⓿ Merge two Sorted Linked List

Input

```
10 - 20 - 30 - 40 - 50
7 - 9 - 12 - 35 - 45 - 60
```

Output

```
7-9-10-12-20-30-35→40-45-50
                              |
                             60
```

⟹ Hum two pointer approach use krengy! pointers < one / two

⟹ Hum dono pointers ki values ko compare krengy, jiski value smaller hui, usko hum new resultant list (res) me daal dengy! Aur jis pointer ki value dali hai uss pointer +1 increase krdengy!

⟹ Yeh hum tabtak krengy jab tak list of smaller size reaches ZERO. Tab larger list ke bachey huay elements (res) me add hojayengy!

```java
public static LinkedList mergeTwoSortedLinkedList
(LinkedList l1, LinkedList l2) {
Node one = l1.head;
Node two = l2.head;
LinkedList res = new LinkedList ();
while (one != null && two != null) {
   if (one.data < two.data) {
     res.addLast (one.data);
     one = one.next;
  } else {
    res.addLast (two.data);
    two = two.next;
   }
 }
 while (one != null) {
   res.addLast (one.data);
   one = one.next;
 }
 while (two != null) {
   res.addLast (two.data);
   two = two.next;
 }
 return res;
}
```

Output section (right column):

```
one →
10 - 20 - 30 - 40 - 50
two →
7 - 9 - 12 - 35 - 45 - 60
res =
```

```
one ↓
10 - 20 - 30 - 40 - 50
        two ↓
7 - 9 - 12 - 35 - 45 - 60
res = 7
```

```
one → 10 - 20 - 30 - 40 - 50
            two ↓
7 - 9 - 12 - 35 - 45 - 60
res = 7-9
```

```
one ↓
10 - 20 - 30 - 40 - 50
     two ↓
7 - 9 - 12 - 35 - 45 - 60
res = 7-9-10
```

∴ AT LAST

```
                              one ↓
10 - 20 - 30 - 40 - 50
                              two ↓
7 - 9 - 12 - 35 - 45 - 60
```

```
res = 7-9-10-12-20-30-35-40
                              |
            60 - 50 - 45
```