Problem 392: Is Subsequence

Problem Information

Difficulty: Easy
Acceptance Rate: 48.68%
Paid Only: No
Tags: Two Pointers, String, Dynamic Programming

Problem Description

Given two strings `s` and `t`, return `true` _if_`s` _is a**subsequence** of _`t` _, or_`false` _otherwise_.

A **subsequence** of a string is a new string that is formed from the original string by deleting some (can be none) of the characters without disturbing the relative positions of the remaining characters. (i.e., `"ace"` is a subsequence of `"_a_ b _c_ d _e_ "` while `"aec"` is not).

Example 1:

Input: s = "abc", t = "ahbgdc" **Output:** true

Example 2:

Input: s = "axc", t = "ahbgdc" **Output:** false

Constraints:

* `0 <= s.length <= 100` * `0 <= t.length <= 104` * `s` and `t` consist only of lowercase English letters.

Follow up: Suppose there are lots of incoming `s`, say `s1, s2, ..., sk` where `k >= 109`, and you want to check one by one to see if `t` has its subsequence. In this scenario, how would you change your code?

Code Snippets

C++:

```cpp
class Solution {
public:
bool isSubsequence(string s, string t) {

}
};
```

Java:

```java
class Solution {
public boolean isSubsequence(String s, String t) {

}
}
```

Python3:

```python
class Solution:
def isSubsequence(self, s: str, t: str) -> bool:
```