We are given that the string "abc" is valid.

From any valid string V, we may split V into two pieces X and Y such that X + Y (X concatenated with Y) is equal to V.  (X or Y may be empty.)  Then, X + "abc" + Y is also valid.

If for example S = "abc", then examples of valid strings are: "abc", "aabcbc", "abcabc", "abcabcababcc".  Examples of **invalid** strings are: "abccba", "ab", "cababc", "bac".

Return true if and only if the given string S is valid.

**Example 1:**

**Input:** "aabcbc"

**Output:** true

**Explanation:**

We start with the valid string "abc".

Then we can insert another "abc" between "a" and "bc", resulting in "a" + "abc" + "bc" which is "aabcbc".

**Example 2:**

**Input:** "abcabcababcc"

**Output:** true

**Explanation:**

"abcabcabc" is valid after consecutive insertings of "abc".

Then we can insert "abc" before the last letter, resulting in "abcabcab" + "abc" + "c" which is "abcabcababcc".

**Example 3:**

**Input:** "abccba"

**Output:** false

**Example 4:**

**Input:** "cababc"

**Output:** false

**Note:**

1. 1 <= S.length <= 20000
2. S[i] is 'a', 'b', or 'c'