#include <bits/stdc++.h>

using namespace std;

const int MAXN = 26;

struct Trie{

bool words;

int prefixes;

Trie\* edges[26];

Trie(){

words = false;

prefixes = 0;

for(int i = 0;i < MAXN; i++)

edges[i] = NULL;

}

};

Trie\* root;

void addWord(Trie\* vertex, int idx, string &word){

if(idx == word.size()){

vertex->prefixes++;

vertex->words= true;

return;

}

vertex->prefixes++;

if(vertex->edges[word[idx]-'a'] == NULL)

vertex->edges[word[idx]-'a'] = new Trie;

addWord(vertex->edges[word[idx]-'a'], idx+1, word);

}

int countPrefixes(Trie\* vertex, int idx, string &word){

if(idx == word.size())

{

return vertex->prefixes;

}

if(vertex->edges[word[idx]-'a'] == NULL)

return 0;

return countPrefixes(vertex->edges[word[idx]-'a'], idx+1, word);

}

int main(){

root = new Trie();

int n, q;

cin >> n >> q;

for(int i = 0;i < n; i++){

string s;

cin >> s;

addWord(root, 0, s);

}

while(q--){

string s;

cin >> s;

cout << countPrefixes(root, 0, s) << endl;

}

return 0;

}