**Adding Two Numbers Represented by Linked Lists By Using Carry**

class Node {

    constructor(data) {

        this.data = data;

        this.next = null;

    }

}

class LinkedList {

    constructor() {

        this.head = null;

        this.tail = null;

    }

    addFront(data) {

        const newNode = new Node(data);

        newNode.next = this.head;

        this.head = newNode;

        if (!this.tail) {

            this.tail = newNode;

        }

    }

    reverse() {

        let prev = null;

        let current = this.head;

        while (current !== null) {

            let nextTemp = current.next;

            current.next = prev;

            prev = current;

            current = nextTemp;

        }

        this.head = prev;

    }

    addTwoLists(firstList, secondList) {

        firstList.reverse();

        secondList.reverse();

        let result = new LinkedList();

        let carry = 0;

        let firstCurrent = firstList.head;

        let secondCurrent = secondList.head;

        while (firstCurrent !== null || secondCurrent !== null || carry > 0) {

            let sum = carry + (firstCurrent ? firstCurrent.data : 0) + (secondCurrent ? secondCurrent.data : 0);

            carry = Math.floor(sum / 10);

            result.addFront(sum % 10);

            if (firstCurrent) firstCurrent = firstCurrent.next;

            if (secondCurrent) secondCurrent = secondCurrent.next;

        }

        return result;

    }

    print() {

        let current = this.head;

        while (current !== null) {

            console.log(current.data);

            current = current.next;

        }

    }

}

const linkedlist1 = new LinkedList();

linkedlist1.addFront(6);

linkedlist1.addFront(4);

linkedlist1.addFront(3);

const linkedlist2 = new LinkedList();

linkedlist2.addFront(5);

linkedlist2.addFront(4);

linkedlist2.addFront(3);

const sumList = linkedlist1.addTwoLists(linkedlist1, linkedlist2);

console.log("<=======>");

sumList.print();