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 Linked List
4  function shiftLinkedList(head, k) {
5    let listLength = 1;
6    let listTail = head;
7    while (listTail.next !== null) {
8      listTail = listTail.next;
9      listLength++;
10   }
11
12   const offset = Math.abs(k) % listLength;
13   if (offset === 0) return head;
14
15   const newTailPosition = k > 0 ? listLength - offset : offset;
16   let newTail = head;
17   for (let i = 1; i < newTailPosition; i++) {
18     newTail = newTail.next;
19   }
20
21   const newHead = newTail.next;
22   newTail.next = null;
23   listTail.next = head;
24   return newHead;
25 }
26
27 // This is the class of the input linked list.
28 class LinkedList {
29   constructor(value) {
30     this.value = value;
31     this.next = null;
32   }
33 }
34
35 exports.shiftLinkedList = shiftLinkedList;
36
```