884. Uncommon Words from Two Sentences

Solved ⊘

`Easy` | 🏷 Topics | 🔒 Companies

A **sentence** is a string of single-space separated words where each word consists only of lowercase letters.

A word is **uncommon** if it appears exactly once in one of the sentences, and **does not appear** in the other sentence.

Given two **sentences** `s1` and `s2`, return *a list of all the **uncommon words***. You may return the answer in **any order**.

Example 1:

```
Input: s1 = "this apple is sweet", s2 = "this apple is sour"
Output: ["sweet","sour"]
```

Example 2:

```
Input: s1 = "apple apple", s2 = "banana"
Output: ["banana"]
```

Constraints:

← All Submissions

</> Code

C# ⌄ 🔒 Auto

Accepted
👤 Zhukovilya submitted at Jun 29, 2024 17:34

📖 Editorial | ✏ Solution

⏱ Runtime	🖽 Memory
110 ms | Beats **18.06%**	46.71 MB | Beats **79.17%** ✓

✦ Analyze Complexity

10%

5%

0%

82ms   91ms   97ms   102ms   107ms   115ms

1.39% of solutions used 110 ms of runtime

82ms 91ms 97ms 102ms 107ms 115ms

Code | C#

```
using System;
using System.Collections.Generic;

public class Solution {
```

```
1  using System;
2  using System.Collections.Generic;
3
4  public class Solution {
5      public string[] UncommonFromSentences(string s1, string s2) {
6          Dictionary<string, int> wordCount = new Dictionary<string, int>();
7
8          // Обрабатываем первую строку
9          foreach (var word in s1.Split(' ')) {
10             if (wordCount.ContainsKey(word)) {
11                 wordCount[word]++;
```

Saved   Ln 38, Col 1

☑ Testcase | >_ Test Result

Accepted Runtime: 106 ms

• Case 1   • Case 2

Input

s1 =
"this apple is sweet"

s2 =
"this apple is sour"

Код:

```csharp
using System;
using System.Collections.Generic;

public class Solution
{
    public string[] UncommonFromSentences(string s1, string s2)
    {
        Dictionary<string, int> wordCount = new Dictionary<string, int>();

        // Обрабатываем первую строку
```

```csharp
        foreach (var word in s1.Split(' '))
        {
            if (wordCount.ContainsKey(word))
            {
                wordCount[word]++;
            }
            else
            {
                wordCount[word] = 1;
            }
        }

        // Обрабатываем вторую строку
        foreach (var word in s2.Split(' '))
        {
            if (wordCount.ContainsKey(word))
            {
                wordCount[word]++;
            }
            else
            {
                wordCount[word] = 1;
            }
        }

        // Находим редкие слова
        List<string> uncommonWords = new List<string>();
        foreach (var entry in wordCount)
        {
            if (entry.Value == 1)
            {
                uncommonWords.Add(entry.Key);
            }
        }

        return uncommonWords.ToArray();
    }
}
```