Fan Switches

Problem Statement

It's holiday time at DA-IICT, and all the students are back home, enjoying themselves. The professors too are off-campus, since they have no work to do and are finally free. Seems like the whole campus is empty, and everyone's having a lot of fun, but this is not true - the RC is still open, and Rahu, the bored librarian, still has to sit at his desk and work, even though there are no books to issue, and no late fines to collect.

It's 9:30 AM and Rahu has just unlocked the doors of the RC. He sees n fan switches on the wall (numbered 0 to n-1), all initially off. Since no one's in the RC at the moment, he knows that he only needs to switch on a subset of these switches, and since he has nothing to do and a lot of time to kill, he decides to turn this into a game. He decides to select a particular switch (say the i^{th} switch), and then toggle all the switches from the i^{th} one to the $n-1^{th}$ one (both inclusive). This means that for any given switch in the range [i, n-1], if it was initially on, it will now be off, and vice versa. He does so repeatedly until only the appropriate fans are on, and all the rest are off.

You are given the final configuration of the switches required. Assuming that all the fans are initially off, please print out the minimum moves required by Rahu to reach this final configuration.

Note : Here, one **move** is defined as selecting a particular switch (say the i^{th} one), and then toggling all the switches in the range [i, n-1]

Input

The first line contains a single integer T (T <= 100), denoting the number of test cases. T lines of test cases follow.

Each test case consists of a single line containing the binary string **finalConfig**, the final configuration of the switches required. The length of this string does not exceed 1000 characters.

The i^{th} character of this string may either be a '0' or a '1'. A '0' at this position indicates that the i^{th} switch is off, otherwise a '1' at this position indicates that the i^{th} switch is on, in the final configuration.

Output

For each test case, output a single line containing the minimum number of **moves** required to reach this final configuration, from the initial configuration wherein all the fan switches are off. Please note the definition of a "**move**" at the end of the problem statement.

Example

Input :

4
0011
000
0100
111000111

Output :

1
0
2
3

Explanation:

1. In the first test case, all you require is one move wherein you select the 2^{nd} switch, and thus toggle the 2^{nd} and 3^{rd} switches, turning them both on.
2. In the second test case, no operations are needed (since all the fans are switched off in the final configuration anyway)
3. In the third test case, a possible sequence of switches selected is - the 1^{st}, and then the 2^{nd}. The sequence of all intermediate configurations achieved is as follows : 0000 (initially) -> 0111 (after selecting switch #1) -> 0100 (after selecting switch #2)