

# Simple Cipher

Problem Code: **SIMCIPH**


[Tweet](#)



Be the first of your friends to like this.

A substitution cipher is a cipher where you decide to replace all instances of 'A' with some character  $S[A]$  (where  $S[A]$  may or may not be equal to 'A'); all instances of 'B' with some character  $S[B]$ ; all instances of 'C' with some character  $S[C]$  and so on. Of course, there are no letters  $x$  and  $y$  such that  $S[x] = S[y]$ . In other words, all  $S[x]$  are mutually distinct.

For the purpose of this problem you may assume that all plain text and cipher text only contains of upper case english letters and spaces. Also, spaces are never replaced with any other character. Thus, the length and order of words in the plain text and cipher text remain exactly the same.

The key for a substitution cipher is, of course, the mapping from the set of alphabet on to itself. Consider the following substitution cipher key. Note that we do not show the mapping for each character to save space. The mappings for only those characters that will appear in the plain text in our example are shown.

```
A -> I
B -> L
D -> R
E -> E
H -> H
I -> A
L -> B
R -> K
S -> T
T -> W
```

The above key encrypts the plaintext "THIS IS READ ABLE" to "WHAT AT KEIR ILBE". Different substitution cipher keys (or mappings) may generate different cipher texts. We are interested in the set of cipher texts generated by some substitution cipher key, such that, each word in the cipher text is an anagram of the original word. In fact, we want you to compute the number of different cipher texts that can be generated such that each word in the cipher text is an anagram of the original word. Note that the problem asks for the number of such cipher texts, NOT the number of such substitution cipher keys.

For clarity, an anagram of a word is another word that can be formed by rearranging the letters of the given word. A rearrangement is not allowed to introduce new characters or delete any character. The original word is also considered an anagram of itself.

Consider the substitution cipher key below. Again, please note that we do not show mappings for each character.

```
A -> E
B -> L
D -> R
E -> A
H -> T
I -> S
L -> B
R -> D
S -> I
T -> H
```

[All Submissions \(/DI16R070/status/SIMCIPH\)](#)

**Successful Submissions**

**+**

This key encrypts the plaintext "THIS IS READ ABLE" to "HTSI SI DAER ELBA". Note how each word in the cipher text is an anagram of the original word. In fact, the plain text "THIS IS READ ABLE" can be mapped to 32 different cipher texts through substitution cipher scheme such that each word in the cipher text is an anagram of the respective word in the plain text.

---

## Input

The first line in input contains the number T, then number of test cases. The next T lines contain the description of plain text for each test case. Each description starts with the number of words in the plain text, followed by a single space, followed by words separated by single space characters. Each plain text will contain at most 4 words (at least 1). The description will contain no leading or trailing whitespaces. Each word in the plain text will contain at most 4 characters (at least 1). Each character in the plain text will be an upper case english letter. There may be at most 10 test cases in a test file. There are no empty lines separating test cases.

---

## Output

For each test case, print a number on a line by itself. This number should be the number of distinct cipher texts that can be achieved by any substitution cipher key such that each word in the cipher text is an anagram of the respective word in the plain text. Please note that the original plain text is also counted towards the set of possible cipher texts achievable.

---

## Sample Input

```
4
4 THIS IS READ ABLE
4 I LOVE CODE CHEF
3 CODE YOUR SELF
2 DONT COPY
```

---

## Sample Output

```
32
4
72
36
```

---

## Explanation

For the second sample input, the four possible cipher texts are "I LOVE CODE CFEH", "I LOVE CODE CHEF", "I VOLE CODE CFEH" and "I VOLE CODE CHEF".

Author: [directi\\_campus \(/users/directi\\_campus/\)](/users/directi_campus/)

Tags: [directi\\_campus \(/tags/problems/directi\\_campus/\)](/tags/problems/directi_campus/)

Date Added: 4-02-2015

Time Limit: 1 - 2 secs

Source Limit: 50000 Bytes

Languages: C, CPP 4.3.2, CPP 6.3, CPP14, JAVA, PYTH, PYTH 3.5

---

## Comments ▶

### **CodeChef (<http://www.codechef.com>) - A Platform for Aspiring Programmers**

CodeChef was created as a platform to help programmers make it big in the world of algorithms, **computer programming** and **programming contests**. At CodeChef we work hard to revive the geek in you by hosting a **programming contest** at the start of the month and another smaller programming challenge in the middle of the month. We also aim to have training sessions and discussions related to **algorithms**, **binary search**, technicalities like **array size** and the likes. Apart from providing a platform for **programming competitions**, CodeChef also has various algorithm tutorials and forum discussions to help those who are new to the world of **computer programming**.

### **Practice Section (<https://www.codechef.com/problems/easy>) - A Place to hone your 'Computer Programming Skills'**

Try your hand at one of our many practice problems and submit your solution in a language of your choice. Our **programming contest** judge accepts solutions in over 35+ programming languages. Preparing for coding contests were never this much fun! Receive points, and move up through the CodeChef ranks. Use our practice section to better prepare yourself for the multiple **programming challenges** that take place through-out the month on CodeChef.

### **Compete (<https://www.codechef.com/problems/easy>) - Monthly Programming Contests and Cook-offs**

Here is where you can show off your **computer programming skills**. Take part in our 10 day long monthly coding contest and the shorter format Cook-off **coding contest**. Put yourself up for recognition and win great prizes. Our **programming contests** have prizes worth up to INR 20,000 (for Indian Community), \$700 (for Global Community) and lots more CodeChef goodies up for grabs.

### **Programming Tools**

[Online IDE \(<https://www.codechef.com/ide>\)](https://www.codechef.com/ide)

[Upcoming Coding Contests \(<http://www.codechef.com/contests#FutureContests>\)](http://www.codechef.com/contests#FutureContests)

[Contest Hosting \(<http://www.codechef.com/hostyourcontest>\)](http://www.codechef.com/hostyourcontest)

[Problem Setting \(<http://www.codechef.com/problemsetting>\)](http://www.codechef.com/problemsetting)

[CodeChef Tutorials \(<http://www.codechef.com/wiki/tutorials>\)](http://www.codechef.com/wiki/tutorials)

[CodeChef Wiki \(<https://www.codechef.com/wiki>\)](https://www.codechef.com/wiki)

### **Practice Problems**

[Easy \(<https://www.codechef.com/problems/easy>\)](https://www.codechef.com/problems/easy)

[Medium \(<https://www.codechef.com/problems/medium>\)](https://www.codechef.com/problems/medium)

[Hard \(<https://www.codechef.com/problems/Hard>\)](https://www.codechef.com/problems/Hard)

[Challenge \(<https://www.codechef.com/problems/challenge>\)](https://www.codechef.com/problems/challenge)

[Peer \(<https://www.codechef.com/problems/extcontest>\)](https://www.codechef.com/problems/extcontest)

[School \(<https://www.codechef.com/problems/school>\)](https://www.codechef.com/problems/school)

[FAQ's \(<https://www.codechef.com/wiki/faq>\)](https://www.codechef.com/wiki/faq)

### **Initiatives**

[Go for Gold \(<http://www.codechef.com/goforgold>\)](http://www.codechef.com/goforgold)

[CodeChef for Schools \(<http://www.codechef.com/school>\)](http://www.codechef.com/school)

[Campus Chapters \(\[http://www.codechef.com/campus\\\_chapter/about\]\(http://www.codechef.com/campus\_chapter/about\)\)](http://www.codechef.com/campus_chapter/about)

[Domain Registration in India \(<http://www.bigrock.in/>\)](http://www.bigrock.in/) and [Web Hosting \(<http://www.bigrock.com/web-hosting/>\)](http://www.bigrock.com/web-hosting/) powered by BigRock