

---

## Merge Two Sorted Linked Lists

X98439\_en

---

Extend the **PositionalList** class provided in the *Public Files* section of this problem statement with a new public method called **merge**.

```
def merge(self, other):  
    """  
    Pre: self and other are two lists sorted in ascending order.  
    Post: After the merge operation 'self' contains its previous  
          elements and all the elements in 'other' is ascending  
          order. Furthermore, 'other' is empty.  
    Observation: Because 'other' must be empty after carrying out  
                 the merge operation, there is no need to create new nodes.  
    """
```

**Merge** implements the well known **merge** algorithm, which combines the elements of two sorted sequences into a single sorted sequence. In this case, the sequences are represented by two sorted doubly linked lists (i.e. instances of the class **PositionalList** provided in the *Public Files* section of this problem statement). The first sorted sequence is represented by the calling object (i.e. **self**) and the second one by the parameter '*other*'.

For example, if *t1* and *t2* are two instances of the *PositionalList* class, such that

```
t1 = 1 3 5 7 9  
t2 = 2 4 6 8
```

The objects *t1* and *t2* after executing **t1.merge(t2)** should be as follows

```
t1 = 1 2 3 4 5 6 7 8 9  
t2 =
```

Although the application *jutge.org* will accept your solution if you implement the method **merge** calling other public or non-public methods of the class *PositionalList*, we recommend implementing **merge** using only the attributes of the nested class **Node** (i.e., *element*, *prev*, *next*) and the attributes of the class **PositionalList** (i.e., *header*, *trailer*, *size*).

In the file provided in the *Public Files* section of the statement, there is a *main* procedure you may use to test the **merge** method. The **input** to this program consists of two sequences of floating point numbers corresponding to the elements of the two sorted lists *t1* and *t2*. Each sequence starts with an integer number *n* that specifies the number of elements in the sequence, followed by *n* floating point numbers representing the elements in the sequence. The **output** of the program represents the state of both lists, *t1* and *t2*, after executing **t1.merge(t2)**. The elements of both lists are printed separated by one white space.

**Sample input**

```
5 1.0 3.0 5.0 7.0 9.0  
4 2.0 4.0 6.0 8.0
```

**Sample output**

	t1 1.0 2.0 3.0 4.0 5.0 6.0 7.0 8.0 9.0
	t2

## Problem information

Author :  
Generation : 2022-12-12 11:38:51

© *Jutge.org*, 2006–2022.  
<https://jutge.org>