258. Add Digits

June 2, 2020 | 6.8K views

Previous Next

Average Rating: 4.68 (53 votes)

Given a non-negative integer `num`, repeatedly add all its digits until the result has only one digit.

Example:

```
Input: 38
Output: 2
Explanation: The process is like: 3 + 8 = 11, 1 + 1 = 2.
             Since 2 has only one digit, return it.
```

Follow up:
Could you do it without any loop/recursion in O(1) runtime?

Overview

The value we're asked to compute is the so-called Digital Root. Logarithmic time solution is easy to write, although the main question here is how to fit into a constant time.

Java | Python3 — Copy

```
class Solution:
    def addDigits(self, num: int) -> int:
        digital_root = 0
        while num > 0:
            digital_root += num % 10
            num = num // 10

            if num == 0 and digital_root > 9:
                num = digital_root
                digital_root = 0

        return digital_root
```

Approach 1: Mathematical: Digital Root

Formula for the Digital Root

There is a known formula to compute a digital root in a decimal numeral system

$$dr_{10}(n) = 0, \quad \text{if } n = 0$$
$$dr_{10}(n) = 9, \quad \text{if } n = 9k$$
$$dr_{10}(n) = n \mod 9, \quad \text{if } n \neq 9k$$

How to derive it? Probably, you already know the following proof from school, where it was used for a divisibility by 9: "The original number is divisible by 9 if and only if the sum of its digits is divisible by 9". Let's revise it briefly.

The input number could be presented in a standard way, where $d_0, d_1, ..d_k$ are digits of n:

$$n = d_0 + d_1 \cdot 10^1 + d_2 \cdot 10^2 + ... + d_k \cdot 10^k$$

One could expand each power of ten, using the following:

$$10 = 9 \cdot 1 + 1$$
$$100 = 99 + 1 = 9 \cdot 11 + 1$$
$$1000 = 999 + 1 = 9 \cdot 111 + 1$$
$$...$$
$$10^k = \underbrace{100..0}_{\text{k times}} = \underbrace{99..9}_{\text{k times}} + 1 = 9 \cdot \underbrace{11..1}_{\text{k times}} + 1$$

That results in

$$n = d_0 + d_1 \cdot (9 \cdot 1 + 1) + d_2 \cdot (9 \cdot 11 + 1) + ... + d_k \cdot (9 \cdot \underbrace{11..1}_{\text{k times}} + 1)$$

and could be simplified as

$$n = (d_0 + d_1 + d_2 + ... + d_k) + 9 \cdot (d_1 \cdot 1 + d_2 \cdot 11 + ... + d_k \cdot \underbrace{11..1}_{\text{k times}})$$

The last step is to take the modulo from both sides:

$$n \mod 9 = (d_0 + d_1 + d_2 + ... + d_k) \mod 9$$

and to consider separately three cases: the sum of digits is 0, the sum of digits is divisible by 9, and the sum of digits is *not* divisible by nine:

$$dr_{10}(n) = 0, \quad \text{if } n = 0$$
$$dr_{10}(n) = 9, \quad \text{if } n = 9k$$
$$dr_{10}(n) = n \mod 9, \quad \text{if } n \neq 9k$$

Implementation

The straightforward implementation is

Java | Python3 — Copy

```
class Solution:
    def addDigits(self, num: int) -> int:
        if num == 0:
            return 0
        if num % 9 == 0:
            return 9
        return num % 9
```

though two last cases could be merged into one

$$dr_{10}(n) = 0, \quad \text{if } n = 0$$
$$dr_{10}(n) = 1 + (n - 1) \mod 9, \quad \text{if } n \neq 0$$

Java | Python3 — Copy

```
class Solution:
    def addDigits(self, num: int) -> int:
        return 1 + (num - 1) % 9 if num else 0
```

Complexity Analysis

- Time Complexity: $\mathcal{O}(1)$.
- Space Complexity: $\mathcal{O}(1)$.

Rate this article:

Previous Next

Comments: 11

Sort By

Type comment here... (Markdown is supported)

Preview Post

Himanshu_jain ★8 June 3, 2020 11:08 AM
amazing.
8 Share Reply

abhisheksirt ★5 June 5, 2020 1:14 PM
Really nice explanation
5 Share Reply

gargnaman ★6 June 6, 2020 1:36 PM
That was some cool Maths.
4 Share Reply

logan138 ★1917 June 4, 2020 2:51 PM
Cool solution
3 Share Reply

GTkds007 ★2 June 24, 2020 3:41 AM
What does the "?" in the return statement stand for?
2 Share Reply
SHOW 1 REPLY

akashmits ★2 June 18, 2020 11:32 PM
nice
2 Share Reply

shubhamrtp99 ★2 June 3, 2020 5:41 PM
Great explanation.
2 Share Reply

mihirkiran ★7 2 days ago
awesome solution
1 Share Reply

na_yan ★4 July 13, 2020 11:35 AM
just mind-blowing!!
1 Share Reply

myllinhtran ★1 July 9, 2020 2:36 PM
Didn't know this before. Very effective.
1 Share Reply