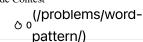
(/) Explore(/explore/) Problems(/problemset/all/) Interview Contest







User Accepted:

Total Accepted:

Total Submissions:

User Tried:

Difficulty:



0

0

0

0

Medium

6196. Partition String Into Substrings With Values at Most K

My Submissions (/contest/weekly-contest-326/problems/partition-string-into-substrings-with-values-at-most-k/submissions/)

Back to Contest (/contest/weekly-contest-326/)

You are given a string s consisting of digits from 1 to 9 and an integer k.

A partition of a string s is called **good** if:

- Each digit of s is part of **exactly** one substring.
- $\bullet\;$ The value of each substring is less than or equal to $\;k\;.$

Return the *minimum* number of substrings in a **good** partition of s. If no **good** partition of s exists, return -1.

Note that:

- The value of a string is its result when interpreted as an integer. For example, the value of "123" is 123 and the value of "1" is 1.
- A **substring** is a contiguous sequence of characters within a string.

Example 1:

```
Input: s = "165462", k = 60
Output: 4
Explanation: We can partition the string into substrings "16", "54", "6", and "2". Each substring has a value less than or equa
It can be shown that we cannot partition the string into less than 4 substrings.
```

Example 2:

```
Input: s = "238182", k = 5
Explanation: There is no good partition for this string.
```

Constraints:

- 1 <= s.length <= 10^5
- s[i] is a digit from '1' to '9'.
- $1 \le k \le 10^9$

```
JavaScript
                                                                                                                        σĎ
                                                                                                                             {f c}
1 v const minimumPartition = (s, k) ⇒ {
        let n = s.length, cur = 0, res = 0;
2
3 ▼
        for (let i = 0; i < n; i++) {
4 •
             if (cur * 10 + (s.charCodeAt(i) - '0'.charCodeAt(0)) <= k) {
 5
                 cur = cur * 10 + (s.charCodeAt(i) - '0'.charCodeAt(0));
 6 ▼
            } else {
                 if (cur == 0 \mid | cur > k) {
 7
 8
                     return -1;
9
                 }
10
                 else {
11
                     res++;
12
                     cur = s.charCodeAt(i) - '0'.charCodeAt(0);
                 }
13
14
            }
15
        if (cur > 0 && cur <= k) res++;
16
17
        return res;
18
    };
```

Custom Testcase Use Example Testcases

Submission Result: Accepted (/submissions/detail/868817759/)

More Details ➤ (/submissions/detail/868817759/)

Share your acceptance!

Copyright © 2022 LeetCode

Help Center (/support) | Jobs (/jobs) | Bug Bounty (/bugbounty) | Online Interview (/interview/) | Students (/student) | Terms (/terms) | Privacy Policy (/privacy)

United States (/region)