Description Solution Discuss (288) Submissions

# 291. Word Pattern II

Medium 799 63 Add to List Share

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 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.

Accepted 65,556 Submissions 139,646

```
class Solutic
public:
    bool isPc
&pattern,stri
&s,unordered_
&mp,int i,int
j,unordered_s
    {
        if(i
pattern.size(
s.size())
            r

        if((i
pattern.size(
s.size()) ||
pattern.size(
s.size()) )
            r

 if(mp.find(p
!=mp.end() )
        {
            s
mp[pattern[i]
            i

 while(k<curr
j<s.size())
            {

 if(curr[k]!=

false;
```

Testcase Run Code Resu

Accepted Runtime

Your input "ab" "aa"

Output false

Expected false

Problems Pick One Prev /149 Next xample ases Run Code