Day -3 DSA

Question- on string 😀

Que1:anagram question check the whether two string are anagram or not.

link=<https://leetcode.com/problems/valid-anagram/description/>

def anagram(s ,t):

hash\_T ,hash\_S={},{}

if(len(s)!=len(t)):

return False

for i in range(len(s)):

hash\_S[s[i]]=1+hash\_S.get(s[i],0)

hash\_T[t[i]]=1+hash\_T.get(t[i],0)

for c in hash\_S:

if(hash\_S[c]) !=hash\_T.get(c,0):

return False

return True

s="anagram"

t="gramana"

print(anagram(s,t))

Que2:

Que:check the where the sentence is palimdrome or not ➕

link=<https://leetcode.com/problems/valid-palindrome/description/>

ans=

def check\_sent(str):

check = "".join([char.lower() for char in s if char.isalnum()])

return check == check[::-1]

print(check\_sent("the given sent"))

Que3:

Que:the longest substring without repeating character:

link=<https://leetcode.com/problems/longest-substring-without-repeating-characters/description/>

def check\_str(s):

char\_set=set()

l=0

res=0

for r in range(len(s)):

while s[r] in char\_set:

char\_set.remove(s[l])

l+=1

char\_set.add(s[r])

res=max(res , r-l+1)

return res

string="aabbcc"

print(check\_str(string))

Que4:missing number

def missingNumber(self, nums: List[int]) -> int:

n = len(nums)

v = [-1] \* (n + 1)

for num in nums:

v[num] = num

for i in range(len(v)):

if v[i] == -1:

return i

return 0