

## 75 Days of Code

### Day11 Problem no: 392

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).

Example 1:

Input: s = "abc", t = "ahbgdc"

Output: true

Example 2:

Input: s = "axc", t = "ahbgdc"

Output: false

For this problem ,

1. Taking two pointers as pointer1 , pointer2 and loop the array while pointer2 < array's length .
2. Increase pointer1 only when s has element that exist in t due to the pointer it will always be in a sequence
3. Compare the length of s with pointer2 , if pointer2 exceed means it has a subsequence else return false

```
18
19 function isSubsequence(s: string, t: string): boolean {
20     let pointer1 = 0;
21     let pointer2 = 0;
22     while (pointer2 < t.length) {
23         if (s[pointer1] === t[pointer2]) {
24             pointer1++;
25         }
26         pointer2++;
27     }
28     if (pointer1 >= s.length) {
29         return true;
30     }
31     return false;
32 }
33
34 let ans = isSubsequence("abc", "abcdefgh");
35 console.log(ans)
36
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL OUTPUTS

[Running] node "c:\Users\Shubham\Desktop\75daysOfCode\75DaysOfCode\day11.js"  
true