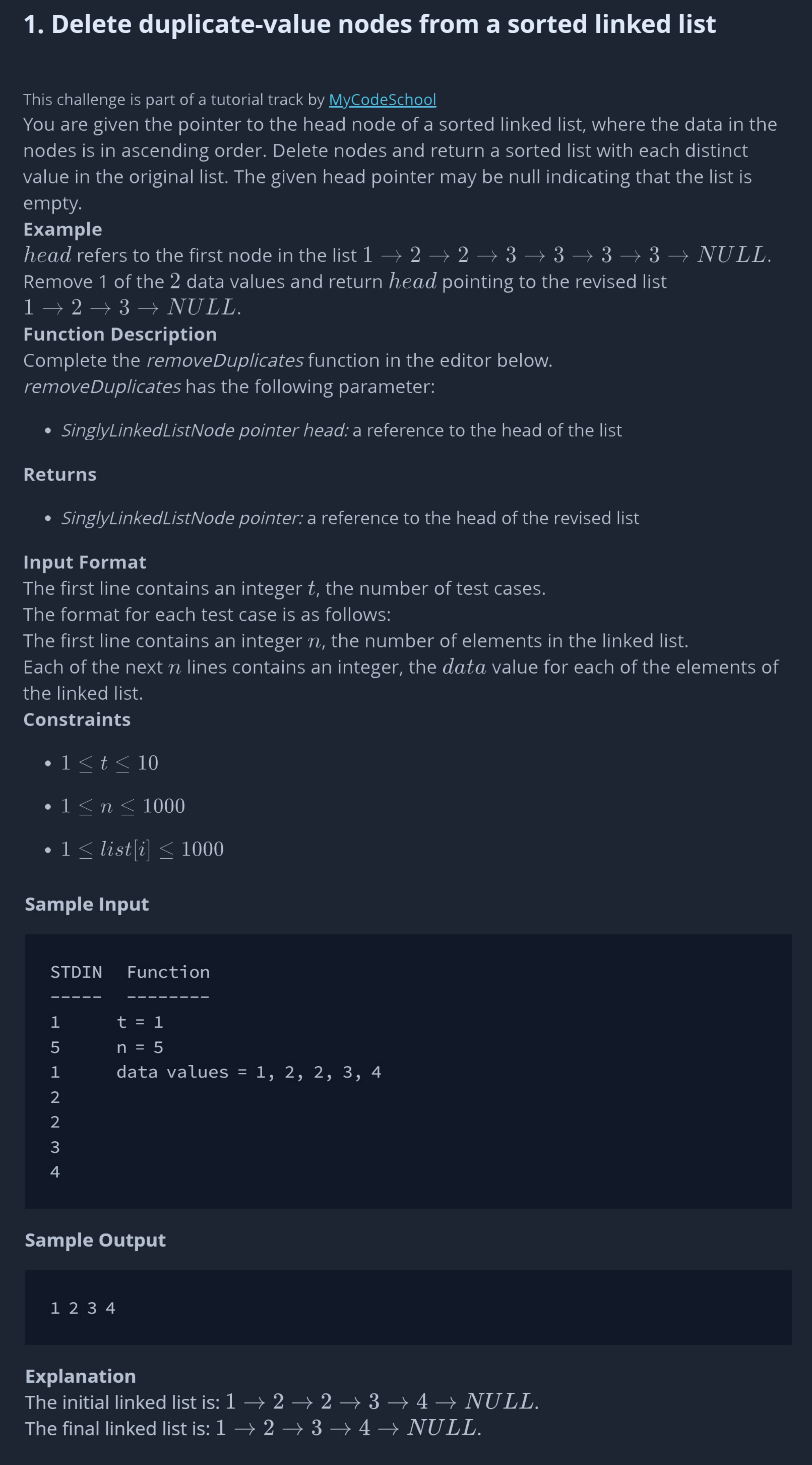

1. Delete duplicate-value nodes from a sorted linked list

This challenge is part of a tutorial track by MyCodeSchool

You are given the pointer to the head node of a sorted linked list, where the data in the nodes is in ascending order. Delete nodes and return a sorted list with each distinct value in the original list. The given head pointer may be null indicating that the list is empty.

Example

$head$ refers to the first node in the list $1 \rightarrow 2 \rightarrow 2 \rightarrow 3 \rightarrow 3 \rightarrow 3 \rightarrow 3 \rightarrow NULL$.

Remove 1 of the 2 data values and return $head$ pointing to the revised list $1 \rightarrow 2 \rightarrow 3 \rightarrow NULL$.

Function Description

Complete the *removeDuplicates* function in the editor below.

removeDuplicates has the following parameter:

- *SinglyLinkedListNode pointer head:* a reference to the head of the list

Returns

- *SinglyLinkedListNode pointer:* a reference to the head of the revised list

Input Format

The first line contains an integer t, the number of test cases.

The format for each test case is as follows:

The first line contains an integer n, the number of elements in the linked list.

Each of the next n lines contains an integer, the $data$ value for each of the elements of the linked list.

Constraints

- $1 \leq t \leq 10$
- $1 \leq n \leq 1000$
- $1 \leq list[i] \leq 1000$

Sample Input

```
STDIN   Function
-----   --------
1      t = 1
5      n = 5
1      data values = 1, 2, 2, 3, 4
2
2
3
4
```

Sample Output

```
1 2 3 4
```

Explanation

The initial linked list is: $1 \rightarrow 2 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow NULL$.

The final linked list is: $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow NULL$.