Assignment_03

March 23, 2025

Name - Shravani Sandeep Raut

SE/ 48

1. Given the head of a linked list, rotate the list to the right by k places.

[7]:
```
class Node:
    def __init__(self, value):
        self.data = value
        self.next = None

class LinkedList:
    def __init__(self):
        self.head = None

    def insert(self, data):
        newNode = Node(data)
        if not self.head:
            self.head = newNode
        else:
            temp = self.head
            while temp.next:
                temp = temp.next
            temp.next = newNode

    def printList(self):
        temp = self.head
        while temp:
            print(temp.data, end=" -> ")
            temp = temp.next
        print("NULL")

    def rotate(self, k):
        if not self.head or k == 0:
            return

        # Find length and last node
        temp, length = self.head, 1
        while temp.next:
```

```
            temp = temp.next
            length += 1

        # Adjust k
        k = k % length
        if k == 0:
            return

        # Find new tail (length - k - 1)
        temp = self.head
        for _ in range(length - k - 1):
            temp = temp.next

        # Rotate the list
        new_head = temp.next
        temp.next = None
        temp = new_head
        while temp.next:
            temp = temp.next
        temp.next = self.head
        self.head = new_head

# Taking user input
ll = LinkedList()
values = list(map(int, input("Enter linked list elements: ").split()))
for val in values:
    ll.insert(val)

k = int(input("Enter k: "))

print("\nOriginal List:")
ll.printList()

ll.rotate(k)

print("\nRotated List:")
ll.printList()
```

```
Original List:
1 -> 2 -> 3 -> 4 -> 5 -> NULL

Rotated List:
4 -> 5 -> 1 -> 2 -> 3 -> NULL
```