Problem G. Gmoogle

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	256 megabytes

You are hired to create alpha version of the new searching engine named GMoogle. Alpha version should work with the content, represented as a database of sentences:

- Content is merged into line S, consisting of characters '`a`'-'`z`', '`A`'-'`Z`', spaces, notation marks ("`.!?`") (quotes are not counted) and decimal digits.
- If one of characters "`.!?`" presents in the S, then it denotes the end of the sentence, except for one special case: if first non-space character after '`.`' is lowercase English letter, then it is an abbreviation sign but not the end of the sentence; for example, string "`I like tea in a 500 ml. cup`" contains one sentence, but strings "`Cup is 500 ml. I want it`" and "`Cup is 500 ml. 500 ml is great for me`" contains two sentences).
- First non-space character after the end of sentence is considered as the first character of the new sentence.
- *word* is contiguous sequence of characters '`a`'-'`z`', '`A`'-'`Z`', delimited by spaces, notation signs or beginning/end of the sentence/string. It is guaranteed that digits can not be neighbors of the letters, i.e. sequences like "`10ml`" or "`R2D2`" are illegal.
- S may contain the sentences containing no words. It is guaranteed that S does not contains two or more characters "`.!?`" in a row.

After the content is indexed, users make requests. Each request can be represented as a string q, consisting of one or more words (definition of the word is given above). Words are separated by arbitrary number of spaces (1 or more), heading and trailing spaces are possible.

Your program has to print all sentences from S, where all words from q are presented (in any order). Words are considered equal, if all the letters at the corresponding positions are the same (case insensitive, i.e. '`B`' and '`b`' are considered the same.

Input

First line of the input contains non-empty line S, consisting of no more than 1000 characters. Next line contains one integer n ($1 \le n \le 100$) — number of the requests. Then n requests $q_1, \ldots, q_n$ follow, each on separate line in the format, described above. Note that in S and q_i trailing and heading spaces are **allowed**.

Output

For each request $q_1, q_2, ..., q_n$ print the request at the separate line. Then print the list of found sentences in same order they present in S, one sentence per line. Requests and answers are printed in the quotes; answers are preceeded by single '`-`' and single space; heading and trailing spaces must be eliminated. Look the sample for clarify.

Example

standard input	standard output
Hello everyone. I want 2 coffee if you have it. I like coffee very much. 4 HELLO Coffee much coffee VoDka	Search results for "HELLO": - "Hello everyone." Search results for "Coffee": - "I want 2 coffee if you have it." - "I like coffee very much." Search results for "much coffee": - "I like coffee very much." Search results for "VoDka":