

Problem D. Delete, delete, delete, ...

| | |
|----------------------|--|
| Source file name: | Delete.c, Delete.cpp, Delete.java, Delete.py |
| Input: | Standard |
| Output: | Standard |
| Time / Memory limit: | 1/2/2 (C++/Java/Python) second(s) / 64 megabytes |
| Author(s): | Hugo Humberto Morales Peña - UTP Colombia |

Professor Humbertov Moraloov has had plenty of free time during his latest vacation and has spent it writing programming challenges to torture (I mean, to properly evaluate the knowledge) his new programming students.

Professor Moraloov has hidden a secret message in a text (that is, in a string of characters). To uncover the hidden message, all occurrences of a given list of characters must be removed from the text. Additionally, it is known that the hidden message does not begin or end with one or more whitespace characters and that it does not contain the same symbol consecutively.

Input

Humbertov Moraloov's image by ChatGPT "Studio Ghibli" style

The input contains a single test case consisting of two lines.

The first line contains a text of minimum length 10 and maximum length 10^6 , which may include lowercase English letters, digits from 0 to 9, and the space character (ASCII 32).

The second line contains a string of characters (symbols) that do not belong to the message, enclosed between an opening bracket ("[" and a closing bracket ("]"), with a length between 1 and 36. It may contain lowercase English letters, digits from 0 to 9, and even the space character (ASCII 32).

Output

Print in a single line the string representing the hidden message. The line must end with a newline character ("n").

Note: The shortest hidden message consists of a single letter or digit. Moreover, the hidden message contains no leading or trailing whitespace.

Example

| Input |
|---|
| king lokif bfbdddaafbbtgffaf kong bsbtbgrgugfcgtfugrfegg bigbig kjong kiko [bfghijklmno] |
| Output |
| data structure |
| Input |
| auub attb appb aoob appb aeeb annb a22b a00b a22b a55b [abc] |
| Output |
| utpopen2025 |

Use fast I/O methods