Topic 9: KMP, Automata

Handout

CS 41100 - CP3 Competitive Programming III (Spring 2024)

Purdue University

Instructor: Zhongtang Luo

Date: March 19, 2024

Sample Problem: Tip of Your Tongue
Link: `https://vjudge.net/problem/Kattis-tipofyourtongue`

Sample Problem: Compress Words
Link: `https://vjudge.net/problem/Codeforces-1200E`

Sample Problem: Indie Album
Link: `https://vjudge.net/problem/Codeforces-1207G`

Tip of Your Tongue

Has this scenario ever happened to you? You're having a lovely conversation with someone, begin to say a word you've used hundreds of times, and the word escapes your mind, leaving you sputtering the first syllable. Well worry no more, because the Institute for Cancelling Pauses in Conversation (ICPC) has developed software to curtail this issue.

If there's a word on the tip of your tongue, enter the first few characters of the word into the ICPC software, and it will tell you how many words in the dictionary have those characters as a prefix. The ICPC quickly discovered that many of their clients don't only have tip-of-your-tongue problems, but also base-of-your-tongue problems, where users only remember some suffix of a word.

To be as useful as possible, the ICPC software needs to support the following queries:

1. `AND` *p s*. Count the number of words in the dictionary that have *p* as a prefix and *s* as a suffix.
2. `OR` *p s*. Count the number of words in the dictionary that have *p* as a prefix or *s* as a suffix.
3. `XOR` *p s*. Count the number of words in the dictionary that have *p* as a prefix or *s* as a suffix, but not both.

Prefixes and suffixes of a word may be the whole world, and can overlap within the word. Due to a quirk in the ICPC software, *p* and *s* must be the same length. You must help the software developers answer all the incoming queries.

Input

The first line of input contains two integers n and q ($1 \le n, q \le 2 \cdot 10^5$), where n is the number of words in the dictionary and q is the number of queries.

Each of the next n lines contains a single string w, which is a word in the dictionary. All characters in dictionary words and queries are lowercase Latin characters ('`a`'–'`z`'). All words in the dictionary will be distinct.

Each of the next q lines contains three strings o, p, and s, where o is one of the operations `AND`, `OR`, or `XOR`, and p and s are strings of one or more lowercase Latin characters ('`a`'—'`z`') with $|p| = |s|$.

The total number of characters in the dictionary and across all queries is at most 10^6; this does not include query operations `AND`, `OR`, or `XOR`, whitespace, or newline characters. Said another way, there are at most 10^6 lowercase letters in the input.

Output

Output q lines, one per query, in the order the queries were given. On each line output a single integer, which is the number of words in the dictionary that match that particular query.

Examples

Input

```
4 4
cat
catcat
octal
occidental
AND cat cat
OR oc at
AND ca at
XOR oc al
```

Output

```
2
4
2
0
```

Source

2023 ICPC North America Qualifier 30 September

Compress Words

Amugae has a sentence consisting of n words. He wants to compress this sentence into one word. Amugae doesn't like repetitions, so when he merges two words into one word, he removes the longest prefix of the second word that coincides with a suffix of the first word. For example, he merges "sample" and "please" into "samplease".

Amugae will merge his sentence left to right (i.e., first merge the first two words, then merge the result with the third word and so on). Write a program that prints the compressed word after the merging process ends.

Input

The first line contains an integer n ($1 \le n \le 10^5$), the number of words in Amugae's sentence.

The second line contains n words separated by single space. Each word is non-empty and consists of uppercase and lowercase English letters and digits ('A', 'B', ..., 'Z', 'a', 'b', ..., 'z', '0', '1', ..., '9'). The total length of the words does not exceed 10^6.

Output

In the only line output the compressed word after the merging process ends as described in the problem.

Examples

Input

```
5
I want to order pizza
```

Output

```
Iwantorderpizza
```

Input

```
5
sample please ease in out
```

Output

```
sampleaseinout
```

Indie Album

Mishka's favorite experimental indie band has recently dropped a new album! Songs of that album share one gimmick. Each name s_i is one of the following types:

- 1 c — a single lowercase Latin letter;
- 2 j c — name s_j ($1 \le j < i$) with a single lowercase Latin letter appended to its end.

Songs are numbered from 1 to n. It's guaranteed that the first song is always of type 1.

Vova is rather interested in the new album but he really doesn't have the time to listen to it entirely. Thus he asks Mishka some questions about it to determine if some song is worth listening to. Questions have the following format:

- i t — count the number of occurrences of string t in s_i (the name of the i-th song of the album) as a continuous substring, t consists only of lowercase Latin letters.

Mishka doesn't question the purpose of that information, yet he struggles to provide it. Can you please help Mishka answer all Vova's questions?

Input

The first line contains a single integer n ($1 \le n \le 4 \cdot 10^5$) — the number of songs in the album.

Each of the next n lines contains the description of the i-th song of the album in the following format:

- 1 c — s_i is a single lowercase Latin letter;
- 2 j c — s_i is the name s_j ($1 \le j < i$) with a single lowercase Latin letter appended to its end.

The next line contains a single integer m ($1 \le m \le 4 \cdot 10^5$) — the number of Vova's questions.

Each of the next m lines contains the description of the j-th Vova's question in the following format:

- i t ($1 \le i \le n$, $1 \le |t| \le 4 \cdot 10^5$) — count the number of occurrences of string t in s_i (the name of the i-th song of the album) as a continuous substring, t consists only of lowercase Latin letters.

It's guaranteed that the total length of question strings t doesn't exceed $4 \cdot 10^5$.

Output

For each question print a single integer — the number of occurrences of the question string t in the name of the i-th song of the album as a continuous substring.

Examples

Input

```
20
1 d
2 1 a
2 2 d
2 3 a
2 4 d
2 5 a
2 6 d
2 7 a
1 d
2 9 o
2 10 k
2 11 i
2 12 d
2 13 o
2 14 k
2 15 i
```

```
2 1 o
2 17 k
2 18 i
2 15 i
12
8 da
8 dada
8 ada
6 dada
3 dada
19 doki
19 ok
16 doki
15 doki
9 d
1 a
20 doki
```

Output

```
4
3
3
2
0
1
1
2
1
1
0
2
```