

DP - Is Subsequence

learning curve:

#2. Trying with DP table =

i \ j	a	h	b	g	d	c
a	T	T	T	T	T	T
b	F	F	T	T	T	T
c	F	F	F	F	F	T

This approach is using pointers

#1.

$S = "abc"$
 $t = "ahbgdc"$

for i in range of $s.length$
 for j in range $(i, t.length)$

count = 0

found = false;

true

if $j = i$ and $(j > indexfound) =$

$indexfound = j$

~~found = false~~ found = true

count + 1

count is being calculated / incremented only if found in the condition

set to false for every new character inside our string s

found = false;

if Count == $s.length$

return true;

else,

return false;

If the count is equal to the length of string s then we will return true

if I had "aahbeef" with $s = "abc"$

First one would have found the second one would also be found true and it will label $indexfound = 3$
 Third one would not be found because the index of c should be $>$ than $indexfound$ but it is 2 instead so according to the question requirement, it will have a correct output.

* In python, boolean is done with True and False.

Problem found is that the code is not good at dealing with repeated occurrence of a character

for i in range $s.length$

for j in $(indexfound, t.length)$

if $(s[i] == t[j])$
 $indexfound = j + 1$

↳ ensures that I don't walk back

after going through the coding process: I have realized that it might not be extremely necessary to write the boolean found values. We could just use the count value to kind of figure out if what we need will occur or not. The reason for that is count will only increment whenever the 2 conditions are met.