# H. BerOS File Suggestion

time limit per test: 3 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Polycarp is working on a new operating system called BerOS. He asks you to help with implementation of a file suggestion feature.

There are $n$ files on hard drive and their names are $f_1, f_2, \ldots, f_n$. Any file name contains between $1$ and $8$ characters, inclusive. All file names are unique.

The file suggestion feature handles queries, each represented by a string $s$. For each query $s$ it should count number of files containing $s$ as a substring (i.e. some continuous segment of characters in a file name equals $s$) and suggest any such file name.

For example, if file names are "`read.me`", "`hosts`", "`ops`", and "`beros.18`", and the query is "`os`", the number of matched files is $2$ (two file names contain "`os`" as a substring) and suggested file name can be either "`hosts`" or "`beros.18`".

## Input

The first line of the input contains integer $n$ ($1 \le n \le 10000$) — the total number of files.

The following $n$ lines contain file names, one per line. The $i$-th line contains $f_i$ — the name of the $i$-th file. Each file name contains between $1$ and $8$ characters, inclusive. File names contain only lowercase Latin letters, digits and dot characters ('`.`'). Any sequence of valid characters can be a file name (for example, in BerOS "`.`", "`..`" and "`...`" are valid file names). All file names are unique.

The following line contains integer $q$ ($1 \le q \le 50000$) — the total number of queries.

The following $q$ lines contain queries $s_1, s_2, \ldots, s_q$, one per line. Each $s_j$ has length between $1$ and $8$ characters, inclusive. It contains only lowercase Latin letters, digits and dot characters ('`.`').

## Output

Print $q$ lines, one per query. The $j$-th line should contain the response on the $j$-th query — two values $c_j$ and $t_j$, where

- $c_j$ is the number of matched files for the $j$-th query,
- $t_j$ is the name of any file matched by the $j$-th query. If there is no such file, print a single character '−' instead. If there are multiple matched files, print any.

## Example

**input** Copy

```
4
test
contests
test.
.test
6
ts
.
st.
.test
```

```
contes.
st
```

**output** Copy

```
1 contests
2 .test
1 test.
1 .test
0 -
4 test.
```

## → Problem tags

brute force | implementation | *1500

No tag edit access

## → Contest materials

- Announcement
- Tutorial (en)