Shopee Programming Contest #1

Jun 27, 2020, 02:00 PM CST - Jun 27, 2020, 06:40 PM CST

Search Engine

Max. score: 20

This problem is no longer available for practice. Apology for any inconvenience!

Who doesn't like to search and see these unexpected search suggestions floating just below the search bar. Everyone likes it!!! As we all know Shopee, one of the largest E-commerce platforms, also has a search bar where users can search for all kinds of items. Shopee wants to build a new search engine. And you are to help Shopee to implement this new engine.

You are given a data set that contains all the item's names, and an item's name is represented as an ordered sequence of strings separated by a single space and the strings contain only lowercase English alphabets(**a-z**) and digits(**0-9**). for example, a valid name could be, "**apple iphone se 2**". Queries for the new search engine will be a sequence of alphanumeric strings separated by space. For example, "**se 2**" or "**11 pro max**" and the search engine has to answer how many different items are there in the data set containing the query sequence in their name in exact order. For example, "**se 2**" matches the item "**apple iphone se 2**", however "**app**" doesn't match this item.

Input

Input starts with an integer **T** (**1 ≤ T ≤ 15**), denoting the number of test cases. The first line of each test case will contain two integers **N** (**1 ≤ N ≤ 10^4**) and Q (**1 ≤ Q ≤ 10^4**). Here, **N** is the number of items in the database and **Q** is the total number of queries. Each of the next **N** lines will contain an item's name as described. Each of the next **Q** lines will contain a search query as described. You can safely assume that each item's name will contain at most 10 spaces and the total length will be between 1 to 50.

Output

For each case, print the case number in a single line. Then for each query **Q** print the number of different names in the database who contains the query sequence in their name in exact order.

Constraints

Total number of characters in the dataset will be not more than $7×10^5$

SAMPLE INPUT	SAMPLE OUTPUT
2 3 6 apple lettuce limes avocado onion cranberries apple limes escarole corn28corn apple lettuce limes avocado limes avocado apple lettuce limes apple app apple limes 3 3 apple iphone se 2 iphone 11 max pro iphone 11 pro max apple iphone max pro iphone	Case 1: 2 2 3 3 0 1 Case 2: 1 1 3

Explanation

For the first test case, both "**limes avocado**" and "**apple lettuce**" match both 1st and 3rd items, "**limes**" and "**apple**" match in all three items, "**app**" doesn't match any item and "**apple limes**" matches the second item.

Time Limit:	1.0 sec(s) for each input file.
Memory Limit:	256 MB
Source Limit:	1024 KB
Marking Scheme:	Score is assigned when all the testcases pass.
Allowed Languages:	Bash, C, C++, C++14, C++17, Clojure, C#, D, Erlang, F#, Go, Groovy, Haskell, Java, Java 8, JavaScript(Rhino), JavaScript(Node.js), Julia, Kotlin, Lisp, Lisp (SBCL), Lua, Objective-C, OCaml, Octave, Pascal, Perl, PHP, Python, Python 3, R(RScript), Racket, Ruby, Rust, Scala, Swift-4.1, Swift, TypeScript, Visual Basic

RECENT SUBMISSIONS

DEVELOPERS	RESULT	LANGUAGE
昀翔湯		Python 3.8
Shankar Swaminat...		Python 3.8
Daryl Mok		C++17
昀翔湯		Python 3.8
Team Broccoli		Python 3.8

View All

CODE EDITOR

Save | Python 3 (python 3.5.2)

```python
T = int(input())
dictionary = dict()

for i in range(T):
    print("Case {}:".format(i + 1))
    N, Q = [int(i) for i in input().split()]

    for _ in range(N):
        split_item = input().split()
        unique_subsequence = set()
        for i in range(len(split_item)):
            for j in range(i + 1, len(split_item) + 1):
                subsequence = " ".join(split_item[i:j])
                unique_subsequence.add(subsequence)

        for subsequence in unique_subsequence:
            if subsequence not in dictionary:
                dictionary[subsequence] = 0
            dictionary[subsequence] += 1

    for _ in range(Q):
        query = input()
        if query in dictionary:
            print(dictionary[query])
        else:
            print(0)
```

1:1 vscode

Provide custom input

COMPILE & TEST SUBMIT

Your Rating:

View all comments