* s="mubeena"

vowels="aeiouAEIOU"

v=0

c=0

for i in s:

if(i in vowels):

v=v+1

else:

c=c+1

print(v,c)

* def ValidPassword(s):

if(len(s)<4):

return 0

elif(" " in s or "/" in s):

return 0

elif(s[0].isdigit()):

return 0

caps=0

nums=0

for i in s:

if(i.isdigit()):

num+=1

elif(i.isupper()):

caps+=1

if(num>0 and caps>0):

* class Solution:

    def isValid(self, word: str) -> bool:

        if(len(word)<3):

            return False

        vowel="aeiouAEIOU"

        v=0

        c=0

        n=0

        for i in word:

            if(i.isalpha()):

                if(i in vowel):

                    v+=1

                else:

                    c+=1

            elif(i.isdigit()):

                    n+=1

            else:

                    return False

        if(v>0 and c>0):

                return True

        return False

* l=[2,3,4,5,5,1]

Max=l[0]

for i in range(0,len(1)):

if(l[i]>Max):

Max=l[i]

Smax=float(" -inf")

for i in range(0,len(l)):

if(l[i]>Smax and l[i]!=max):

Smax=l[i]

print(Smax)

----l=[0,1,0,3,12]

k=[0]\*len(l)

x=0

for i in l:

if(i!=0):

k[x]=i

x+=1

print(k)

---class Solution:

    def isValid(self, s: str) -> bool:

        stack=[]

        for i in s:

            if(i=="{" or i=="[" or i=="("):

                stack.append(i)

            else:

                if(len(stack)==0):

                    return False

                x=stack.pop()

                if(x=="(" and i==")" or x=="[" and i=="]" or x=="{" and i=="}"):

                    continue

                else:

                    return False

        return len(stack)==0

-----import sys

class Solution:

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

        cs=0

        ms=float("-inf")

        for i in arr:

            cs+=i

            ms=max(cs,ms)

            if(cs<0):

                cs=0

        return ms

* class Solution:

    def spiralOrder(self, mat: List[List[int]]) -> List[int]:

        sr=0

        er=len(mat)-1

        sc=0

        ec=len(mat[0])-1

        l=[]

        while(sr<=er and sc<=ec):

            for i in range(sc,ec+1):

                l.append(mat[sr][i])

            sr+=1

            for i in range(sr,er+1):

                l.append(mat[i][ec])

            ec-=1

            if(sr<=er):

                for i in range(ec,sc-1,-1):

                    l.append(mat[er][i])

                er-=1

            if(sc<=ec):

                for i in range(er,sr-1,-1):

                    l.append(mat[i][sc])

                sc+=1

        return l