Sherlock and Anagrams

Given a string S, find the number of "unordered anagrammatic pairs" of substrings. In other words, find the number of *unordered* pairs of substrings of S that are anagrams of each other.

Two strings are **anagrams** of each other if the letters of one string can be rearranged to form the other string.

Input Format
First line contains T, the number of testcases. Each testcase consists of string S in one line.

Constraints
$1 \le t \le 10$
$2 \le length(s) \le 100$
String s contains only the lowercase letters of the English alphabet.

Output Format
For each testcase, print the required answer in one line.

Sample Input 0

```
2
abba
abcd
```

Sample Output 0

```
4
0
```

Sample Input 1

```
5
ifailuhkqq
hucpoltgty
ovarjsnrbf
pvmupwjjjf
iwwhrlkpek
```

Sample Output 1

```
3
2
2
6
3
```

Explanation

Sample 0
Let's say $S[i,j]$ denotes the substring $S_i, S_{i+1}, \cdots, S_j$.

testcase 1:
For S=`abba`, anagrammatic pairs are: $\{S[1,1], S[4,4]\}$ (`a` and `a`), $\{S[1,2], S[3,4]\}$ (`ab` and `ba`), $\{S[2,2], S[3,3]\}$ (`b` and `b`) and $\{S[1,3], S[2,4]\}$ (`abb` and `bba`).

testcase 2:

No anagrammatic pairs.

Sample 1
Left as an exercise to you.