

Problem GOOGLE: Google

Efficiently searching the web is getting an increasingly important topic. Today, you have to write a simple Google. Given a set of documents (simple strings), your program has to answer queries of the form `string1 string2 string3` by returning a list of documents that match the query along with the number of occurrences of query strings in these documents.

The documents have to be sorted in descending order of query string occurrences.

An Example

We have three documents:

1. "this_is_a_document_that_contains_many_underscores"
2. "_this_is_another_document_which_contains_even_more_underscores_"
3. "thethirddocumentdoesnotcontainanyunderscoresatall"

Working on these documents, your program should return

Document 1: 2 matches

Document 2: 2 matches

Document 3: 2 matches

for the query `underscore document`, since all of the above documents contain one occurrence of "underscore" and one occurrence of "document".

For the query `is_ even_more`, your program should return

Document 2: 3 matches

Document 1: 2 matches

since the third document does not contain any of the query strings, the first document contains two occurrences of the string "is_", and only the second document contains both "is_" (2x) and "even_more" (1x).

Input

The first line of the input contains t , the number of test cases that follow. The first line of each test case consists of two integer numbers, $d < 20$ (the number of documents in this test case) and $q < 200$ (the number of queries to be processed). Each of the next d lines contains a string, the j 'th document of this test case. Each of the remaining q lines consists of one integer $0 < n_i < 20$, the number of strings in the query, followed by n_i strings.

No string in the input will contain a whitespace character. No string will be longer than 10000 characters.

Output

For every query in every test case, print one line for each document that contains at least one occurrence of any string found in the query (so, at most d lines, where d is the number of documents in that test case). Use the format shown in the example above and in the Sample Output. Sort the documents that match the query in descending order of the number of occurrences. If there are several documents with the same number of occurrences, print the one first that appeared earlier in the input (like in the example above). If no document matches the query, print one line: "No document found."

Print a blank line after each query. Please take care of "1 match" and "2 matches".

Sample Input 1

```
2
3 4
this_is_a_document_that_contains_many_underscores
_this_is_another_document_which_contains_even_more_underscores_
thethirddocumentdoesnotcontainanyunderscoresatall
2 underscore document
2 is_ even_more
3 does not contain
2 contain any
1 2
blablabla
3 bla bla blabla
1 blahblah
```

Sample Output 1

Document 1: 2 matches

Document 2: 2 matches

Document 3: 2 matches

Document 2: 3 matches

Document 1: 2 matches

Document 3: 3 matches

Document 2: 2 matches

Document 1: 1 match

Document 1: 2 matches

Document 3: 2 matches

Document 2: 1 match

Document 1: 8 matches

No document found.