# URLify: Write a method to replace all spaces in a string with '%20'. You may assume that the string has sufficient space at the end to hold the additional characters, and that you are given the "true" length of the string. (Note: If implementing in Java, please use a character array so that you can perform this operation in place.)

# so Id ask if i have to make the swap from space to url space in place? YES

# double space has to be compressed? No, it should be two %20

#Ok so first I was thinking of each space I encounter move the rest of the string to the back so im going to do a quick implementations of that, but that wont be efficient

#IDK what means the true length of the str i would suppose that is excluding the extra space at the end so an example would be like

# 'asd asd ', 7 so two extra spaces for the extra %20 i need to add

def urilify(str1, str\_len): # O(n^2)

max\_len = len(str1)

if max\_len == str\_len:

return str1

for ith in range(max\_len):

if ith == str\_len:

return str1[:max\_len]

if str1[ith]== " ":

str1=str1[:ith]+"%20"+str1[ith+1:] #if all the string is made of spaces\*\*

ith+=2

Return str1[:max\_len+1]

##if all the string is spaces then assuming the string is N i have to move n chars N times

#means that n^2

# im ignoring that they are givin me the extra space at the end because of some reason

def urilify\_2(str1,str\_len):#O(n)

str1 = invert(str1)#this is O(n)

ith = 0

jth = len(str1) - str\_len

while jth < len(str1):#this is O(n)

if str1[jth]==" ":

str1[ith]="0"

str1[ith+1]="2"

str1[ith+2]="%"

ith+=2

else:

str1[ith]=str1[jth]

jth+=1

ith+=1

return ''.join(invert(str1))#this is O(2n)

def invert(str1):#inverting a list is O(n)

str\_len = len(str1)

str1 = list(str1)#this is O(n)

for ith in range(int(str\_len/2)):#this is O(n/2)

tempo=str1[ith]#space O(n)

str1[ith]= str1[str\_len-ith-1]

str1[str\_len-ith-1]=tempo

return str1

print(urilify\_2("asdasd", 6))

print(urilify\_2("asd asd ", 8))

print(urilify\_2("asd a sd ", 9))

#so I could improve this solution and as soon as I start inverting the string start checking if its a space and swap with %20 and as soon as i finish i just revert it back