# Merge two sorted linked lists ★

254 more points to get your next star!

Problem \*\*

X

Rank: 973289 | Points: 221/475

# Your Merge two sorted linked lists submission got 5.00 points.

You are now 254 points away from the 4th star for your problem solving badge.

Try the next challenge

Problem

Submissions

Leaderboard

Editorial 🛆

This challenge is part of a tutorial track by MyCodeSchool

Given pointers to the heads of two sorted linked lists, merge them into a single, sorted linked list. Either head pointer may be null meaning that the corresponding list is empty.

### Example

headA refers to  $1 \rightarrow 3 \rightarrow 7 \rightarrow \textit{NULL}$ 

headB refers to 1 o 2 o NULL

The new list is  $\mathbf{1} \to \mathbf{1} \to \mathbf{2} \to \mathbf{3} \to \mathbf{7} \to NULL$ 

## **Function Description**

Complete the mergeLists function in the editor below.

mergeLists has the following parameters:

- SinglyLinkedListNode pointer headA: a reference to the head of a list
- SinglyLinkedListNode pointer headB: a reference to the head of a list

#### Returns

• SinglyLinkedListNode pointer: a reference to the head of the merged 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  $\boldsymbol{n}$ , the length of the first linked list.

The next  $\boldsymbol{n}$  lines contain an integer each, the elements of the linked list.

The next line contains an integer  $\boldsymbol{m}$ , the length of the second linked list.

The next  $m{m}$  lines contain an integer each, the elements of the second linked list.

#### Constraints

- $1 \le t \le 10$
- $1 \le n, m \le 1000$
- $1 \leq list[i] \leq 1000$ , where list[i] is the  $i^{th}$  element of the list.

## Sample Input

1

3

1

3



2 3 4

# Sample Output

1 2 3 3 4

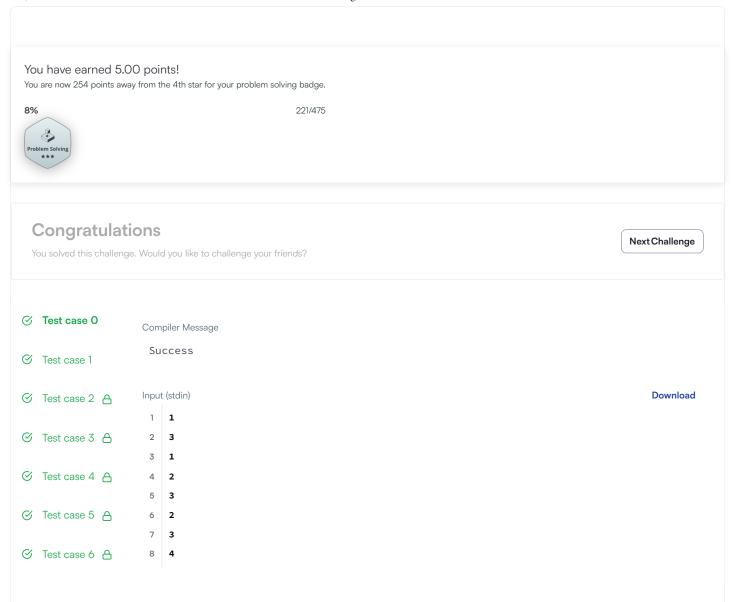
# Explanation

The first linked list is:  $1 \rightarrow 3 \rightarrow 7 \rightarrow NULL$ 

The second linked list is:  $\mathbf{3} \to \mathbf{4} \to NULL$ 

Hence, the merged linked list is:  $1 \rightarrow 2 \rightarrow 3 \rightarrow 3 \rightarrow 4 \rightarrow NULL$ 

```
Change Theme Language Python 3
                                                                                                            1
                                                                                                                E 2 .
     45
                int data
                SinglyLinkedListNode next
     46
     47
     48
     49
          def mergeLists(head1, head2):
     50
              if head1 is None and head2 is None:
                  return None
     51
     52
     53
              head = head1
     54
              cur1 = head1.next
              cur2 = head2
     55
              if head1.data > head2.data:
     56
                  head = head2
     57
     58
                  cur1 = head1
     59
                  cur2 = head2.next
     60
              cur = head
     61
     62
              while cur1 and cur2:
                  if cur1.data < cur2.data:</pre>
     63
     64
                      cur.next = cur1
                      cur = cur1
     65
                      cur1 = cur1.next
     66
     67
     68
                      cur.next = cur2
                      cur = cur2
     69
     70
                      cur2 = cur2.next
              # post precessing
     71
              if cur1 is None:
     72
     73
                  cur.next = cur2
              if cur2 is None:
     74
     75
                  cur.next = cur1
     76
              return head
     77
     78
          if
               name
                      == '
                            main ':--
                                                                                                       Line: 71 Col: 22
EMACS
                                                                                                   Run Code
                                                                                                              Submit Code
 Test against custom input
```



Blog | Scoring | Environment | FAQ | About Us | Helpdesk | Careers | Terms Of Service | Privacy Policy