Recently, there has been a breach of user information from the mega-popular social network *Secret Network*. Among the confidential information are the passwords of all users.

Mihael, a young student who has been exploring computer security lately, found the whole thing really interesting. While experimenting with the social network, he found another security breach! When you input any string of characters that contains a substring equal to the actual password, the login will be successful. For example, if the user whose password is *abc* inputs one of the strings *abc*, *abcd* or *imaabcnema*, the system will successfully log him in, whereas the login will fail for *axbc*.

Mihael wants to know how many ordered pairs of different users exist such that the first user, using their own password, can login as the second user.

INPUT
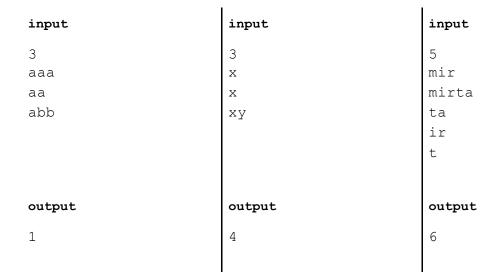
The first line of input contains the positive integer N ($1 \leq N \leq 20\ 000$), the number of users. Each of the following N lines contains the user passwords. The passwords consist of at least one and at most 10 lowercase letters of the English alphabet.

OUTPUT

The first and only line of output must contains the number of ordered pairs from the task.

SCORING

In test cases worth 40 points total, it will hold $1 \leq N \leq 2000$.

SAMPLE TESTS

input	input	input
3	3	5
aaa	x	mir
aa	x	mirta
abb	xy	ta
		ir
		t

output	output	output
1	4	6

Clarification of the second test case:
The first user can login as the second user, the second user can login as the first, and the third user can login as both the first and the second user.