Problem 3407: Substring Matching Pattern

Problem Information

Difficulty: Easy
Acceptance Rate: 27.80%
Paid Only: No
Tags: String, String Matching

Problem Description

You are given a string `s` and a pattern string `p`, where `p` contains **exactly one** `'*'` character.

The `'*'` in `p` can be replaced with any sequence of zero or more characters.

Return `true` if `p` can be made a substring of `s`, and `false` otherwise.

Example 1:

Input: s = "leetcode", p = "ee*e"

Output: true

Explanation:

By replacing the `'*'` with `"tcod"`, the substring `"eetcode"` matches the pattern.

Example 2:

Input: s = "car", p = "c*v"

Output: false

Explanation:

There is no substring matching the pattern.

Example 3:

Input: s = "luck", p = "u*"

Output: true

Explanation:

The substrings `"u"`, `"uc"`, and `"uck"` match the pattern.

Constraints:

* `1 <= s.length <= 50` * `1 <= p.length <= 50 ` * `s` contains only lowercase English letters. * `p` contains only lowercase English letters and exactly one `'*'`

Code Snippets

C++:

```
class Solution {
public:
bool hasMatch(string s, string p) {

}
};
```

Java:

```
class Solution {
public boolean hasMatch(String s, String p) {

}
}
```

Python3:

```
class Solution:
def hasMatch(self, s: str, p: str) -> bool:
```