D. Calendar

BerOilGasDiamondBank has branches in n cities, at that n is an even number. The bank management wants to publish a calendar with the names of all those cities written in two columns: the calendar should consist of exactly $n/2$ lines of strictly equal length, each of which contains exactly two names and exactly one separator character between them. The name of every city should be used in the calendar exactly once. For historical reasons the symbol d is used as the separator of words in the calendar.

The BerOilGasDiamondBank management wants to show that all its branches are equally important to it, that's why the order of their appearance in the calendar should be following: if we "glue"(concatinate) all the $n/2$ calendar lines (from top to bottom) to make a single line, then the lexicographically minimal line is obtained. No separator character will be used to separate calendar lines. For example, if the lines are "bertown!berville", "newberville!bera", then the resulting line is "bertown!bervillenewberville!bera". In some sense one has to find the lexicographically minimal calendar, where the comparison of calendars happens line by line.

Help BerOilGasDiamondBank and construct the required calendar.

Input

The first line contains an integer n ($1 \le n \le 10^4$, n is even) which is the number of branches. Then follow n lines which are the names of the cities. All the names consist of lowercase Latin letters; their lengths are no less than 1 and no more than 10 symbols. The next line contains a single symbol d (d has an ASCII-code from 33 to 126 inclusively, excluding lowercase Latin letters) which is the separator between words in the calendar lines. It is guaranteed that the calendar is possible to be constructed and all the names are different.

Output

Print $n/2$ lines of similar length which are the required calendar. Every line should contain exactly two words and exactly one separator between them. If there are several solutions, print the lexicographically minimal one. The lexicographical comparison of lines is realized by the "<" operator in the modern programming languages.

Examples

input

```
4
b
aa
hg
c
.
```

output

```
aa.b
c.hg
```

input

```
2
aa
a
!
```

output

```
a!aa
```

input

```
2
aa
a
|
```

output

```
aa|a
```