# 题目

给你一个下标从 0 开始的字符串数组 words 和一个字符 x 。

请你返回一个 下标数组 ，表示下标在数组中对应的单词包含字符 x 。

注意 ，返回的数组可以是 任意 顺序。

示例 1：

输入：words = ["leet","code"], x = "e"

输出：[0,1]

解释："e" 在两个单词中都出现了："leet" 和 "code" 。所以我们返回下标 0 和 1 。

示例 2：

输入：words = ["abc","bcd","aaaa","cbc"], x = "a"

输出：[0,2]

解释："a" 在 "abc" 和 "aaaa" 中出现了，所以我们返回下标 0 和 2 。

示例 3：

输入：words = ["abc","bcd","aaaa","cbc"], x = "z"

输出：[]

解释："z" 没有在任何单词中出现。所以我们返回空数组。

提示：

1 <= words.length <= 50

1 <= words[i].length <= 50

x 是一个小写英文字母。

words[i] 只包含小写英文字母。

# 分析

## 方法一：一次遍历

思路：

遍历words，判断x是否在words[i] 中，如果是则把i加入答案。

代码：

class Solution {

public:

vector<int> findWordsContaining(vector<string> &words, char x) {

vector<int> ans;

for (int i = 0; i < words.size(); i++) {

auto &s = words[i];

if (find(s.begin(), s.end(), x) != s.end()) {

ans.push\_back(i);

}

}

return ans;

}

};

复杂度分析

时间复杂度：O(L)，其中L为所有字符串的长度之和。

空间复杂度：O(1)。