



链滴

[每日 LeetCode] 1002. Find Common Characters

作者: [Hanseltu](#)

原文链接: <https://ld246.com/article/1554304405229>

来源网站: 链滴

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

Description:

Given an array **A** of strings made only from lowercase letters, return a list of all characters that show up in all strings within the list (**including duplicates**) . For example, if a character occur 3 times in all strings but not 4 times, you need to include that character three times in the final answer.

You may return the answer in any order.

Example 1:

Input: ["bella","label","roller"]

Output: ["e","l","l"]

Example 2:

Input: ["cool","lock","cook"]

Output: ["c","o"]

Note:

1. $1 \leq A.length \leq 100$
2. $1 \leq A[i].length \leq 100$
3. $A[i][j]$ is a lowercase letter

思路：本文求二维字符串数组中每组字符中的重复字符。考虑维护一个长度为26的数组，对每个串进行扫描，保证每次存储的是已经比较过的公共字符。其中 **count_min** 数组来保存每个字符在其所属字符串中出现的最小次数， **count** 数组来保存每个字符在每一种字符串中出现的最小次数。

C++代码

```
class Solution
{
public:
    vector<string> commonChars(vector<string> &A)
    {
        vector<string> res;
        vector<int> count(26, INT_MAX);
        for (auto a : A)
        {
            vector<int> count_min(26, 0);
            for (auto c : a)
                ++count_min[c - 'a'];
            for (int i = 0; i < 26; i++)
                count[i] = min(count_min[i], count[i]);
        }
        for (int i = 0; i < 26; i++)
        {
            for (int j = 0; j < count[i]; j++)
```

```
    {  
        res.push_back(string(1, i + 'a'));  
    }  
}  
return res;  
}
```

运行时间: 12ms

运行内存: 10M