# Sherlock and Anagrams ☆

**Your Sherlock and Anagrams submission got 50.00 points.** Share Tweet

Try the next challenge

Problem Submissions Leaderboard Editorial Topics

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.

For 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. It must return an integer that represents the number of anagrammatic pairs of substrings in $s$.

sherlockAndAnagrams has the following parameter(s):

- s: a string .

**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**

$1 \leq q \leq 10$
$2 \leq |s| \leq 100$
String $s$ contains only lowercase letters $\in$ ascii[a-z].

**Output Format**

For each query, return the number of unordered anagrammatic pairs.

**Sample Input 0**

```
2
abba
abcd
```

**Sample Output 0**

```
4
0
```

**Explanation 0**

The list of all anagrammatic pairs is $[a, a]$, $[ab, ba]$, $[b, b]$ and $[abb, bba]$ at positions $[[0], [3]]$, $[[0, 1], [2, 3]]$, $[[1], [2]]$ and $[[0, 1, 2], [1, 2, 3]]$ respectively.

No anagrammatic pairs exist in the second query as no character repeats.

**Sample Input 1**

```
2
ifailuhkqq
kkkk
```

**Sample Output 1**

```
3
10
```

**Explanation 1**

For the first query, we have anagram pairs $[i, i]$, $[q, q]$ and $[ifa, fai]$ at positions $[[0], [3]]$, $[[8], [9]]$ and $[[0, 1, 2], [1, 2, 3]]$ respectively.

For the second query:

There are 6 anagrams of the form $[k, k]$ at positions $[[0], [1]]$, $[[0], [2]]$, $[[0], [3]]$, $[[1], [2]]$, $[[1], [3]]$ and $[[2], [3]]$.

There are 3 anagrams of the form $[kk, kk]$ at positions $[[0, 1], [1, 2]]$, $[[0, 1], [2, 3]]$ and $[[1, 2], [2, 3]]$.

There is 1 anagram of the form $[kkk, kkk]$ at position $[[0, 1, 2], [1, 2, 3]]$.

**Sample Input 2**

```
1
cdcd
```

**Sample Output 2**

```
5
```

**Explanation 2**

There are two anagrammatic pairs of length $1$: $[c, c]$ and $[d, d]$.

There are three anagrammatic pairs of length $2$: $[cd, dc]$, $[cd, cd]$, $[dc, cd]$ at positions $[[0, 1], [1, 2]]$, $[[0, 1], [2, 3]]$, $[[1, 2], [2, 3]]$ respectively.

Change Theme C++

```cpp
#include <bits/stdc++.h>

using namespace std;

int nc2(int n){
    return ((n)*(n-1))/2;
}
int sherlockAndAnagrams(string s) {
    int n=s.size();
    unordered_map<string,int> hmap;
    int ans=0;
    string substring;
    for(int i=1;i<n;i++){
        hmap={};
        for(int j=0;j<n-i+1;j++){
            substring=s.substr(j,i);
            sort(substring.begin(),substring.end());
            if(hmap.find(substring)!=hmap.end()){
                hmap[substring]+=1;
                cout<<substring<<" : ";
            }
            else{
                hmap[substring]=1;
            }
        }
        for(auto i:hmap){
            ans+=nc2(i.second);
        }

    }
    return ans;
}

int main()
{
    ofstream fout(getenv("OUTPUT_PATH"));

```

```
38      int q;
39      cin >> q;
40      cin.ignore(numeric_limits<streamsize>::max(), '\n');
```

Line: 19 Col: 25

Upload Code as File | Test against custom input | Run Code | Submit Code

You have earned 50.00 points!

These points will also count towards your progress in the Problem Solving Badge.

89% 444.69/475


## Congratulations

You solved this challenge. Would you like to challenge your friends?

Next Challenge

- Test case 0
- Test case 1
- Test case 2
- Test case 3
- Test case 4
- Test case 5
- Test case 6

Compiler Message

Success

Input (stdin) Download

```
2
abba
abcd
```

Expected Output Download

```
4
0
```