2539. Count the Number of Good Subsequences

Description

A **subsequence** of a string is good if it is not empty and the frequency of each one of its characters is the same.

Given a string `s`, return *the number of good subsequences of* `s`. Since the answer may be too large, return it modulo `10^9 + 7`.

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.

Example 1:

```
Input: s = "aabb"
Output: 11
Explanation: The total number of subsequences is 2^4. There are five subsequences which are not good: "aab b", "a abb", "a a bb", "aa b b",
and the empty subsequence. Hence, the number of good subsequences is 2^4-5 = 11.
```

Example 2:

```
Input: s = "leet"
Output: 12
Explanation: There are four subsequences which are not good: "l ee t", "l eet", "leet", and the empty subsequence. Hence, the number of good
subsequences is 2^4-4 = 12.
```

Example 3:

```
Input: s = "abcd"
Output: 15
Explanation: All of the non-empty subsequences are good subsequences. Hence, the number of good subsequences is 2^4-1 = 15.
```

Constraints:

- `1 <= s.length <= 10^4`
- `s` consists of only lowercase English letters.