

I: License Plates

Source file name: `plates.c`, `plates.cpp`, `plates.java`, or `plates.py`

Author: Edwin Niño

On their daily drive to high school, Antonia and her mom Maria, used to challenge each other using the license plates of other cars. In their country, license plates always contain a string of exactly three letters. The challenge was to tell the longest word that contains as a subsequence the string of letters found in a license plate. For example, if a license plate contained the string OMI, then PROGRAMMING and COLOMBIA could be used as answers to the challenge. However, the former word was preferred because it was longer.

Those were the old days. Antonia is now a Computer Science student and Maria wants to make some changes to keep the challenge interesting to her daughter. In the new version of the challenge, Maria writes down a word S and Antonia must calculate the number of different three-letter strings that are each a subsequence of S .

You must write a program to help Maria calculate in advance the output of the new version of the challenge.

Input

The first line of input contains T ($T \geq 0$) indicating the number of test cases. Each test case consists of one single line containing a string S ($1 \leq |S| \leq 10^5$). You may assume that S is made only of uppercase letters from the English alphabet (i.e., A, B, ..., Z) and that there is no blank line between test cases.

The input must be read from standard input.

Output

For each test case, print a single line containing the number of different three-letter strings that are each a subsequence of S .

The output must be written to standard output.

Sample Input	Sample Output
5	110
PROGRAMMING	46
COLOMBIA	9
NUEVE	1
AAAAAAA	0
PQ	