Problem H: Translations

Bob Roberts is in charge of performing translations of documents between various languages. To aid him in this endeavor his bosses have provided him with translation files. These files come in twos — one containing sample phrases in one of the languages and the other containing their translations into the other language. However, some over-zealous underling, attempting to curry favor with the higher-ups with his initiative, decided to alphabetically sort the contents of all of the files, losing the connections between the phrases and their translations. Fortunately, the lists are comprehensive enough that the original translations can be reconstructed from these sorted lists. Bob has found this is most usually the case when the phrases all consist of two words. For example, given the following two lists:

Language 1 Phrases	**Language 2 Phrases**
arlo zym	*bus seat*
flub pleve	*bus stop*
pleve dourm	*hot seat*
pleve zym	*school bus*

Bob is able to determine that *arlo* means *hot*, *zym* means *seat*, *flub* means *school*, *pleve* means *bus*, and *dourm* means *stop*. After doing several of these reconstructions by hand, Bob has decided to automate the process. And if Bob can do it, then so can you.

Input

Input will consist of multiple input sets. Each input set starts with a positive integer n, $n \leq 250$, indicating the number of two-word phrases in each language. This is followed by $2n$ lines, each containing one two-word phrase: the first n lines are an alphabetical list of phrases in the first language, and the remaining n lines are an alphabetical list of their translations into the second language. Only upper and lower case alphabetic characters are used in the words. No input set will involve more than 25 distinct words. No word appears as the first word in more than 10 phrases for any given language; likewise, no word appears as the last word in more than 10 phrases. A line containing a single 0 follows the last problem instance, indicating end of input.

Output

For each input set, output lines of the form

word1/word2

where *word1* is a word in the first language and *word2* is the translation of *word1* into the second language, and a slash separates the two. The output lines should be sorted according to the first language words, and every first language word should occur exactly once. There should be no white space in the output, apart from a single blank line separating the outputs from different input sets. Imitate the format of the sample output, below. There is guaranteed to be a unique correct translation corresponding to each input instance.

Sample Input

```
4
arlo zym
flub pleve
pleve dourm
pleve zym
bus seat
bus stop
hot seat
school bus
2
iv otas
otas re
ec t
eg ec
0
```

Sample Output

```
arlo/hot
dourm/stop
flub/school
pleve/bus
zym/seat

iv/eg
otas/ec
re/t
```