**Data Structure Lab**

**ODD 2022**

**Lab Test-1**

**Time: 60 min**

**Marks: 20**

**Instructions**:

1. Submit a pdf file having code and output screenshots
2. FilenameshouldbeRollNo\_Name\_LabTest1\_DS.pdf
3. Output should be at least 3 unique test cases
4. Output Screenshot should NOT be cropped at all

**Set 1 (ODD System No)**

Design and implement stack data structure with following operations:

Stack(): constructs an empty frequency stack.

void push(int val): pushes an integer val onto the top of the stack.

int pop(): removes and returns the most frequent element in the stack.

If there is a tie for the most frequent element, the element closest to the stack's top is removed and returned.

**Example**

Stack fs = new Stack();

fs.push(5); // The stack is [5]

fs.push(7); // The stack is [5,7]

fs.push(5); // The stack is [5,7,5]

fs.push(7); // The stack is [5,7,5,7]

fs.push(4); // The stack is [5,7,5,7,4]

fs.push(5); // The stack is [5,7,5,7,4,5]

fs.pop(); // return 5, as 5 is the most frequent. The stack becomes [5,7,5,7,4].

fs.pop(); // return 7, as 5 and 7 is the most frequent, but 7 is closest to the top. The stack becomes [5,7,5,4].

fs.pop(); // return 5, as 5 is the most frequent. The stack becomes [5,7,4].

fs.pop(); // return 4, as 4, 5 and 7 is the most frequent, but 4 is closest to the top. The stack becomes [5,7].

**Set 2 (Even System No)**

Consider an array of lists having **k** sorted linked lists. Merge all the linked-lists into one sorted linked-list and return it.

**Example 1:**

**Input:** lists = [[1,4,5],[1,3,4],[2,6]]

**Output:** [1,1,2,3,4,4,5,6]

**Explanation:** The linked-lists are:

[

1->4->5,

1->3->4,

2->6

]

Merging them into one sorted list:

1->1->2->3->4->4->5->6

**Example 2:**

**Input**: lists = []

**Output**: []

**Example 3:**

Input: lists = [[]]

Output: []

**Constraints:**

k == lists.length

0 <= k <= 104

0 <= lists[i].length <= 500

-104 <= lists[i][j] <= 104

lists[i] is sorted in ascending order.

The sum of lists[i].length will not exceed 104.