You are given two strings s and t. In one step, you can append **any character** to either s or t.

Return *the minimum number of steps to make*s*and*t***anagrams****of each other.*

An **anagram** of a string is a string that contains the same characters with a different (or the same) ordering.

**Example 1:**

**Input:** s = "**lee**tco**de**", t = "co**a**t**s**"

**Output:** 7

**Explanation:**

- In 2 steps, we can append the letters in "as" onto s = "leetcode", forming s = "leetcode**as**".

- In 5 steps, we can append the letters in "leede" onto t = "coats", forming t = "coats**leede**".

"leetcodeas" and "coatsleede" are now anagrams of each other.

We used a total of 2 + 5 = 7 steps.

It can be shown that there is no way to make them anagrams of each other with less than 7 steps.

**Example 2:**

**Input:** s = "night", t = "thing"

**Output:** 0

**Explanation:** The given strings are already anagrams of each other. Thus, we do not need any further steps.

**Constraints:**

* 1 <= s.length, t.length <= 2 \* 105
* s and t consist of lowercase English letters.