**Problem Statement:**

You are given two singly linked lists representing two non-negative integers. The digits are stored in reverse order, and each node contains a single digit. Write a Java program to add the two numbers and return the sum as a linked list (also in reverse order).

**SOLUTION:**

class ListNode {

int value; // Changed from 'val' to 'value' for clarity

ListNode next;

ListNode(int value) {

this.value = value;

}

}

public class LinkedListAddition {

public ListNode addTwoNumbers(ListNode firstNumber, ListNode secondNumber) {

// We'll use a dummy starter node to make building the result easier

ListNode resultStarter = new ListNode(0);

ListNode currentDigit = resultStarter;

int carryOver = 0;

// Walk through both numbers digit by digit

while (firstNumber != null || secondNumber != null) {

// Get the current digits (use 0 if we've reached the end of a number)

int firstDigit = (firstNumber != null) ? firstNumber.value : 0;

int secondDigit = (secondNumber != null) ? secondNumber.value : 0;

// Add the digits along with any carry from the previous addition

int total = carryOver + firstDigit + secondDigit;

carryOver = total / 10; // The new carry is the tens digit

int digitToStore = total % 10; // We store the ones digit

// Create a new node for this digit and move forward

currentDigit.next = new ListNode(digitToStore);

currentDigit = currentDigit.next;

// Move to the next digits in both numbers if they exist

if (firstNumber != null) firstNumber = firstNumber.next;

if (secondNumber != null) secondNumber = secondNumber.next;

}

// If there's any carry left after all digits are processed

if (carryOver > 0) {

currentDigit.next = new ListNode(carryOver);

}

// The actual result starts after our dummy node

return resultStarter.next;

}

// Helper to display our number (for testing)

public static void printNumber(ListNode number) {

while (number != null) {

System.out.print(number.value + " ");

number = number.next;

}

System.out.println();

}

public static void main(String[] args) {

// Let's test with the example: 342 + 465 = 807 (stored as 2->4->3 and 5->6->4)

ListNode firstNumber = new ListNode(2);

firstNumber.next = new ListNode(4);

firstNumber.next.next = new ListNode(3);

ListNode secondNumber = new ListNode(5);

secondNumber.next = new ListNode(6);

secondNumber.next.next = new ListNode(4);

LinkedListAddition calculator = new LinkedListAddition();

ListNode sum = calculator.addTwoNumbers(firstNumber, secondNumber);

printNumber(sum); // Should show: 7 0 8 (which is 807 in reverse)

}

}