Digital Pakistan Speed Programing Competition Online Mock Contest

Instructions

- Do not open the booklet unless you are explicitly told to do so. You can only read these instructions below.
- If you have any question regarding the problems, seek a clarification from the judges using DOMJudge.
- Before submitting a run, make sure that it is executable via command line. For Java, it must be executable via "javac" and for GNU C++ via "g++". Java programmers need to remove any "package" statements and source code's file name must be the same as of main class. C++ programmers need to remove any getch() / system("pause") like statements.
- Do not attach input files while submitting a run, only submit/attach source code files, i.e., *.java or *.cpp or *.py.
- Language supported: C/C++, Java and Python3
- Source code file name should not contain white space or special characters.
- You must take input from Console i.e.: Standard Input Stream (stdin in C, cin in C++, System.in in Java, stdin in Python)
- You must print your output to Console i.e.: Standard Output Stream (stdout in C, cout in C++, System.out in Java)
- Please, don't create/open any file for input or output.
- Please strictly meet the output format requirements as described in problem statements, because your program will be auto judged by computer. Your output will be compared with judge's output byte-by-byte and not tolerate even a difference of single byte. So, be aware! **Pay special attention to spaces, commas, dots, newlines, decimal places, case sensitivity etc.**
- All your programs must meet the time constraint specified.
- The decision of judges will be absolutely final.

Problem 03: Calligraphy Crisis

Time limit: 6 seconds

Ahmed is a renowned calligraphy artist who specializes in crafting beautiful Arabic and English letter designs. His shop is famous for custom calligraphy, where customers order personalized artwork with their names or meaningful words.

One day, while Ahmed was busy preparing an order for an important customer, a mischievous kid snuck into his shop and shuffled the letters on his unfinished works! Instead of the elegant, flowing designs, the letters were now in complete disarray.

Ahmed was furious, but there was no time to redraw everything from scratch. Instead, he decided to salvage whatever he could. He needs to quickly check whether any of his existing calligraphy pieces contain a prefix that is an anagram of a customer's requested word.

Can you help Ahmed restore order in his shop by finding which of his artworks can still be used for the incoming customer requests?

Definitions

Prefix: A prefix of a word is any substring of that word that starts at the beginning and can include the whole word itself.

Example: All possible prefixes of "trap" are "t", "tr", "tra", and "trap".

Anagram: Two strings are anagrams of each other if one can be rearranged to form the other.

Example: "abc" and "bca" are anagrams, but "abc" and "bcaa" are not.

Input

The first line contains an integer N — the number of calligraphy pieces Ahmed has in his shop.

The next N lines contain strings S_i representing a calligraphy piece.

The next line contains an integer Q — the number of customer requests.

The next Q lines contain a single word Q_i, representing a customer's request.

Output

For each request Q_i, print the number of calligraphy pieces that contain a prefix that is an anagram of Q_i.

If no such calligraphy piece exists, print "-1".

Limits

$1 \leq N, Q \leq 10^5$

The length of S_i and Q_i is in the inclusive range $[1, 20]$.

S_i and Q_i only include lowercase english alphabets.

The calligraphy pieces are unique, i.e: all N values of S_i are unique.

Sample Input	Sample Output
5 rat art tarp part trap 3 tra ar tp	4 2 -1