392. Is Subsequence

Tags	
Property	@August 18, 2022

Question

原文：

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 `"abcde"` while `"aec"` is not).

我的理解：

假如字串s被字串t包含return true，否則return false，但包含必須順序也相同，舉例：`"ace"`被`"abcde"`包含但`"aec"`不被包含

翻譯：

给定两个字符串`s`和`t`，如果`s`是`t`的**子序列**，则返回`true`，否则*`false`。

一个字符串的**子序列**是一个新的字符串，它是通过删除一些（可以是没有）字符而形成的，不会影响到其余字符的相对位置。(例如，"ace "是 "abcde "的一个子序列，而 "aec "不是）。

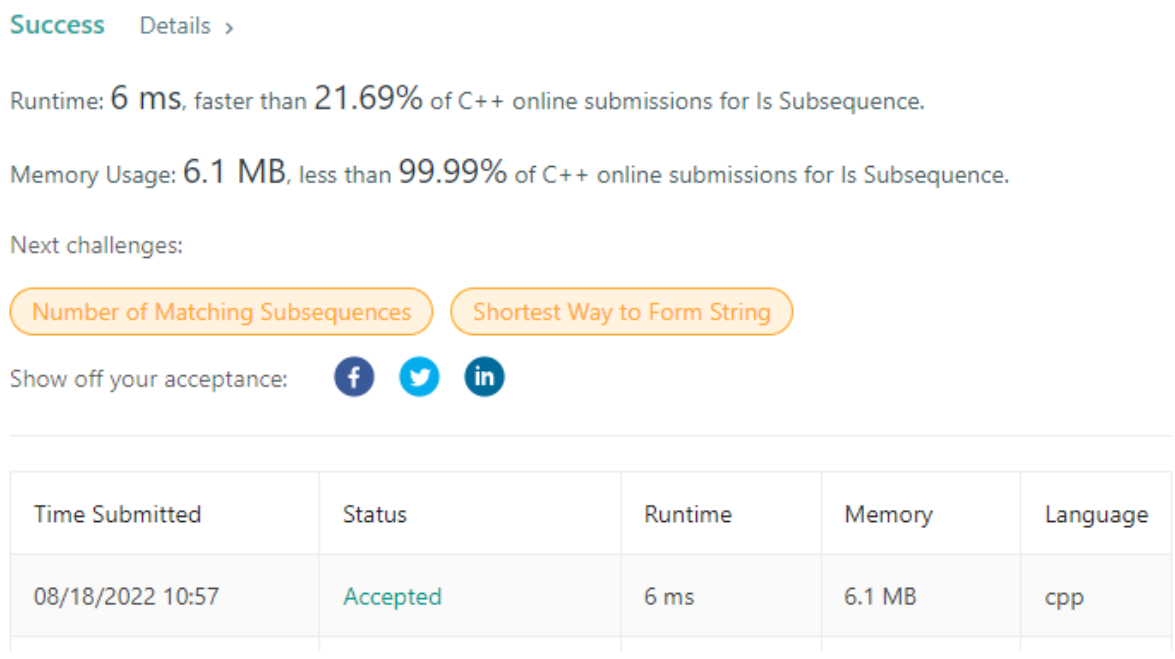
自評翻譯正確性：99%

- Word Memory：
  - disturbing干擾

Code

```
class Solution {
public:
    bool isSubsequence(string s, string t) {
        int i=0,j;
        if(s=="")
            return true;
        for(j=0;j<t.size();j++){
            if(s[i]==t[j]){
                i++;
            }
            if(i==s.size()){
                return true;
            }
        }
        return false;
    }
};
```

> 思路：先拿第一位s的字符去跟t做依序比對，如果有相同的字符s t同時增進一位，直到依序搜索完t如果s也被搜索完代表t包含s return true，否則return false



優良code參考

> 思路：