

Our Solution(s)

Run Code

Your Solutions

Run Code

Solution 1

Solution 2

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3 class Program {
4     class LinkedList {
5         var value: Int
6         var next: LinkedList?
7
8         init(value: Int) {
9             self.value = value
10        }
11    }
12
13    // O(n + m) time | O(1) space - where n is the number of nodes in
14    // Linked List and m is the number of nodes in the second Linked L
15    func mergeLinkedLists(_ headOne: LinkedList, _ headTwo: LinkedList
16        var p1 = headOne as LinkedList?
17        var p2 = headTwo as LinkedList?
18        var p1Prev: LinkedList?
19
20        while p1 != nil, p2 != nil {
21            if p1!.value < p2!.value {
22                p1Prev = p1
23                p1 = p1!.next
24            } else {
25                if p1Prev != nil {
26                    p1Prev!.next = p2
27                }
28                p1Prev = p2
29                p2 = p2!.next
30                p1Prev!.next = p1
31            }
32        }
33    }
```

Solution 1

Solution 2

Solution 3

```
1 class Program {
2     // This is an input class. Do not edit.
3     class LinkedList {
4         var value: Int
5         var next: LinkedList?
6
7         init(value: Int) {
8             self.value = value
9         }
10    }
11
12    func mergeLinkedLists(_ headOne: LinkedList, _ headTwo: LinkedList
13        // Write your code here.
14        return headOne // replace me
15    }
16 }
17
```

Our Tests

Custom Output

Submit Code

```
1 class Program {
2     // This is an input class. Do not edit.
3     class LinkedList {
4         var value: Int
5         var next: LinkedList?
6
7         init(value: Int) {
8             self.value = value
9         }
10    }
11
12    func mergeLinkedLists(_ headOne: LinkedList, _ headTwo: LinkedList
13        // Write your code here.
14        return headOne // replace me
15    }
16 }
```

Run or submit code when you're ready.

Run or submit code when you're ready.