Given two strings s and p, return *an array of all the start indices of* p*'s anagrams in* s. You may return the answer in **any order**.

An **Anagram** is a word or phrase formed by rearranging the letters of a different word or phrase, typically using all the original letters exactly once.

**Example 1:**

Input: s = "cbaebabacd", p = "abc"  
Output: [0,6]  
Explanation:  
The substring with start index = 0 is "cba", which is an anagram of "abc".  
The substring with start index = 6 is "bac", which is an anagram of "abc".

**Example 2:**

Input: s = "abab", p = "ab"  
Output: [0,1,2]  
Explanation:  
The substring with start index = 0 is "ab", which is an anagram of "ab".  
The substring with start index = 1 is "ba", which is an anagram of "ab".  
The substring with start index = 2 is "ab", which is an anagram of "ab".

**Constraints:**

* 1 <= s.length, p.length <= 3 \* 104
* s and p consist of lowercase English letters.