Recursion Playlist...

Leetcode - 273 Hard

Video - 4

Facebook
Instagram → codestorywithMIK

(Twitter) → CSwithMIK

→ codestorywithMIK

273. Integer to English Words

Hard | Topics | Companies | Hint

Convert a non-negative integer `num` to its English words representation.

Example 1:

```
Input: num = 123
Output: "One Hundred Twenty Three"
```

Example 2:

```
Input: num = 12345
Output: "Twelve Thousand Three Hundred Forty Five"
```

Example 3:

Example 3:

> **Input:** num = 1234567
> **Output:** "One Million Two Hundred Thirty Four Thousand Five Hundred Sixty Seven"

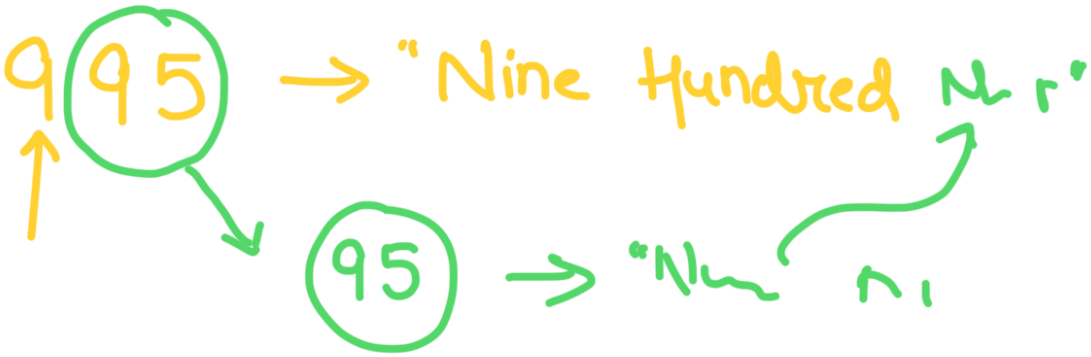
Constraints:

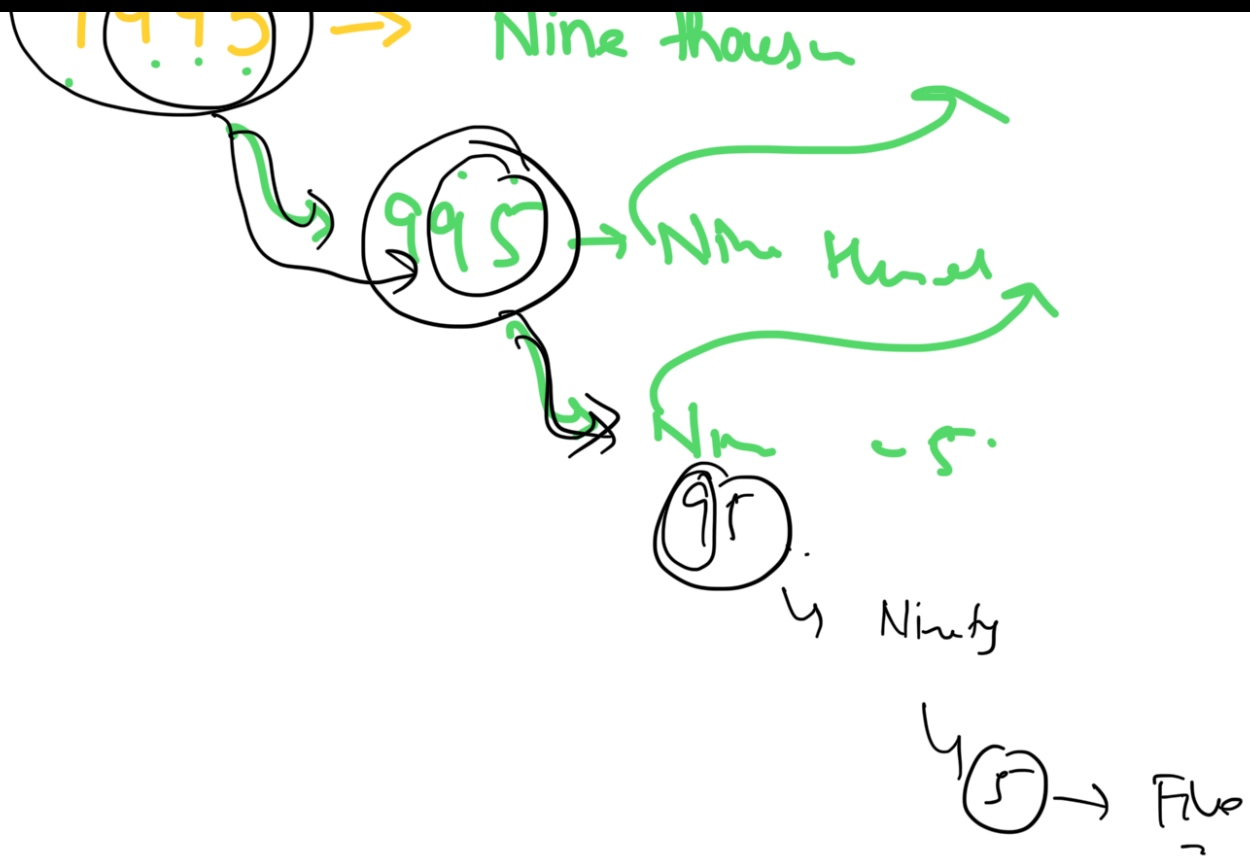
- `0 <= num <= 2^31 - 1`

Approach :-

Why Recursion?

95 → "Ninety Five"





Approach...

0 → "Zero"

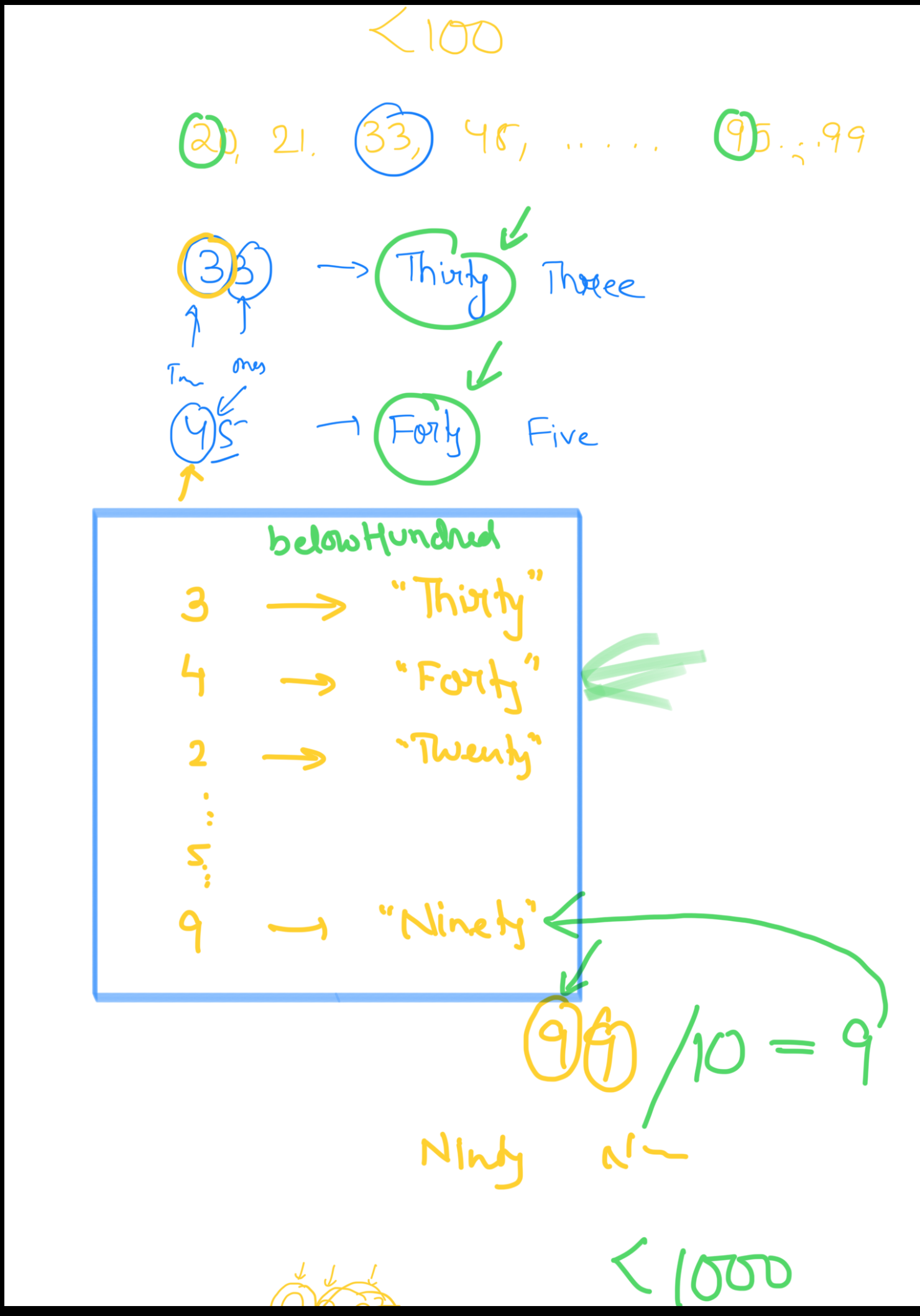
1 to 9 = 1 → "one"
2 → "Two"
3 → "Three

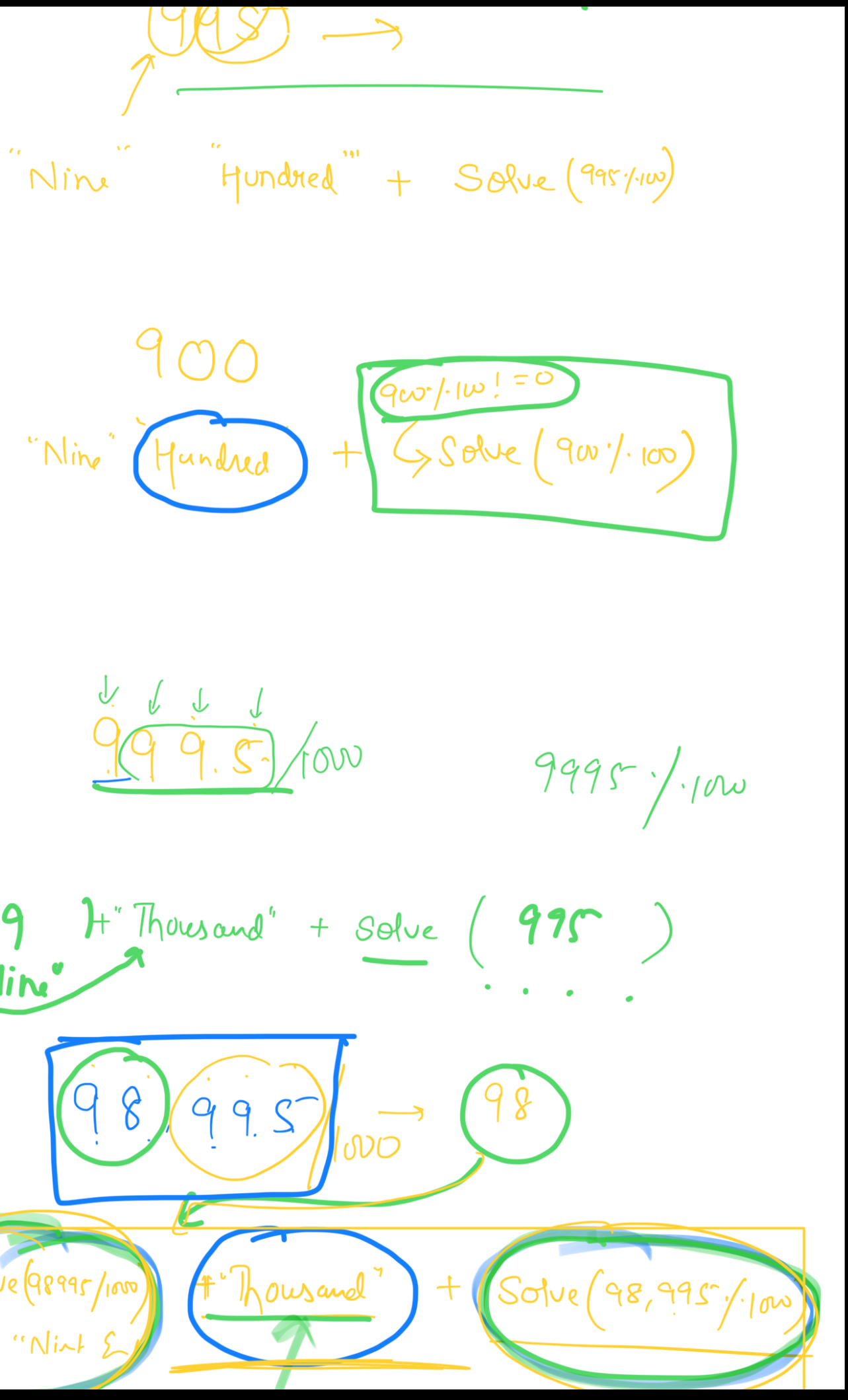
belowTen

belowTwenty

10, 11, 12, 13, 14, 15, 16, 17, 18, 19

10 → "Ten"
11 → "Eleven"
12 → "Twelve"
:


<100
20, 21, 33, 45, ... 95, ..., 99
33
Thirty
Three
Tens
Ones
45
Forty
Five
belowHundred
3 → "Thirty"
4 → "Forty"
2 → "Twenty"
9 → "Ninety"
99 /10 = 9
Ninty
<1000


995
"Nine" "Hundred" + Solve (995%100)
900
"Nine" Hundred + 900%100 != 0
Solve (900%100)
9995%100
Solve(9)+"Thousand" + Solve (995)
"Nine"
98 995 1000
98
Solve(98995/100
"Nine
#"Thousand" + Solve(98,995%100

"Hundred" < 1000 → 845/100

"Thousand" < 1,00,000 → 8451/100

"Million" < 10.00.00.000. → 100000

"Billion" >= 1000,000,000

$2^{31} - 1 = 2.14.74.83.647$

Time & Space:-

num = abcdef

↳ # of digits = $\log_{10}(num) + 1$

$\propto$ # of digits in num

$T.C = O(\log_{10}(num))$

S.C = O(1) → map

O(log10(num))