There are three computers connected by a network. One of them is a server and the other two are clients. The server has some files, each with a different name. The full name of each file consists of two parts, a name and an extension. Both clients know the full names of all of the files stored on the server. From among its files, the server chooses a single file and sends the name part of that file's full name to one of the clients and the extension part of the full name to the other client.

The clients then begin communicating in an effort to determine which file was selected by the server (they want to learn the file's full name). However, the clients have to communicate in a very restricted manner. Clients take turns sending messages to each other, but they can only say when they don't know the full name of the file. If one client does not know the full name of the chosen file, that client may send a message saying "*I don't know the full file name*" to the other client. The two clients alternate, sending only this message back and forth. This continues until either one of the clients knows the full file name or they decide to quit. The client that received the name part of the full file name always waits for the other client to send the first message.

Pretend that you know all the full file names that reside on the server (both the name and the extension part) and you are listening to the conversation between the clients. Based on this conversation, you should determine the set of files that might have been chosen by the server. Files in this set are called **candidate files**.

Input

The first line is the number of test cases, followed by a blank line.

The first line of each test case in the input file contains two integers N and M, separated by a space. N ($1 \leq N \leq 1000$) is the number of files of the server, and M ($1 \leq M \leq 100$) is the number of messages exchanged between the clients in their effort to determine the full file name.

Each of the next N lines, in each test case, contains one full file name. Full file names are given in a manner similar to MS-DOS 8.3 format. Each full file name is represented in *name.extension* form, where both the name and the extension consist of only capital, alphabetic characters and decimal digits. The name part will have at least one character and at most eight. The extension part will have at most three characters and may be empty. If extension is empty then separating dot may be omitted.

Each full file name appears in each test case no more than once.

Each test case will be separated by a single line.

Output

Write on the first line of the output, for each test case, the number of candidate files based on the given set of files and the number of messages exchanged between the clients. Write zero if there are no candidate files.

On the following lines, for each test case, write the full names of all candidate files. Each of these names should be written on a separate line. They should appear in the same order and with exactly the same spelling as in the input file. That means that if the separating dot was omitted in the input for a particular file then it should also be omitted for this file in the output and vice versa. No file may be listed more than once.

Print a blank line between the outputs for two consecutive test cases.

Sample Input

```
19 2
LICENCE.TMP
WIN32.LOG
FILEID.
PSTOTEXT.TXT
GSVIEW32.EXE
GSVIEW32.ICO
GSVIEWDE.HLP
LICENCE
GSVIEWEN.HLP
GSVW32DE.DLL
FILEID.TMP
GSVW32EN.DLL
PSTOTXT3.DLL
PSTOTXT3.EXE
GSV16SPL.EXE
GVWGS32.EXE
ZLIB32.DLL
PRINTER.INI
README.TXT
```

Sample Output

```
6
LICENCE.TMP
FILEID.
LICENCE
FILEID.TMP
PSTOTXT3.DLL
PSTOTXT3.EXE
```