**[Find All Anagrams in a String](https://leetcode.com/problems/find-all-anagrams-in-a-string/)**

**package** leetcodeMedium;

**import** java.util.ArrayList;

**import** java.util.List;

**public** **class** FindAnagramsInString {

**public** **static** **void** main(String[] args) {

// **TODO** Auto-generated method stub

System.***out***.println(*findAnagrams*("aa" , "bb"));

}

**public** **static** List<Integer> findAnagrams(String s, String p) {

List<Integer> result = **new** ArrayList<>();

**if**(s.length() < p.length()) {

**return** result;

}

**int**[] text = **new** **int**[26];

**int**[]pattern = **new** **int**[26];

**for**(**int** i = 0 ; i < p.length(); i++) {

text[s.charAt(i) - 'a']++;

pattern[s.charAt(i) - 'a']++;

}

**int** start = 0;

**int** end = p.length();

**while**(end < s.length()) {

**if**(*isAnagram*(pattern , text)) {

result.add(start);

}

text[s.charAt(start) - 'a']--;

start++;

text[s.charAt(end) - 'a']++;

end++;

}

**if**(*isAnagram*(pattern, text)) {

result.add(start);

}

**return** result;

}

**public** **static** **boolean** isAnagram(**int**[] pattern , **int**[] text) {

**for**(**int** i = 0 ; i < 26 ; i++) {

**if**(pattern[i] != text[i]) {

**return** **false**;

}

}

**return** **true**;

}

}

Time complexity : O(t + p) t is length of text and p is length of pattern

Space Complexity : O(1) constant space