

Our Solution(s)

Run Code

Your Solutions

Run Code

Solution 1

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3 // O(n) time | O(1) space - where n is the number of nodes in the Link
4 function reverseLinkedList(head) {
5   let p1 = null,
6       p2 = head;
7   while (p2 !== null) {
8     const p3 = p2.next;
9     p2.next = p1;
10    p1 = p2;
11    p2 = p3;
12  }
13  return p1;
14 }
15
16 exports.reverseLinkedList = reverseLinkedList;
17
```

Solution 1Solution 2Solution 3

```
1 function reverseLinkedList(head) {
2   // Write your code here.
3 }
4
5 // Do not edit the line below.
6 exports.reverseLinkedList = reverseLinkedList;
7
```

Our Tests

Custom Output

Submit Code

```

18
19
20 # Merge sort
21 def merge_sort(arr):
22     if len(arr) < 2:
23         return arr
24     mid = len(arr) // 2
25     left = merge_sort(arr[:mid])
26     right = merge_sort(arr[mid:])
27     return merge(left, right)
28
29 def merge(left, right):
30     result = []
31     i = j = 0
32     while i < len(left) and j < len(right):
33         if left[i] <= right[j]:
34             result.append(left[i])
35             i += 1
36         else:
37             result.append(right[j])
38             j += 1
39     result.extend(left[i:])
40     result.extend(right[j:])
41     return result
42
43 # Example usage
44 arr = [5, 2, 8, 1, 9, 3, 7, 4, 6]
45 sorted_arr = merge_sort(arr)
46 print(sorted_arr)

```

Run or submit code when you're ready.