Reverse a Linked List Java

Reverse a Linked List without using extra space.

```
Old List
1 -> 2 -> 3 -> 4 -> null
New List
4 -> 3 -> 2 -> 1 -> null
```

Iterative Method

Time complexity - O(n)

Space complexity - O(1)

```
public void reverseList() {
    if(head == null || head.next == null) {
        return;
    }

    Node prevNode = head;
    Node currNode = head.next;
    while(currNode != null) {
        Node nextNode = currNode.next;
        currNode.next = prevNode;
        prevNode = currNode;
        currNode = nextNode;
    }
    head.next = null;
    head = prevNode;
}
```

Recursive Method

Time complexity - O(n)

Space complexity - O(1)

```
public Node reverseListRecursive(Node head) {
    //empty node || last node or only one node
    if(head == null || head.next == null) {
        return head;
    }

    Node newHead = reverseListRecursive(head.next);

    head.next.next = head;
    head.next = null;
    return newHead;
}
```

Collections Method

Time complexity - O(n)

Space complexity - O(1)

```
LinkedList<Integer> list2 = new LinkedList<>();
list2.add(1);
list2.add(2);
Collections.reverse(list2);
```

Homework Problems

- 1. https://leetcode.com/problems/swap-nodes-in-pairs/
- 2. https://leetcode.com/problems/remove-nth-node-from-end-of-list/
- 3. https://leetcode.com/problems/reverse-linked-list-ii/
- 4. https://leetcode.com/problems/remove-nth-node-from-end-of-list/