(/) **Explore** ^Day 26   Problems(/problemset/all/)   Interview ^New   ^(/contest/) Contest   Discuss(/discuss/)   🚀 June LeetCoding Challenge 2021 (/discuss/general-Store discussion/655704/)   ☰ (/discuss/general- ☆ Premium (/subscribe? ref=nb_npl)   ▷+   🔔   👤

5799. Number of Wonderful Substrings

My Submissions (/contest/weekly-contest-247/problems/number-of-wonderful-substrings/submissions/)

Back to Contest (/contest/weekly-contest-247/)

A **wonderful** string is a string where **at most one** letter appears an **odd** number of times.

- For example, `"ccjjc"` and `"abab"` are wonderful, but `"ab"` is not.

Given a string `word` that consists of the first ten lowercase English letters (`'a'` through `'j'`), return the **number of wonderful non-empty substrings** in `word`. If the same substring appears multiple times in `word`, then count **each occurrence** separately.

A **substring** is a contiguous sequence of characters in a string.

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

Example 1:

```
Input: word = "aba"
Output: 4
Explanation: The four wonderful substrings are underlined below:
- "aba" -> "a"
- "aba" -> "b"
- "aba" -> "a"
- "aba" -> "aba"
```

Example 2:

```
Input: word = "aabb"
Output: 9
Explanation: The nine wonderful substrings are underlined below:
- "aabb" -> "a"
- "aabb" -> "aa"
- "aabb" -> "aab"
- "aabb" -> "aabb"
- "aabb" -> "a"
- "aabb" -> "abb"
- "aabb" -> "b"
- "aabb" -> "bb"
- "aabb" -> "b"
```

Example 3:

```
Input: word = "he"
Output: 2
Explanation: The two wonderful substrings are underlined below:
- "he" -> "h"
- "he" -> "e"
```

Constraints:

- `1 <= word.length <= 10`^{5}
- `word` consists of lowercase English letters from `'a'` to `'j'`.

JavaScript ▾   ⟨/⟩   ⟳   ⚙

```javascript
1 ▾ /**
2    * @param {string} word
3    * @return {number}
4    */
5 ▾ var wonderfulSubstrings = function(word) {
6
7   };
```

☐ **Custom Testcase**   (Use Example Testcases)

⏵ Run   ☁ Submit