

0066. 加一

👤 [ITCharge](#) ⌚ 大约 1 分钟

- 标签：数组、数学
- 难度：简单

题目链接

- [0066. 加一 - 力扣](#)

题目大意

描述： 给定一个非负整数数组，数组每一位对应整数的一位数字。

要求： 计算整数加 1 后的结果。

说明：

- $1 \leq \text{digits.length} \leq 100$ 。
- $0 \leq \text{digits}[i] \leq 9$ 。

示例：

- 示例 1:

输入: `digits = [1,2,3]`

输出: `[1,2,4]`

解释: 输入数组表示数字 123，加 1 之后为 124。

py

- 示例 2:

输入: `digits = [4,3,2,1]`

输出: `[4,3,2,2]`

解释: 输入数组表示数字 4321。

py

解题思路

思路 1：模拟

这道题把整个数组看成了一个整数，然后个位数加 1。问题的实质是利用数组模拟加法运算。

如果个位数不为 9 的话，直接把个位数加 1 就好。如果个位数为 9 的话，还要考虑进位。

具体步骤：

1. 数组前补 0 位。
2. 将个位数字进行加 1 计算。
3. 遍历数组
 1. 如果该位数字大于等于 10，则向下一位进 1，继续下一位判断进位。
 2. 如果该位数字小于 10，则跳出循环。

思路 1：代码

```
def plusOne(self, digits: List[int]) -> List[int]:
    digits = [0] + digits
    digits[len(digits) - 1] += 1
    for i in range(len(digits)-1, 0, -1):
        if digits[i] != 10:
            break
        else:
            digits[i] = 0
            digits[i - 1] += 1

    if digits[0] == 0:
        return digits[1:]
    else:
        return digits
```

py

思路 1：复杂度分析

- 时间复杂度： $O(n)$ 。一重循环遍历的时间复杂度为 $O(n)$ 。
- 空间复杂度： $O(1)$ 。