

## 630 Anagrams (II)

One of the preferred kinds of entertainment of people living in final stages of XX century is filling in the crosswords. Almost every newspaper and magazine has a column dedicated to entertainment but only amateurs have enough after solving one crossword. Real professionals require more than one crossword for a week. And it is so dull — just crosswords and crosswords — while so many other riddles are waiting out there. For those are special, dedicated magazines. There are also quite a few competitions to take part in, even reaching the level of World Championships. Anyway — a lot.

You were taken on by such a professional for whom riddle solving competing is just a job. He had a brilliant idea to use a computer in work not just to play games. Somehow anagrams found themselves first in the line. You are to write a program which searches for anagrams of given words, using a given vocabulary, tediously filled with new words by yours employer.

### Input

The first line contains  $T$  the number of test cases. There is a blank line after this. Then,  $T$  test cases follow — each separated by a blank line. Each test case has the following structure:

```
< number of words in vocabulary >
< word 1 >
.....
< word N >
< test word 1 >
.....
< test word k >
END
```

$< number of words in vocabulary >$  is an integer number  $N < 1000$ .  $< word 1 >$  up to  $< word N >$  are words from the vocabulary.  $< test word 1 >$  up to  $< test word k >$  are the words to find anagrams for. All words are lowercase (word 'END' means end of data — it is **NOT** a test word). You can assume all words are not longer than 20 characters.

### Output

For each  $< test word >$  — in the order in which it appeared — list the found anagrams in the following way:

```
Anagrams for: < test word >
< No >) < anagram >
.....
```

$< No >$  should be printed on 3 chars.

In case of failing to find any anagrams your output should look like this:

```
Anagrams for: < test word >
No anagrams for: <test word>
```

Print a blank line between datasets.

**Sample Input**

```
1
8
atol
lato
microphotographics
rata
rola
tara
tola
pies
tola
kola
aatr
photomicrographics
END
```

**Sample Output**

```
Anagrams for: tola
  1) atol
  2) lato
  3) tola
Anagrams for: kola
No anagrams for: kola
Anagrams for: aatr
  1) rata
  2) tara
Anagrams for: photomicrographics
  1) microphotographics
```