Problem #345: Reverse Vowels in a String. (Easy)

<https://leetcode.com/problems/reverse-vowels-of-a-string/>

Given a string s, reverse only all the vowels in the string and return it.

The vowels are 'a', 'e', 'i', 'o', and 'u', and they can appear in both lower and upper cases, more than once.

**Example 1:**

**Input:** s = "hello"

**Output:** "holle"

**Example 2:**

**Input:** s = "leetcode"

**Output:** "leotcede"

**Constraints:**

* 1 <= s.length <= 3 \* 105
* s consist of **printable ASCII** characters.

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

**My Solution:**

1. Let i be the pointer at the beginning of the string and j be that from the end.
2. Create a list called alist from string s.

I and j are traveling towards each other and while they do not cross each other,

if the character at i pointer and j pointer are vowels, then swap them. Otherwise, keep advancing i and j towards each other.

class Solution:

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

vowels = {'a', 'e', 'i', 'o', 'u', 'A', 'E', 'I', 'O', 'U'}

i = 0

j = len(s) - 1

alist = list(s)

while (i < j):

if alist[i] not in vowels:

i += 1

if alist[j] not in vowels:

j -= 1

if i < j and alist[i] in vowels and alist[j] in vowels:

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

i += 1

j -= 1

#print("i = ", i, "j = ", j, "alist = ", alist)

return "".join(alist)