Problem H. P-P-Palindrome

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	512 megabytes

Given n strings $S_1, S_2, \ldots S_n$, you need to calculate the number of different *P-P-Palindromes* given by these n strings.

A palindrome is a string that can be read the same from left to right and from right to left. For example, "`a`", "`level`", and "`otto`" are palindromes, while "`aab`" and "`icpc`" are not.

A *P-P-Palindrome* is an ordered pair of **nonempty** palindromes (P, Q) such that both P and Q are the substrings of some in $S_1, S_2, \ldots S_n$ and $P + Q$ is also a palindrome. Here $P + Q$ means concatenating P and Q in order, or more specifically, let $P = p_1 p_2 \ldots p_{|P|}$ and $Q = q_1 q_2 \ldots q_{|Q|}$, then $P + Q = p_1 p_2 \ldots p_{|P|} q_1 q_2 \ldots q_{|Q|}$, where $|S|$ is the length of string S.

Note that two *P-P-Palindromes* are considered different if and only if P differs or Q differs.

Input

The first line contains an integer n ($1 \le n \le 10^6$), indicating the number of given strings.

Then n lines follow, the i-th of which contains a string S_i ($1 \le |S_i| \le 10^6$) consisting of lowercase English letters only.

It is guaranteed that the total length of the given strings does not exceed 10^6.

Output

Output a line containing a single integer, indicating the number of different *P-P-Palindromes* given by the n strings.

Examples

standard input	standard output
2 aaaa aaa	16
3 abaaa abbbba bbbaba	28