

Problem Challenge 2

We'll cover the following ^

- String Anagrams (hard)
- Try it yourself

String Anagrams (hard)

Given a string and a pattern, find **all anagrams of the pattern in the given string**.

Anagram is actually a **Permutation** of a string. For example, "abc" has the following six anagrams:

1. abc
2. acb
3. bac
4. bca
5. cab
6. cba

Write a function to return a list of starting indices of the anagrams of the pattern in the given string.

Example 1:


```
Input: String="ppqp", Pattern="pq"
Output: [1, 2]
Explanation: The two anagrams of the pattern in the given string are "pq" and "qp".
```


Example 2:


```
Input: String="abbcabc", Pattern="abc"
Output: [2, 3, 4]
Explanation: The three anagrams of the pattern in the given string are "bca", "cab", and "abc".
```


Try it yourself

Try solving this question here:

 Java

 Python3

 JS


 C++

```
1 import java.util.*;
2
3 class StringAnagrams {
4     public static List<Integer> findStringAnagrams(String str, String pattern) {
5         List<Integer> resultIndices = new ArrayList<Integer>();
6         // TODO: Write your code here
7         return resultIndices;
8     }
9 }
10
```

Test

Save

Reset





← Back

Next →

Solution Review: Problem Challenge 1

Solution Review: Problem Challenge 2

✓ Completed

 Report an Issue  Ask a Question