**[Rotate Doubly Linked list by N nodes.](https://www.geeksforgeeks.org/rotate-doubly-linked-list-n-nodes/)**

class Node {

    constructor(val) {

        this.val = val;

        this.next = null;

        this.prev = null;

    }

}

class DoublyLinkedList{

    constructor(val) {

        this.head = null;

        this.tail = null;

    }

    addLast(val) {

        const newNode = new Node(val);

        if(!this.head) {

            this.head = newNode;

            this.tail = newNode;

            return

        }

        newNode.prev = this.tail;

        this.tail.next = newNode;

        this.tail = newNode;

    }

    size() {

        let current = this.head;

        let length = 0;

        while (current) {

            length++;

            current = current.next;

        }

        return length;

    }

    print() {

        let outputString = '';

        let current = this.head;

        while (current) {

            outputString += current.val + '<->';

            current = current.next;

        }

        console.log(outputString + 'null');

    }

    rotateByK(k){

        let current = this.head;

        let prev = null;

        let cutOffPoint = this.size() - k;

        let movement = 0;

        while (movement < cutOffPoint) {

            movement++;

            prev = current;

            current = current.next;

        }

        let newHead = current;

        let newCurrent = current;

        prev.next = null;

        while (newCurrent.next) {

            newCurrent = newCurrent.next;

        }

        newCurrent.next = this.head;

        this.head = newHead;

    }

}

let dll = new DoublyLinkedList();

dll.addLast(1);

dll.addLast(2);

dll.addLast(3);

dll.addLast(4);

dll.addLast(5);

console.log(dll.size())

dll.rotateByK(2);

dll.print();