369. Plus One Linked List

Dec. 21, 2019 | 5.6K views

Average Rating: 4.77 (22 votes)

Given a non-negative integer represented as **non-empty** a singly linked list of digits, plus one to the integer.

You may assume the integer do not contain any leading zero, except the number 0 itself.

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

Example :

```
Input: [1,2,3]
Output: [1,2,4]
```

Solution

Overview.

"Plus One" is a subset of a problem set "Add Number", and the solution patterns are the same.

All these problems could be solved in linear time, and the question here is how to solve without using addition operation or how to fit into constant space complexity.

The choice of algorithm should be based on input format:

1. Integers. Usually addition operation is not allowed for such a case. Use Bit Manipulation Approach. Here is an example: Add Binary.
2. Strings. Use schoolbook bit by bit computation. Note, that to fit into constant space is not possible for languages with immutable strings, for ex. for Java and Python. Here is an example: Add Binary.
3. Arrays. The same textbook addition. Here is an example: Add to Array Form of Integer.
4. Linked Lists, current problem. Sentinel Head + Textbook Addition.

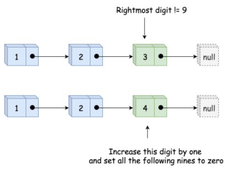
Note, that straightforward idea to convert everything into integers and then use addition could be risky for Java interviews because of possible overflow issues, here is in more details.

Approach 1: Sentinel Head + Textbook Addition.

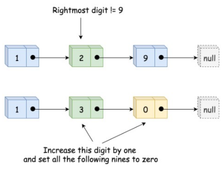
Textbook Addition

Let's identify the rightmost digit which is not equal to nine and increase that digit by one. All the following nines should be set to zero.

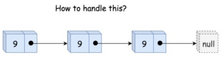
Here is the simplest use case which works fine.



Here is more difficult case which still passes.



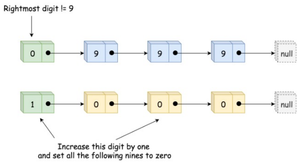
And here is the case which breaks everything.



Sentinel Head

To handle the last use case, one needs so called Sentinel Node. Sentinel nodes are widely used for trees and linked lists as pseudo-heads, pseudo-tails, etc. They are purely functional, and usually don't hold any data. Their main purpose is to standardize the situation to avoid edge case handling.

For example, here one could add pseudo-head with zero value, and hence there will always be not-nine node.



Algorithm

- Initialize sentinel node as `ListNode(0)` and set it to be the new head: `sentinel.next = head`.
- Find the rightmost digit not equal to nine.
- Increase that digit by one.
- Set all the following nines to zero.
- Return sentinel node if it was set to 1, and head `sentinel.next` otherwise.

Implementation

Java | Python — Copy

```
class Solution:
    def plusOne(self, head: ListNode) -> ListNode:
        # sentinel head
        sentinel = ListNode(0)
        sentinel.next = head
        not_nine = sentinel

        # find the rightmost not-nine digit
        while head:
            if head.val != 9:
                not_nine = head
            head = head.next

        # increase this rightmost not-nine digit by 1
        not_nine.val += 1
        not_nine = not_nine.next

        # set all the following nines to zeros
        while not_nine:
            not_nine.val = 0
            not_nine = not_nine.next

        return sentinel if sentinel.val else sentinel.next
```

Complexity Analysis

- Time complexity : $\mathcal{O}(N)$ since it's not more that two passes along the input list.
- Space complexity : $\mathcal{O}(1)$.

Analysis written by @liaison and @andvary

Rate this article:

Previous Next

Comments: 11

Sort By

Type comment here... (Markdown is supported)

Preview Post

denis19 ★4 January 3, 2020 8:43 AM

Nice article. Pretty clean solution. Not sure where ListNode is coming from. I guess it's just an assumed class of a standard LinkedList?

3 Share Reply

SHOW 1 REPLY

xavier ★38 March 25, 2020 9:15 AM

Easy Python Solution:

```
class Solution:
    def plusOne(self, head):
```

Read More

1 Share Reply

hfsvania ★6 February 2, 2020 9:53 PM

awesome

1 Share Reply

dingli ★14 February 1, 2020 6:14 PM

awesome

1 Share Reply

mansi1504 ★1 January 10, 2020 3:20 PM

Awesome

1 Share Reply

siddpeter ★14 January 6, 2020 3:09 AM

Brailliant!

1 Share Reply

flak153 ★2 December 22, 2019 8:02 AM

Please check grammar

1 Share Reply

SHOW 1 REPLY

venu_sri_sushma ★0 May 3, 2020 10:36 AM

how does it work for [1,2,9,4]

0 Share Reply

SHOW 1 REPLY

junsayulia ★1 May 2, 2020 1:53 AM

Hi can anybody please explain why he's checking against 9? Thank you

0 Share Reply

SHOW 2 REPLIES

tinaba ★0 April 30, 2020 10:18 AM

"not_nine" is not a good name for the latter half of the code (from line 16).
It should be something like "nine" so that the code will be easier for us to read.

0 Share Reply

1 2