**Virus Outbreak**(100 Marks)

In the Martian land faraway, a new virus has evolved and is attacking the individuals at a fast pace. The scientists have figured out the virus composition, **V**. The big task is to identify the people who are infected. The sample of **N** people is taken to check if they are POSITIVE or NEGATIVE. A report is generated which provides the current blood composition **B** of the person.

**POSITIVE or NEGATIVE ?**

If the blood composition of the person is a subsequence of the virus composition V, then the person is identified as POSITIVE otherwise NEGATIVE.

Example:

Virus Composition, V = coronavirus

Blood Composition of the person , B = ravus

The person in question is POSITIVE as B is the subsequence of the V.

The scientists are busy with their research for medicine and request you to build a program which can quickly figure out if the person is POSITIVE or NEGATIVE. They will provide you with the virus composition V and all the people’s current blood composition. Can you help them?

**Note:** The virus and blood compositions are lowercase alphabet strings.

**Input Format**

The first line of the input consists of the virus composition, V

The second line of he input consists of the number of people, N

Next N lines each consist of the blood composition of the ith person, Bi

**Constraints**

1<= N <=10

1<= |B|<= |V|<= 10^5

**Output Format**

For each person, print POSITIVE or NEGATIVE in a separate line

Sample TestCase 1

Input

coronavirus

3

abcde

crnas

onarous

Output

NEGATIVE

POSITIVE

NEGATIVE

**Time Limit(X):**

0.50 sec(s) for each input.

**Memory Limit:**

512 MB

**Source Limit:**

100 KB

**Allowed Languages:**

C, C++, C++11, C++14, C#, Java, Java 8, Kotlin, PHP, PHP 7, Python, Python 3, Perl, Ruby, Node Js, Scala, Clojure, Haskell, Lua, Erlang, Swift, VBnet, Js, Objc, Pascal, Go, F#, D, Groovy, Tcl, Ocaml, Smalltalk, Cobol, Racket, Bash, GNU Octave, Rust, Common LISP, R, Julia, Fortran, Ada, Prolog, Icon, Elixir, CoffeeScript, Brainfuck, Pypy, Lolcode, Nim, Picolisp, Pike, pypy3

def main(str1, str2):

m = len(str1)

n = len(str2)

j = 0 # Index of str1

i = 0 # Index of str2

# Traverse both str1 and str2

# Compare the current character of str2 with

# first unmatched character of str1

# If matched, then move ahead in str1

while j < m and i < n:

if str1[j] == str2[i]:

j = j+1

i = i + 1

# If all characters of str1 matched,

# then j is equal to m

return j == m

# Driver Program

str2 = str(input())

N = int(input())

for i in range(N):

str1 = str(input())

if main(str1, str2):

print("POSITIVE")

else:

print( "NEGATIVE")