# Sparse Arrays



There is a collection of input strings and a collection of query strings. For each query string, determine how many times it occurs in the list of input strings. Return an array of the results.

**Example**

$strings = [' ab',' ab', abc']$

$queries = [' ab',' abc',' bc']$

There are $2$ instances of '$ab$', $1$ of '$abc$' and $0$ of '$bc$'. For each query, add an element to the return array,

$results = [2, 1, 0]$.

**Function Description**

Complete the function matchingStrings in the editor below. The function must return an array of integers representing the frequency of occurrence of each query string in strings.

matchingStrings has the following parameters:

- string strings[n] - an array of strings to search
- string queries[q] - an array of query strings

**Returns**

- int[q]: an array of results for each query

**Input Format**

The first line contains and integer $n$, the size of $strings[]$.

Each of the next $n$ lines contains a string $strings[i]$.

The next line contains $q$, the size of $queries[]$.

Each of the next $q$ lines contains a string $queries[i]$.

**Constraints**

$1 \leq n \leq 1000$

$1 \leq q \leq 1000$

$1 \leq |strings[i]|, |queries[i]| \leq 20$ .

**Sample Input 1**

Copy Download

```
4
aba
baba
aba
xzxb
3
aba
xzxb
ab
```

**Sample Output 1**

```
2
1
0
```

**Sample Input 2**

Copy Download

```
3
def
de
```

Author ikbalkazar

Difficulty Medium

Max Score 100

Submitted By 24821

NEED HELP?

View discussions

View editorial

View top submissions

RATE THIS CHALLENGE

MORE DETAILS

Download problem statement

Download sample test cases

Suggest Edits

```
fgh
3
de
lmn
fgh
```

**Sample Output 2**

```
1
0
1
```

**Sample Input 3**

Copy Download

```
13
abcde
sdaklfj
asdjf
na
basdn
sdaklfj
asdjf
na
asdjf
na
basdn
sdaklfj
asdjf
5
abcde
sdaklfj
asdjf
na
basdn
```

**Sample Output 3**

```
1
3
4
3
2
```

Change Theme Language C#

```
using System.CodeDom.Compiler;
using System.Collections.Generic;
using System.Collections;
using System.ComponentModel;
using System.Diagnostics.CodeAnalysis;
using System.Globalization;
using System.IO;
using System.Linq;
using System.Reflection;
using System.Runtime.Serialization;
using System.Text.RegularExpressions;
using System.Text;
using System;

class Result
{

    /*
```

Line: 68 Col: 1

Upload Code as File

Test against custom input

Run Code

Submit Code