

Problem 4 – Fake Text Markup Language

Joro's trying to top the leaderboard on "weirdest things by the lectors" (yep, couldn't think of a better translation). Last time he used Paint for a color picker on a CSS lecture, now he's inventing a whole new markup language – the Fake Text Markup Language. It looks a bit like HTML – it has opening and closing tags, and those tags define the formatting of text... on the console! So, if we receive an FTML formatted text, we need to print the text content according to the following rules:

- Tags can be opening and closing, e.g.: `<upper>` and `</upper>`
- Tags can be nested, e.g.: `<upper> this is a <lower> nested tag</lower> in another tag </upper>`
- Nesting is measured in "**depth**", in this case `<lower>` is **deeper** than `<upper>`
- Tags **cannot** "intersect", e.g.: `<upper> this is <lower> very </upper> wrong </lower>` is **not valid**
- Tags don't appear in the resulting text, but they affect it
- When tags are nested, the effect of the "**deepest**" is considered first, then the effect of the less deeper and so on
 - e.g.: `<upper>Before nested <lower>Inside nested</lower> After nested</upper>` will apply the effects of `<lower>` and then of `<upper>`
 - The end result will be: `BEFORE NESTED INSIDE NESTED AFTER NESTED`
- The `<upper>` tag converts text to its uppercase variant
 - e.g.: `<upper>tExT</upper>` results in: `TEXT`
- The `<lower>` tag converts text to its lowercase variant
 - e.g.: `<lower>tExt</lower>` results in: `text`
- The `<toggle>` tag:
 - if a character is uppercase, it converts it to lowercase
 - if a character is lowercase, it converts it to uppercase
 - e.g. `<toggle>tExT<toggle>` results in: `TeXt`
- `<upper>`, `<lower>` and `<toggle>` tags don't affect punctuation or whitespaces
- The `` tag deletes all text in it
 - e.g.: `this is this is deleted some text` results in: `this is some text`
- The `<rev>` tag reverses all text in it
 - e.g.: `123 reversed is <rev>123</rev>` results in: `123 reversed is 321`
- The FTML keeps all whitespaces and new lines (it doesn't remove them like HTML)
 - the only exception is, obviously, the `` tag

Write a program, which formats code, according to the rules described.

Input

On the first line of the input, there will be the number N – the number of lines in the FTML text.

On each of the next lines there will be a line of the FTML text, consisting of English letters, punctuation and whitespaces (and of course FTML tags)

Output

On the console, you should print the lines of the formatted text.

Constraints

- $0 < N < 500$
- The text input will NOT contain any '<' or '>', other than the ones for the tags
- Allowed working time for your program: 0.1 seconds. Allowed memory: 16 MB.

Examples

Input example	Output example
2 So<rev><upper>saw</upper> txet em</rev> <lower><upper>here</upper></lower>	Some text WAS here
3 <toggle><rev>ERa</rev></toggle> you <rev>noc</rev><lower>FUSED</lower> <rev>?<rev>already </rev></rev>	Are you confused already ?