, j)$  where:

- $0 \leq i < j < \text{nums.length}$  and
- $\text{nums}[i] > 2 * \text{nums}[j]$ .

You can check and submit your solution [here](#).