A **value-equal** string is a string where **all** characters are the same.

* For example, "1111" and "33" are value-equal strings.
* In contrast, "123" is not a value-equal string.

Given a digit string s, decompose the string into some number of **consecutive value-equal** substrings where **exactly one** substring has a **length of**2 and the remaining substrings have a **length of**3.

Return true*if you can decompose*s*according to the above rules. Otherwise, return*false.

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

**Example 1:**

**Input:** s = "000111000"

**Output:** false

**Explanation:** s cannot be decomposed according to the rules because ["000", "111", "000"] does not have a substring of length 2.

**Example 2:**

**Input:** s = "00011111222"

**Output:** true

**Explanation:** s can be decomposed into ["000", "111", "11", "222"].

**Example 3:**

**Input:** s = "011100022233"

**Output:** false

**Explanation:** s cannot be decomposed according to the rules because of the first '0'.

**Constraints:**

* 1 <= s.length <= 1000
* s consists of only digits '0' through '9'.