Solution 1

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

package main

type LinkedList struct {
  Value int
  Next  *LinkedList
}

// O(n) time | O(1) space - where n is the number of nodes in the Linked List
func ShiftLinkedList(head *LinkedList, k int) *LinkedList {
  listLength := 1
  listTail := head
  for listTail.Next != nil {
    listTail = listTail.Next
    listLength += 1
  }

  offset := abs(k) % listLength
  if offset == 0 {
    return head
  }

  newTailPosition := listLength - offset
  if k <= 0 {
    newTailPosition = offset
  }

  newTail := head
  for i := 1; i < newTailPosition; i++ {
    newTail = newTail.Next
  }

  newHead := newTail.Next
  newTail.Next = nil
  listTail.Next = head
  return newHead
}

func abs(k int) int {
  if k > 0 {
    return k
  }
  return -k
}

```