Python programming-CSA0814

DAY 7

(13 aug 24)

**1.FIND THE FREQUENCY OF GIVEN ELEMENTS IN A LIST**

lst=[1,2,3,4,5,4,3,2,1]

c=list.count(3)

print(c)

**OUTPUT:**

2

**2.MATRIX MULTIPLICATION**

Import numpy as n

a=n.array([[1,2,3],

[2,3,4],

[5,6,7]])

b=n.array([[1,2,3],

[4,5,6],

[7,8,9]])

c=n.dot(a,b)

print(c)  
**OUTPUT:**

**[[ 30 36 42]**

**[ 42 51 60]**

**[ 78 96 114]]**

**3.FINDING ISOMORPHIC OR NOT**

def isomorphic(str1, str2):

if len(str1) != len(str2):

return "not isomorphic"

else:

map1, map2 = {}, {}

for i in range(len(str1)):

ch1, ch2 = str1[i], str2[i]

if ch1 not in map1:

map1[ch1] = ch2

if ch2 not in map2:

map2[ch2] = ch1

if map1[ch1] != ch2 or map2[ch2] != ch1:

return "not isomorphic"

return "isomorphic"

str1 = 'abacba'

str2 = 'xpxcpx'

print(isomorphic(str1, str2))

**OUTPUT:**

Isomorphic

**4.PERFORM THE FOLLOWING STRING OPERATION:TREE,HOME=THROEMEE**

s1="abcd"

s2="123456"

new=" "

length=max(len(s1),len(s2))

for i in range(length):

if i<len(s1):

new+=s1[i]

if i<len(s2):

new+=s2[i]

print(new)

**OUTPUT:**

a1b2c3d456

**5.FINDING STARTING AND ENDING INDEX OF TARGET**

l1 = [1, 2, 3, 4, 5, 4, 3, 2, 1]

t = 2

if t in l1:

start = l1.index(t)

end = len(l1)-1- l1[::-1].index(t)

print("First occurrence of", t, "is at index:", start)

print("Last occurrence of", t, "is at index:", end)

else:

print("Element not found")

**OUTPUT:**

First occurrence of 2 is at index: 1

Last occurrence of 2 is at index: 7

**6.PRINT THE NUMBER OF UPPER,LOWER AND NUMBERS UNTIL \* COMES.**

upper, lower, num = 0, 0, 0

while True:

s = input("Enter a string (or space to exit): ")

if s == ' ':

break

upper += sum(1 for c in s if c.isupper())

lower += sum(1 for c in s if c.islower())

num += sum(1 for c in s if c.isdigit())

print(f"Uppercase: {upper}, Lowercase: {lower}, Numbers: {num}")

**OUTPUT:**

Enter a string (or space to exit): dfj

Uppercase: 0, Lowercase: 3, Numbers: 0

**7.FINDING THE STRING IS VALID OR NOT**

a=input("enter the string:")

if a.isalnum():

print("valid")

else:

print("invalid")

**OUTPUT:**

enter the string:123start

valid

**8.BINARY ADDITION**

def binaryaddition(a,b):

return bin(int(a,2)+int(b,2))[2:]

result=binaryaddition('1010','1101')

print(result)

**OUTPUT:**

10111

**9.FINDING HAYSTACK OCCURANCE**

def findsubstr(haystack:str,needle:str):

return haystack.find(needle)

result = findsubstr("Hello, world!", "world")

print(result)

**OUTPUT:**

7

**10.COUNTING LEXICOGRAPHYCALLY SORTED VOWEL**

def count(n:int)->int:

return (n+4)+(n+3)+(n+2)+(n+1)//24

print(count(2))

**OUTPUT:**

15