6138. Longest Ideal Subsequence

My Submissions (/contest/weekly-contest-305/problems/longest-ideal-subsequence/submissions/) Back to Contest (/contest/weekly-contest-305/)

User Accepted:	0
User Tried:	0
Total Accepted:	0
Total Submissions:	0
Difficulty:	Medium

You are given a string `s` consisting of lowercase letters and an integer `k`. We call a string `t` **ideal** if the following conditions are satisfied:

- `t` is a **subsequence** of the string `s`.
- The absolute difference in the alphabet order of every two **adjacent** letters in `t` is less than or equal to `k`.

Return *the length of the **longest** ideal string*.

A **subsequence** is a string that can be derived from another string by deleting some or no characters without changing the order of the remaining characters.

Note that the alphabet order is not cyclic. For example, the absolute difference in the alphabet order of `'a'` and `'z'` is `25`, not `1`.

Example 1:

```
Input: s = "acfgbd", k = 2
Output: 4
Explanation: The longest ideal string is "acbd". The length of this string is 4, so 4 is returned.
Note that "acfgbd" is not ideal because 'c' and 'f' have a difference of 3 in alphabet order.
```

Example 2:

```
Input: s = "abcd", k = 3
Output: 4
Explanation: The longest ideal string is "abcd". The length of this string is 4, so 4 is returned.
```

Constraints:

- `1 <= s.length <= 10^5`
- `0 <= k <= 25`
- `s` consists of lowercase English letters.

JavaScript

```javascript
const ord = (c) => c.charCodeAt();

const longestIdealString = (s, k) => {
    let dp = Array(26).fill(0);
    for (const c of s) {
        let max = 0, i = ord(c) - 97;
        for (let j = 0; j < 26; j++) {
            if (Math.abs(i - j) <= k) max = Math.max(max, dp[j]);
        }
        dp[i] = max + 1;
    }
    return Math.max(...dp);
};

```

Custom Testcase Use Example Testcases

Run Submit

Submission Result: Accepted (/submissions/detail/767246516/) More Details > (/submissions/detail/767246516/)

Share your acceptance!