Problem 369: Plus One Linked List

Problem Information

Difficulty: Medium
Acceptance Rate: 61.22%
Paid Only: Yes
Tags: Linked List, Math

Problem Description

Given a non-negative integer represented as a linked list of digits, _plus one to the integer_.

The digits are stored such that the most significant digit is at the `head` of the list.

Example 1:

Input: head = [1,2,3] **Output:** [1,2,4]

Example 2:

Input: head = [0] **Output:** [1]

Constraints:

* The number of nodes in the linked list is in the range `[1, 100]`. * `0 <= Node.val <= 9` * The number represented by the linked list does not contain leading zeros except for the zero itself.

Code Snippets

C++:

```
/**
* Definition for singly-linked list.
* struct ListNode {
* int val;
```

```
* ListNode *next;
* ListNode() : val(0), next(nullptr) {}
* ListNode(int x) : val(x), next(nullptr) {}
* ListNode(int x, ListNode *next) : val(x), next(next) {}
* };
*/
class Solution {
public:
ListNode* plusOne(ListNode* head) {

}
};
```

Java:

```
/**
* Definition for singly-linked list.
* public class ListNode {
* int val;
* ListNode next;
* ListNode() {}
* ListNode(int val) { this.val = val; }
* ListNode(int val, ListNode next) { this.val = val; this.next = next; }
* }
*/
class Solution {
public ListNode plusOne(ListNode head) {

}
}
```

Python3:

```
# Definition for singly-linked list.
# class ListNode:
# def __init__(self, val=0, next=None):
# self.val = val
# self.next = next
class Solution:
def plusOne(self, head: ListNode) -> ListNode:
```