



Problem C (Program filename: C.cpp, C.dpr, or C.java)

Fixing Codes

A binary string is a string of characters from the set $\{0, 1\}$. A code is a multiset of binary strings (i.e., a string can be repeated arbitrary number of times). A fixed code is a code that none of its strings is a prefix of another string. We say that a code $A = \{a_1, a_2, \dots, a_n\}$ is extended to code $B = \{b_1, b_2, \dots, b_n\}$ if and only if for $1 \leq i \leq n$, a_i be a prefix of b_i . The cost of this extension is $\sum_{i=1}^n |b_i| - |a_i|$ where $|a_i|$ is the number of characters in a_i .

For this problem you are given a fixed code C , and a new binary string s . You have to find the minimum needed cost to extend the code $C \cup \{s\}$ into a fixed code. In other words, you are to append the minimum number of bits to zero or more codes in $C \cup \{s\}$ to make it a fixed code.

Input (filename: C.in)

The first line of the input contains a single integer t ($1 \leq t \leq 20$) which is the number of test cases in the input. For $1 \leq i \leq t$ the line $i + 1$ consists a nonzero number of binary strings. The number of binary strings in each line is at most 41, and the length of each binary string is no more than 40 characters. The last string in each line stands for the new incoming string s and the other strings in that line make the fixed code of the relevant test case.

Output (filename: C.out)

The output consists m lines. The solution to i th test case should be written in the line i of the output.

Sample Input

```
2
001 01 00
000 001 010 011 100 101 110 1
```

Sample Output

```
1
2
```