# 291. Word Pattern II

## Description

Given a `pattern` and a string `s`, return `true` *if* `s` ***matches*** *the* `pattern`.

A string `s` **matches** a `pattern` if there is some **bijective mapping** of single characters to **non-empty** strings such that if each character in `pattern` is replaced by the string it maps to, then the resulting string is `s`. A **bijective mapping** means that no two characters map to the same string, and no character maps to two different strings.

**Example 1:**

```
Input: pattern = "abab", s = "redblueredblue"
Output: true
Explanation: One possible mapping is as follows:
'a' -> "red"
'b' -> "blue"
```

**Example 2:**

```
Input: pattern = "aaaa", s = "asdasdasdasd"
Output: true
Explanation: One possible mapping is as follows:
'a' -> "asd"
```

**Example 3:**

```
Input: pattern = "aabb", s = "xyzabcxzyabc"
Output: false
```

**Constraints:**

- `1 <= pattern.length, s.length <= 20`
- `pattern` and `s` consist of only lowercase English letters.