



Flipping Out

Here is an interesting way of generating coin flips that looks random, but it's not! First, generate a set of *patterns* (consecutive runs of coin flips) that are either heads or tails. Then, to generate each new coin flip, look back at all of the past coin flips in order, and count the number of times that each pattern occurs. Add them all up, and if that number is even, then the next flip is tails, otherwise the next flip is heads.

Consider these three *patterns*:

HTH
THH
T

They will generate this sequence of flips: **THHTHTHT...**

- 1) **T** because at the start there are 0 occurrences of any of the *patterns*, and 0 is even.
- 2) **H** because there is 1 occurrence of the *pattern* **T**
- 3) **H** because there is still only 1 occurrence of the *pattern* **T**
- 4) **T** because there is one **T** and one **THH**, so 2 total
- 5) **H** because there are 2 **Ts** and one **THH**, 3 total
- 6) **T** because there are 2 **Ts**, one **THH** and one **HTH**, 4 total
- 7) **H** because there are 3 **Ts**, one **THH** and one **HTH**, 5 total
- 8) **T** because there are 3 **Ts**, one **THH** and 2 **HTHs**, 6 total (it's OK that the **HTHs** overlap)

Suppose that a single *pattern* was missing from the set of *patterns*. From a sequence of flips, can you determine how many options there are for the missing *pattern*? Note that the missing pattern cannot be a duplicate of one of the given patterns, and also cannot be empty.

Input

Each input will consist of a single test case. Note that your program may be run multiple times on different inputs. Each test case will begin with a line containing a single integer n ($1 \leq n \leq 100,000$), representing the number of *patterns*.

The next n lines will each contain a string consisting only of capital **Ts** and capital **Hs**. The first $n-1$ of these strings are the *patterns*, and the last is a sequence of generated flips. The sum of the lengths of all n strings will be $\leq 10^6$. All strings will be unique, and no strings will be empty.



2017 ACM ICPC Southeast USA Regional Contest

Output

Output a single integer, which is the number of options for the missing *pattern*. Output **-1** if there are infinitely many options.

Sample Input

Sample Output

2 H HTTTT	0
1 THHHHH	1
5 H TTH HHTHT TH TTHTTT	1
5 TTH H T HT THTTHTTH	2