Problem #917: Reverse Only Letters (Easy)

------------------------------------------------

<https://leetcode.com/problems/reverse-only-letters/>

My Solution:

1. Convert the string to a list called alist.

2. Let n be the length of the list.

3. Initialize two pointers i and j to 0 and n-1 respectively.

4. Iterate through alist while i is less than j.

If the characters at index i and j are both letters, then switch them and advanced to the next letter, i to the right and j to the left.

If one of the characters at index i and j or at both are not letters, then advance the one that is not a letter -- i to the right and j to the left.

5. Join the items in alist to form a string and return the string.

-----------------------------------------------------------------------------------------------

class Solution:

def reverseOnlyLetters(self, S: str) -> str:

alist = list(S)

n = len(alist)

i = 0

j = n -1

while (i < j):

if alist[i].isalpha() and alist[j].isalpha():

alist[i], alist[j] = alist[j], alist[i]

i += 1

j -= 1

elif alist[i].isalpha() and not alist[j].isalpha():

j -= 1

elif not alist[i].isalpha() and alist[j].isalpha():

i += 1

else: # both alist[i] and alist[j] are not letters

i += 1

j -=1

return(''.join(alist))

============================

Alternate Solution is to use a stack and a list to hold the result.

def reverseOnlyLetters(self, s: str) -> str:

stack = []

for char in s:

if char.isalpha():

stack.append(char)

res = []

for char in s:

if char.isalpha():

res.append(stack.pop())

else:

res.append(char)

return ''.join(res)