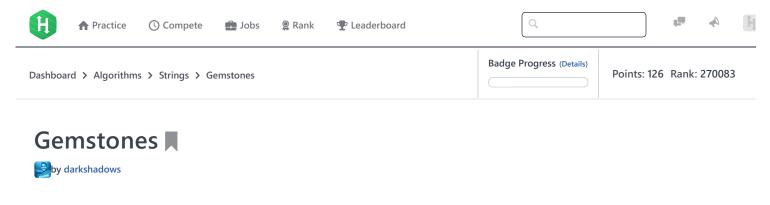
15/11/2017 HackerRank



John has discovered various rocks. Each rock is composed of various elements, and each element is represented by a lower-case Latin letter from 'a' to 'z'. An element can be present multiple times in a rock. An element is called a *gem-element* if it occurs at least once in each of the rocks.

Discussions

Editorial A

**Topics** 

Given the list of N rocks with their compositions, display the number of gem-elements that exist in those rocks.

Leaderboard

#### **Input Format**

Problem

The first line consists of an integer, N, the number of rocks.

Submissions

Each of the next N lines contains a rock's composition. Each composition consists of lower-case letters of English alphabet.

#### **Constraints**

 $1 \le N \le 100$ 

Each composition consists of only lower-case Latin letters ('a'-'z').

 $1 \le length \ of \ each \ composition \le 100$ 

## **Output Format**

Print the number of gem-elements that are common in these rocks. If there are none, print 0.

### **Sample Input**

3 abcdde baccd eeabg

# **Sample Output**

2

### **Explanation**

Only "a" and "b" are the two kinds of gem-elements, since these are the only characters that occur in every rock's composition.

F in
Submissions:62190
Max Score:20
Difficulty: Easy
Rate This Challenge:
なかなかな

15/11/2017 HackerRank

Alphabets Dictionary

More

```
Current Buffer (saved locally, editable) & • •
                                                                                            Java 7
                                                                                                                              Ö
 1 ▼ import java.io.*;
 2 import java.util.*;
 3 import java.text.*;
    import java.math.*;
 5
    import java.util.regex.*;
 6
 7 ▼ public class Solution {
 8
 9 ₹
         static int gemstones(String[] arr){
10
             // Complete this function
11
12
13 ▼
         public static void main(String[] args) {
             Scanner in = new Scanner(System.in);
14
             int n = in.nextInt();
15
             String[] arr = new String[n];
16 ▼
             for(int arr_i = 0; arr_i < n; arr_i++){</pre>
17 ▼
                 arr[arr_i] = in.next();
18 ▼
19
20
             int result = gemstones(arr);
21
             System.out.println(result);
22
         }
23
     }
24
                                                                                                                     Line: 1 Col: 1
                                                                                                         Run Code
                                                                                                                      Submit Code
1 Upload Code as File
                      Test against custom input
```

Join us on IRC at #hackerrank on freenode for hugs or bugs.

Contest Calendar | Blog | Scoring | Environment | FAQ | About Us | Support | Careers | Terms Of Service | Privacy Policy | Request a Feature