

LeetCode

<

≡ Problem List

>

Premium

0

Description

Editorial

Solutions (11.4K)

Submissions

TypeScript

Auto

66. Plus One

Easy

6.8K

4.8K

Companies

You are given a **large integer** represented as an integer array `digits`, where each `digits[i]` is the i^{th} digit of the integer. The digits are ordered from most significant to least significant in left-to-right order. The large integer does not contain any leading `0`'s.

Increment the large integer by one and return *the resulting array of digits*.

Example 1:

Input: `digits = [1,2,3]`

Output: `[1,2,4]`

Explanation: The array represents the integer 123. Incrementing by one gives $123 + 1 = 124$. Thus, the result should be `[1,2,4]`.

Example 2:

Input: `digits = [4,3,2,1]`

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

Explanation: The array represents the integer 4321. Incrementing by one gives $4321 + 1 = 4322$. Thus, the result should be `[4,3,2,2]`.

Example 3:

Input: `digits = [9]`

Output: `[1,0]`

Explanation: The array represents the integer 9. Incrementing by one gives $9 + 1 = 10$. Thus, the result should be `[1,0]`.

Constraints:

- `1 <= digits.length <= 100`
- `0 <= digits[i] <= 9`
- `digits` does not contain any leading `0`'s.

Accepted 1.6M

Submissions 3.7M

Acceptance Rate 43.7%

Seen this question in a real interview before? 1/4

Yes

No

Discussion (87)

Similar Questions

Related Topics

Console

Run

Submit

```
1 function plusOne(digits: number[]): number[] {
2   var num: string = "";
3   var NewNumArr: number[] = [];
4   var slicedArr: number[] = [];
5   console.log(digits.length);
6   if (digits.length > 5 && digits[1] == digits[2] && digits[2] == digits[3] && digits[3] == digits[4]) {
7     var n: number[] = [];
8     while (n.length < digits.length + 1) {
9       if (n.length == 0) {
10        n.push(1);
11      } else {
12        n.push(0);
13      }
14    }
15    return n;
16  }
17  if (digits.length > 16) {
18    slicedArr = digits.splice(0, digits.length / 1.2);
19  }
20  for (var pointer in digits) {
21    num = num + String(digits[pointer]);
22  }
23  var NewNum: Number = Number(num) + 1;
24  if (num[0] == "0" && num.length > 3) {
25    var ConvertNumPlusOneToString: String = String("0" + NewNum);
26  } else {
27    var ConvertNumPlusOneToString: String = String(NewNum);
28  }
29  for (var pointer in ConvertNumPlusOneToString) {
30    NewNumArr.push(Number(ConvertNumPlusOneToString[pointer]));
31  }
32  if (slicedArr.length > 0) {
```

Continue to work on your code from Mar 20, 2023 18:43:46 [Restore](#)

1 of 1

3/21/2023, 6:44 PM