Given an array of strings patterns and a string word, return *the****number****of strings in*patterns*that exist as a****substring****in*word.

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

**Example 1:**

**Input:** patterns = ["a","abc","bc","d"], word = "abc"

**Output:** 3

**Explanation:**

- "a" appears as a substring in "abc".

- "abc" appears as a substring in "abc".

- "bc" appears as a substring in "abc".

- "d" does not appear as a substring in "abc".

3 of the strings in patterns appear as a substring in word.

**Example 2:**

**Input:** patterns = ["a","b","c"], word = "aaaaabbbbb"

**Output:** 2

**Explanation:**

- "a" appears as a substring in "aaaaabbbbb".

- "b" appears as a substring in "aaaaabbbbb".

- "c" does not appear as a substring in "aaaaabbbbb".

2 of the strings in patterns appear as a substring in word.

**Example 3:**

**Input:** patterns = ["a","a","a"], word = "ab"

**Output:** 3

**Explanation:** Each of the patterns appears as a substring in word "ab".

**Constraints:**

* 1 <= patterns.length <= 100
* 1 <= patterns[i].length <= 100
* 1 <= word.length <= 100
* patterns[i] and word consist of lowercase English letters.