

Problem H. LCP Queries

- Time Limit: 2.5 sec

Problem Statement

A string x is called a *prefix* of a string y if x can be obtained by repeating the removal of the last letter of y zero or more times. For example, “abac”, “aba”, “ab”, “a”, and an empty string are the prefixes of “abac”.

For two strings x and y , let $\text{LCP}(x, y)$ be the length of the longest common prefix of x and y . For example, $\text{LCP}(\text{“abacab”}, \text{“abacbba”}) = 4$ because the longest common prefix of these two strings is “abac”. Note that $\text{LCP}(x, y)$ is always defined for any strings x and y because at least an empty string is one of their common prefixes.

You are given n strings s_1, \dots, s_n and m strings t_1, \dots, t_m of lowercase English letters. Then, you are given q queries. In each query you are given an integer sequence a_1, \dots, a_k . Let u be the concatenation of t_{a_1}, \dots, t_{a_k} . Your task is to calculate $\sum_{i=1}^n \text{LCP}(u, s_i)$.

Input

The input consists of a single test case of the following format.

```
n
s1
⋮
sn
m
t1
⋮
tm
q
Query1
⋮
Queryq
```

The first line consists of an integer n . Each of the next n lines consists of a non-empty string s_i of lowercase English letters. The next line consists of an integer m . Each of the next m lines consists of a non-empty string t_j of lowercase English letters.

The next line consists of an integer q . Then q queries are given in order. Each of the queries is given in a single line in the following format.

```
k a1 ⋯ ak
```

k is a positive integer which represents the length of the integer sequence of this query. Each a_i is an integer between 1 and m , inclusive.

You can assume that $1 \leq n \leq 200,000$, $1 \leq m \leq 200,000$ and $1 \leq q \leq 200,000$. The sum of lengths of s_i does not exceed 200,000. Similarly, the sum of lengths of t_i does not exceed 200,000. The sum of k over all queries does not exceed 200,000.

Output

Output q lines. The i -th line should be the answer to the i -th query.

Sample Input

```
5
abcde
aaa
a
ab
bcd
5
a
bc
de
aaaa
b
5
1 1
3 1 2 3
2 2 3
5 5 4 3 2 1
3 3 3 3
```

Sample Output

```
4
9
3
1
0
```