

---

**Search in ordered list****X93037\_en**

---

Write an efficient program to search on an ordered list of strings. Strings in the list are ordered as follows: first criterion is the string length, shorter strings appear before. When two strings have the same length they are sorted according to the usual string order (the lexicographic order).

**Exam score:** 2.500000 **Automatic part:** 0.000000%

**Input**

An integer  $n$  greater than zero followed by a list of  $n$  different strings sorted according to the criteria described above. Afterwards, there is a string sequence of strings without order at all.

**Output**

For each string in the sequence, a line with the position of the string in the ordered list. If the string is not in the list, the output line is -1.

**Sample input**

```
14
jk ng pi ana max noe alex jose luis olga
ng
jk
joan
pere
pedro
alfonso
julia
max
```

**Sample output**

```
1
0pere marta pedro alfonso
-1
10
12
13
-1
4
```

**Observation**

Fill in the following code without changing any line of code already provided.

```
// some additional functions may be necessary here

//pre: v is ordered according to string length first and then
//      by usual string order. All strings are different.
//post: returns the position of s in v.
//      If s is not in v, returns -1
int effi_search(const vector<string>& v, const string& s) {
    //
    // some lines of code are needed here
    //
}
```

```
int main() {  
    int n;  
    cin >> n;  
    vector<string> v(n);  
    for (int i = 0; i < n; ++i) cin >> v[i];  
    string s;  
    while (cin >> s)  
        cout << effi_search(v, s) << endl;  
}
```

### **Problem information**

Author : Pro1

Generation : 2020-01-07 16:56:50

© *Jutge.org*, 2006–2020.

<https://jutge.org>