Problem T
Non-Decreasing Bit Strings
Time limit: 5 seconds
Memory limit: 256 megabytes

Problem Description

Bit strings are strings consisting of only 0 and 1. For instances, 00, 1001, and the empty string ϵ ("" in C/C++/Java/Python programming languages) are all bit strings. For convenience, we use $s[i]$ to denote the i-th bit of bit string s. For two bit strings s and t, we say $s \leq t$ if one of the following conditions holds.

- s is a prefix of t.

- There exists a positive integer k such that $s[k] < t[k]$ and $s[j] = t[j]$ for every $j < k$.

For examples, 001$\leq$01011 and 00$\leq$000. Note that $\epsilon \leq t$ for any bit string t.

You are given a sequence $s_1, \ldots, s_n$ of bit strings, and you are allowed to modify the strings by removing some characters from their right ends. For example, you may obtain 110, 11, 1 and ϵ from 1100. However, you cannot get 100 from 1100. You are asked to turn the sequence into a non-decreasing one. That is, for every adjacent bit strings s_i and s_{i+1}, we have $s_i \leq s_{i+1}$. Please write a program to comput how many bits are required to be removed.

Input Format

The first line of the input contains an integer T ($T \leq 10$) indicating the number of test cases. Each test case consists of two parts. The first part is one line containing a positive integers n where $n \leq 10^5$. The second part consists of n lines. The i-th line is the bit string s_i. You may assume the input file is no more than 15 megabytes.

Output Format

For each test case, output how many bits are required to be removed to turn $s_1, \ldots, s_n$ into a non-decreasing sequence.

Sample Input

```
2
3
01
001
000
4
1
01
001
000
```

Sample Output

```
2
3
```