**Anagram**

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.

1. **Problem solving:**

**Length Check:**

Check if the lengths of the two input strings s and t are equal. If not, return false since they cannot be anagrams.

**Initialize Character Count Array:**

Create an array (charCount) to store the count of each character. Initialize the array with zeros, assuming ASCII characters.

**Count Characters in String s:**

Iterate through each character in string s, convert the character to its ASCII code, and increment the corresponding count in the charCount array.

**Decrement Character Count in String t:**

Iterate through each character in string t, convert the character to its ASCII code, and decrement the corresponding count in the charCount array.

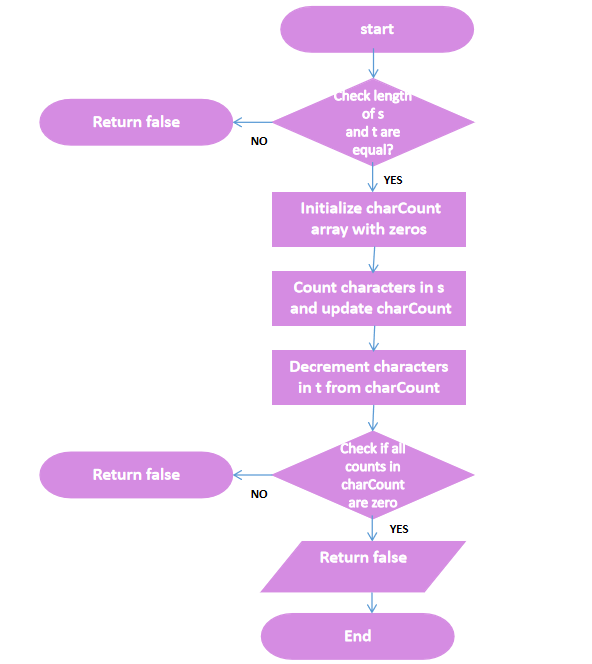
**Check if All Counts are Zero:**

Iterate through the charCount array. If any count is not zero, it means there is a character mismatch between the two strings. In this case, return false.

**Return true if All Checks Passed:**

If all checks passed (i.e., the lengths are equal, and the character counts are zero for all characters), return true. This means the input strings are anagrams.

1. **Flow chart**



**End**

1. **Output** 