

Description

Given an m by n grid of letters and a list of words, find the location in the grid at which the word can be found.

A word matches a straight, uninterrupted line of letters in the grid. A word can match the letters in the grid regardless of case (i.e., upper- and lowercase letters are to be treated as the same). The matching can be done in any of the eight horizontal, vertical, or diagonal directions through the grid.

Input

The input begins with a single positive integer on a line by itself indicating the number of cases, followed by a blank line. There is also a blank line between each two consecutive cases.

Each case begins with a pair of integers m followed by n on a single line, where $1 \leq m, n \leq 50$ in decimal notation. The next m lines contain n letters each, representing the grid of letters where the words must be found. The letters in the grid may be in upper- or lowercase. Following the grid of letters, another integer k appears on a line by itself ($1 \leq k \leq 20$). The next k lines of input contain the list of words to search for, one word per line. These words may contain upper- and lowercase letters only - no spaces, hyphens, or other non-alphabetic characters.

Output

For each word in each test case, output a pair of integers representing its location in the corresponding grid. The integers must be separated by a single space. The first integer is the line in the grid where the first letter of the given word can be found (1 represents the topmost line in the grid, and m represents the bottommost line). The second integer is the column in the grid where the first letter of the given word can be found (1 represents the leftmost column in the grid, and n represents the rightmost column in the grid). If a word can be found more than once in the grid, then output the location of the uppermost occurrence of the word (i.e., the occurrence which places the first letter of the word closest to the top of the grid). If two or more words are uppermost, output the leftmost of these occurrences. All words can be found at least once in the grid.

The output of two consecutive cases must be separated by a blank line.

Sample Input

```
1

8 11
abcDEFGhigg
hEbkWalDork
FtyAwaldORm
FtsimrLqsrc
byoArBeDeyv
Klcbqwikomk
strEBGadhrb
yUiqlxcnBjf
4
Waldorf
Bambi
Betty
Dagbert
```

Sample Output

```
2 5
```

2 3
1 2
7 8

Submitting Your Code and Report

Your file name must be "assign1.cpp", and it is the only file you should submit. Please do not submit the executable file of your program. Turn in your project using the "oops submit" command as follows:

```
$ oops submit assign1 assign1.cpp
```

Late submissions will lose 20 points per day.

If you do not follow this submission guideline, you will lose 10 points out of 100.

You should also submit a hard copy of your code to TA. Your report must have a cover page with your student ID and name. In the report, your code must be well commented to explain your algorithm. Also, the sample input and the output of your program must be included in the report.