Problem B. Byteland Trip

Input file:	`standard input`
Output file:	`standard output`
Time limit:	6 seconds
Memory limit:	512 megabytes

The Byteland Empire consists of n planets arranged in a row. Planets are numbered from 1 to n. You are planning your journey to Byteland, but lack of convenient means of transportation annoys you.

Nowadays the only transport available in the Byteland Empire is teleportation. Unfortunately, the teleportation system in Byteland is quite tricky to use. Each planet has a single teleportation center, which can be either forward-bound or backward-bound. The difference between forward-bound and backward-bound teleportation centers is that from a forward-bound teleportation center you can travel to any planet with greater index, while from a backward-bound teleportation center you can travel to any planet with smaller index. Formally, if planet i has a forward-bound teleportation center, you can use it to travel to any planet $j > i$, while if planet i has a backward-bound teleportation center, you can travel to any planet $j < i$.

Your dream is to visit every planet **exactly once**. Your trip can start on any planet. For each planet i of the Byteland Empire you would like to know the number of valid trips that start at any other planet, visit every planet exactly once, and end on the planet i. As it often happens with you, the values you want to compute may be too large, so you are only interested in their remainders modulo $10^9 + 7$.

Input

The input consists of a single string s of length n ($1 \le n \le 5000$) — the description of teleportation centers on all planets. Each character of this string is either '<' or '>'. The i-th character is equal to '<' if the i-th planet has a backward-bound teleportation center, and is equal to '>' if it has a forward-bound teleportation center.

Output

Print n integers — for each planet i the number of valid trips that end on planet i modulo $10^9 + 7$.

Examples

standard input	standard output
>><	1 2 1
><>>><	1 2 2 4 8 1

Note

In the first sample, the only valid trip that ends on planet 1 is $2 \to 3 \to 1$, all valid trips that end on planet 2 are $3 \to 1 \to 2$ and $1 \to 3 \to 2$, and the only valid trip that ends on planet 3 is $1 \to 2 \to 3$.