Due date:

- Submit your final Java code and print screen for each question to Lab9 on Learning Hub prior to the indicated due date.
- Please do not zip or compress your submissions. Learning Hub allows you to upload multiple files.

Queston1:

[5 points] Jerry Garcia has a lot of albums, and he keeps them in a stack on the floor. Jerry wants to rearrange his album stack to place his favorite albums nearer to the top. A single operation is available to change the order of the albums in the stack: he can pull an album out of its position in the stack and place it on the top.

Given an original ordering of the album stack and a required ordering for the same stack, your job is to determine a minimal sequence of operations that rearranges the original stack into the required stack.

The first line of the input consists of a single integer *K* giving the number of test cases. Each test case consists of an integer *n* giving the number of albums in the stack. The next *n* lines specify the original ordering of the album stack. Each of the lines contains the name of an album, starting with the album on the top of the stack and working down to the album at the bottom of the stack. Albums have unique names, each of which is a string of no more than eighty characters drawn from a character set consisting of the alphanumeric characters, the space character and the period ('.'). The next *n* lines in the input gives the desired ordering of the stack, once again by naming albums from top to bottom. Each test case consists of exactly 2*n*+1 lines in total. The number of albums (*n*) will be less than or equal to two hundred.

For each test case, the output consists of a sequence of album names, one per line, indicating the order in which albums are to be pulled from their positions in the stack and placed on top. This sequence of operations should transform the original stack into the required stack and should be as short as possible. If more than one solution of shortest length is possible, any of the solutions may be reported. Print a blank line after each test case.

Sample Input

```
2
3
Blues for Allah
Skeletons from the Closet
Terrapin Station
Skeletons from the Closet
Blues for Allah
Terrapin Station
9
Blues for Allah
Skeletons from the Closet
Terrapin Station
Dead Zone
Pizza Tapes
Skull and Roses
Workingmans Dead
In the Dark
Dead Set
Blues for Allah
Skull and Roses
Terrapin Station
Skeletons from the Closet
Dead Zone
Pizza Tapes
Workingmans Dead
In the Dark
Dead Set
```

Sample Output

```
Skeletons from the Closet

Terrapin Station
Skull and Roses
Blues for Allah
```

Question2

[5 points] A common word puzzle found in many newspapers and magazines is the word changing. By taking a starting word and successively altering a single letter to make a new word, one can build a sequence of words which changes the original word to a given end word. For instance, the word ``spice'' can be transformed in four steps to the word ``stock'' according to the following sequence: spice, slice, slick, stick, stock. Each successive word differs from the previous word in only a single character position while the word length remains the same.

Given a dictionary of words from which to make transformations, plus a list of starting and ending words, you should write a program to determine the number of steps in the shortest possible transformation.

Input and Output

The first line of the input is an integer N, indicating the number of test sets that your correct program should test followed by a blank line. Each test set will have two sections. The first section will be the dictionary of available words with one word per line, terminated by a line containing an asterisk (*) rather than a word. There can be up to 200 words in the dictionary; all words will be alphabetic and in lower case, and no word will be longer than ten characters. Words can appear in the dictionary in any order.

Following the dictionary are pairs of words, one pair per line, with the words in the pair separated by a single space. These pairs represent the starting and ending words in a transformation. All pairs are guaranteed to have a transformation using the dictionary given. The starting and ending words will appear in the dictionary.

Two consecutive input set will separated by a blank line.

The output should contain one line per word pair for each test set, and must include the starting word, the ending word, and the number of steps in the shortest possible transformation, separated by single spaces. Two consecutive output set will be separated by a blank line.

Sample Input

1

dip

lip

mad

map

maple

may

pad

pip

pod

pop

sap

sip

slice

slick

spice

stick

stock

*

spice stock

may pod

Sample Output

spice stock 4

may pod 3