Given two strings s and t, find the number of ways you can choose a non-empty substring of s and replace a **single character** by a different character such that the resulting substring is a substring of t. In other words, find the number of substrings in s that differ from some substring in t by **exactly** one character.

For example, the underlined substrings in "computer" and "computation" only differ by the 'e'/'a', so this is a valid way.

Return *the number of substrings that satisfy the condition above.*

A **substring** is a contiguous sequence of characters within a string.

**Example 1:**

**Input:** s = "aba", t = "baba"

**Output:** 6

**Explanation:** The following are the pairs of substrings from s and t that differ by exactly 1 character:

("aba", "baba")

("aba", "baba")

("aba", "baba")

("aba", "baba")

("aba", "baba")

("aba", "baba")

The underlined portions are the substrings that are chosen from s and t.

​​**Example 2:**

**Input:** s = "ab", t = "bb"

**Output:** 3

**Explanation:** The following are the pairs of substrings from s and t that differ by 1 character:

("ab", "bb")

("ab", "bb")

("ab", "bb")

​​​​The underlined portions are the substrings that are chosen from s and t.

**Example 3:**

**Input:** s = "a", t = "a"

**Output:** 0

**Example 4:**

**Input:** s = "abe", t = "bbc"

**Output:** 10

**Constraints:**

* 1 <= s.length, t.length <= 100
* s and t consist of lowercase English letters only.