1299 – Fantasy Cricket

ICC World Cup 2011 has just finished. During the world cup, lots of cricket fans were playing an online game named "Fantasy Cricket".

The score board of fantasy cricket looks like the following image. After each match of the world cup the score board of fantasy cricket updates.

Rank		Manager	Team Name	Total
1	▲	Mind the Gap	**Mind the Gap XI**	12384
2	▼	Old Man	**Old Man XI**	12344
3	▲	Legend	**Legend XI**	11611
4	▼	Far Cry	**Far Cry XI**	11221
5	▶	Accepted	**Accepted XI**	10111
6	▲	WA	**WA XI**	10001

Figure 1

Each player plays a role of a manager here. In the rank list there is a symbol besides each manger. There are three kinds of symbols. These are explained in the following table.

Symbol	Explanation	ASCII Representation
▲	The rank of the player has upgraded after last match, i.e. if the current rank of the player is **K**, the rank of the player before the last match was greater than **K**.	**U**
▼	The rank of the player has downgraded after the last match, i.e. if the current rank of the player is **K**, the rank of the player before the last match was less than **K**.	**D**
▶	The rank of the player has not changed after the last match, i.e. if the current rank of the player is **K**, the rank of the player before the last match was also **K**.	**E**

You will be given such a score board after a particular match. Can you determine any possible valid ordering of the players exactly before that round? For this problem you have to print the number of possible ordering before the last match.

Here is an example –

Rank		Manager
1	▲	A
2	▼	B
3	▲	C
4	▼	D

Figure 2

Possible Previous Order 1	Possible Previous Order 2
B	B
A	D
D	A
C	C

Figure 3

For this rank (figure 2), only two different ordering are possible (figure 3) before the last match which comply the current ordering with the symbols.

Name of the managers are not important for this problem. So, for a scoreboard, you will be given a sequence of ASCII representation of the symbols (stated above), i.e. you will be given a string which only contains **'U', 'D'** and **'E'**.

Input

Input starts with an integer **T (≤ 300)**, denoting the number of test cases.

Each case starts with a line containing a string. The length of the string will be between **1** and **1000**. The string will contain characters from **{'U', 'D', 'E'}**.

Output

For each case, print the case number and the number of possible orderings modulo **1000000007**.

Sample Input	Output for Sample Input
3 UDUD EEE DU	Case 1: 2 Case 2: 1 Case 3: 0