Assignment3

March 24, 2025

0.0.1 Dhruv Choudhary RollNo.09

[2]:
```python
class ListNode:
    def __init__(self, val=0, next=None):
        self.val = val
        self.next = next

def rotateRight(head, k):
    if not head or not head.next or k == 0:
        return head

    # Step 1: Find the length of the linked list
    length = 1
    tail = head
    while tail.next:
        tail = tail.next
        length += 1

    # Step 2: Adjust k if it's greater than length
    k = k % length
    if k == 0:
        return head

    # Step 3: Find the new tail (length - k - 1) and new head
    new_tail = head
    for _ in range(length - k - 1):
        new_tail = new_tail.next

    new_head = new_tail.next
    new_tail.next = None  # Break the link
    tail.next = head  # Connect tail to head

    return new_head

# Helper function to convert linked list to list
def linkedListToList(head):
    values = []
    while head:
```

```
        values.append(head.val)
        head = head.next
    return values

# Function to create linked list from list
def createLinkedList(arr):
    if not arr:
        return None
    head = ListNode(arr[0])
    current = head
    for val in arr[1:]:
        current.next = ListNode(val)
        current = current.next
    return head

# Example Usage:
input_list = [1, 2, 3, 4, 5]
k = 2
head = createLinkedList(input_list)

print("Input List:", linkedListToList(head))
print("k:", k)

new_head = rotateRight(head, k)

print("Output List:", linkedListToList(new_head))
```

```
Input List: [1, 2, 3, 4, 5]
k: 2
Output List: [4, 5, 1, 2, 3]
```