MIT and TIM

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

Busy Beaver is studying Archaeology and Materials at MIT! He has been studying ancient inscriptions made only of the letters `M`, `I`, and `T` and notices the following rule: any time the substring `MIT` appears, it may be rearranged into `TIM`, and any time `TIM` appears, it may be rearranged back into `MIT`.

Now Busy Beaver wants to form as many occurrences as possible of his favorite pattern, `MITIT`, in the string. Help him determine the maximum number of contiguous substrings equal to `MITIT` that can appear after performing any number of operations (possibly zero).

Input

The first line contains an integer T ($1 \leq T \leq 10^5$) — the number of test cases.

The only line of each test case contains a string of length at most 10^5 consisting of the characters `M`, `I`, and `T`.

The total length of all strings does not exceed 10^5.

Output

For each test case, output a single integer — the maximum number of substrings `MITIT` that can appear after performing any number of operations.

Example

standard input	standard output
6	1
TITIMMIT	2
TITITIMITIMTMTMTMMITMI	0
MIMTITMTIMTITMTITITMTI	0
ITMTTMITMITMTMTITITMIM	0
MMITMITTTIMTITITTTITIT	5
MITITIMIMIMITITITITIMIMIMIMIMITITITITIIMITMTIMTIITMITMTIMTITIITITMTIMI	

Note

In the first test case, we can do the following operations: `TITIMMIT` $\rightarrow$ `TIMITMIT` and `TIMITMIT` $\rightarrow$ `MITITMIT`. We can prove that it is impossible to construct two copies of `MITIT` inside this string.