

Problem Challenge 2

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 using namespace std;
2
3 #include <iostream>
4 #include <string>
5 #include <unordered_map>
6 #include <vector>
7
8 class StringAnagrams {
9 public:
10 static vector<int> findStringAnagrams(const string &str, const string &pattern) {
11     vector<int> resultIndices;
12     // TODO: Write your code here
13     return resultIndices;
14 }
15 };
16
```

Save

↩

⌂

Show Results

Show Console

×

0 of 2 Tests Passed

Result	Input	Expected Output	Actual Output	Reason
✗	findStringAnagrams(ppqp, pq)	[1, 2]	[]	Incorrect Output
✗	findStringAnagrams(abbcab, abc)	[2, 3, 4]	[]	Incorrect Output

3.888s