Problem A: Grading Bins

Filename: *asmall, alarge*
Time Limit: *8 seconds*

Arup teaches many large classes and has to sort exam papers to hand back to his class. He usually places his papers in "bins" by the starting letter of the last name, similar to bucket sort. Unfortunately, since last names are distributed in strange ways, these bins rarely have the same size. For example, there might only be one last name that starts with the letter 'I' but 30 that start with the letter 'M'.

Arup would really like to know if it's possible to have bins labeled in such a way that the number of papers in each bin is equal AND ***there is more than one bin***. This will force him to create a different system for bin placement. Now, instead of each bin being represented by a single letter, each bin can be represented with a ***non-empty*** prefix string. The set of these prefix strings must be such that one is not a prefix of another. For example, consider the following class: ADAMS, ADLER, ATKINS, ATKINSON, BELL, BELLAMY, DAVIS and DIETRICH. The bins could be labeled "AD", "ATKINS", "BEL", and "D", and each bin would contain exactly 2 names. (Also notice that we could have labeled the third bin "B", "BE", or "BELL" instead, but that the last bin must be labeled "D" and the first bin must be labeled "AD".)

The Problem

Given a list of last names (all guaranteed to be unique) in a class, determine how many bins the class could be partitioned into according to the system noted above. Note that it may not be possible to partition the class in any way.

The Input

The first line of input will consist of a single positive integer, c ($c \leq 25$), representing the number of input cases to process. The first line of each input case will contain a single positive integer, n ($2 \leq n \leq 10^5$), representing the number of students in the class. The following n lines will contain the names, already sorted in alphabetical order. All names will contain uppercase letters only, be no longer than 20 letters long, and be distinct names.

Partial Credit Input (40%)

The maximum number of test cases, c will be bounded by $c \leq 10$. The number of names in the input, n will be bounded by $2 \leq n \leq 100$. All of the other bounds will be the same. (The names will be no longer than 20 characters long, will contain uppercase letters only, and be distinct and in sorted order.)

The Output

For each input case, output two lines. The first line should output the number of different bin arrangements, b, that exist. The second line should contain b integers, in sorted order, each followed by a space, representing the number of possible bins. (Notice that there SHOULD BE a space after the last item on a line is listed.) Also note that if $b = 0$, then the second line should simply be a blank line. This is illustrated in the sample on the next page.

Sample Input

```
3
8
ADAMS
ADLER
ATKINS
ATKINSON
BELL
BELLAMY
DAVIS
DIETRICH
2
BELL
BELLAMY
8
ADAMS
ADLER
ATKINS
ATKINSON
BELL
BELLAMY
BLAINE
BLOOMBERG
```

Sample Output

```
1
4
0

2
2 4
```

Note: In the first case, it's impossible to split the data into two bins, since the last four names start with different letters, any bin containing them would break two rules: (1) the bin prefix would be empty, which it's not allowed to be, and (2) even if the empty prefix were allowed, it is itself a prefix of the bin "A", which is also not allowed. It is also not possible to split the data into 8 bins since BELL is a prefix of BELLAMY, so those two can NOT be in separate bins.