

Sherlock and Anagrams

Two strings are *anagrams* of each other if the letters of one string can be rearranged to form the other string. Given a string, find the number of pairs of substrings of the string that are anagrams of each other.

Example

$s = mom$

The list of all anagrammatic pairs is $[m, m]$, $[mo, om]$ at positions $[[0], [2]]$, $[[0, 1], [1, 2]]$ respectively.

Function Description

Complete the function *sherlockAndAnagrams* in the editor below.

sherlockAndAnagrams has the following parameter(s):

- *string s*: a string

Returns

- *int*: the number of unordered anagrammatic pairs of substrings in *s*

Input Format

The first line contains an integer *q*, the number of queries.

Each of the next *q* lines contains a string *s* to analyze.

Constraints

$$1 \leq q \leq 10$$

$$2 \leq \text{length of } s \leq 100$$

s contains only lowercase letters in the range `ascii[a-z]`.