Assignment 4

Topics:

1. Queues
2. Linked List
3. Trees

(Maximum marks -15)

Q-1) Implement Queue using Stacks (5 marks)

https://leetcode.com/problems/implement-queue-using-stacks/

(5 marks)
(Easy)

Implement a first in first out (FIFO) queue using only two stacks. The implemented queue should support all the functions of a normal queue (push, peek, pop, and empty).

Implement the MyQueue class:

- void push(int x) Pushes element x to the back of the queue.
- int pop() Removes the element from the front of the queue and returns it.
- int peek() Returns the element at the front of the queue.
- boolean empty() Returns true if the queue is empty, false otherwise.

Notes:

- You must use only standard operations of a stack, which means only push to top, peek/pop from top, size, and is empty operations are valid.
- Depending on your language, the stack may not be supported natively. You may simulate a stack using a list or deque (double-ended queue) as long as you use only a stack's standard operations.

Example 1:

Input

["MyQueue", "push", "push", "peek", "pop", "empty"]

[[], [1], [2], [], [], []]

Output

[null, null, null, 1, 1, false]

Explanation

MyQueue myQueue = new MyQueue();

myQueue.push(1); // queue is: [1]

myQueue.push(2); // queue is: [1, 2] (leftmost is front of the queue)

myQueue.peek(); // return 1

myQueue.pop(); // return 1, queue is [2]

myQueue.empty(); // return false

Q-2) Palindrome Linked List (5 marks)

https://leetcode.com/problems/palindrome-linked-list/

(Easy)

5612

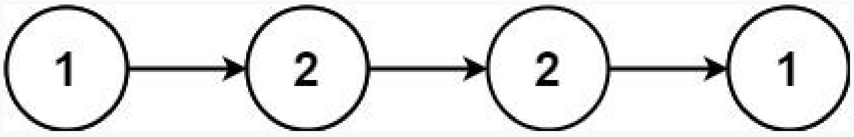
448

Add to List

Share

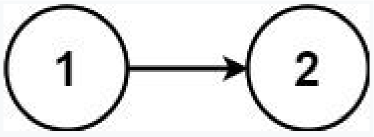
Given the head of a singly linked list, return true if it is a palindrome.

Example 1:



Input: head = [1,2,2,1]
Output: true

Example 2:



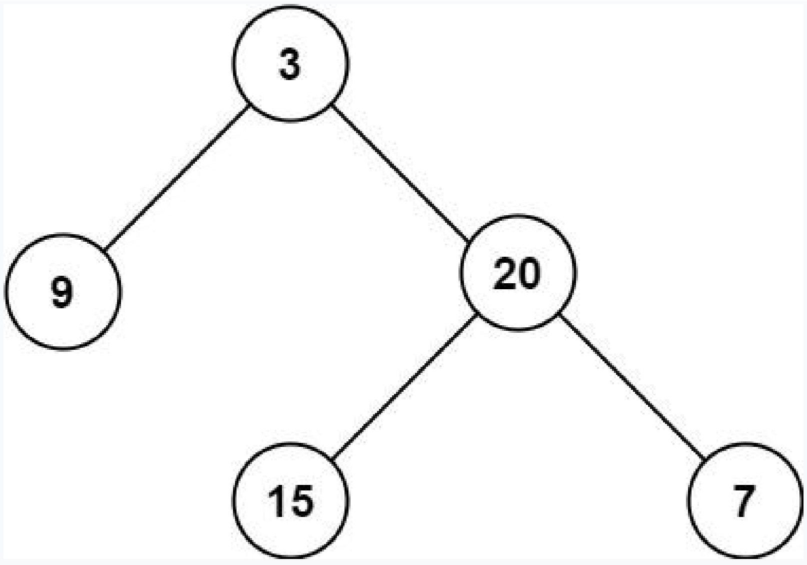
Input: head = [1,2]
Output: false

Q-3) Maximum Depth of Binary Tree(or height of a BT): (5 marks)
https://leetcode.com/problems/maximum-depth-of-binary-tree/

Given the root of a binary tree, return *its maximum depth*.

A binary tree's maximum depth is the number of nodes along the longest path from the root node down to the farthest leaf node.

Example 1:



Input: root = [3,9,20,null,null,15,7]
Output: 3

Example 2:

Input: root = [1,null,2]
Output: 2

Q - 4) [<u>BONUS QUESTION</u>] Implement a stack, using two queues. (4 marks)

Marks distribution:
Questions 1,2 and 3 carry 5 marks each.

Question 4 is a bonus question, that means if you leave that question you don't lose a mark, but if you solve it, you can get an extra 4 marks.

Remark: maximum marks you can get is 15, bonus question helps only if you are not able to solve another question.