```swift
// Copyright © 2020 AlgoExpert, LLC. All rights reserved.

class Program {
    class LinkedList {
        var value: Int
        var next: LinkedList?

        init(value: Int) {
            self.value = value
        }
    }

    // O(n + m) time | O(1) space - where n is the number of nodes in the first
    // Linked List and m is the number of nodes in the second Linked List
    func mergeLinkedLists(_ headOne: LinkedList, _ headTwo: LinkedList) -> LinkedList {
        var p1 = headOne as LinkedList?
        var p2 = headTwo as LinkedList?
        var p1Prev: LinkedList?

        while p1 != nil, p2 != nil {
            if p1!.value < p2!.value {
                p1Prev = p1
                p1 = p1!.next
            } else {
                if p1Prev != nil {
                    p1Prev!.next = p2
                }
                p1Prev = p2
                p2 = p2!.next
                p1Prev!.next = p1
            }
        }

        if p1 == nil, p1Prev != nil {
            p1Prev!.next = p2
        }

        if headOne.value < headTwo.value {
            return headOne
        }
        return headTwo
    }
}

```