**Shri Vaishnav Institute of Information Technology**

**Department of Computer Science Engineering Program-B.Tech(TCS)**

**ASSIGNMENT-4**

**Q.1 A website requires the users to input username and password to register. Write a program to check the validity of password input by users.**

**Following are the criteria for checking the password:**

**1. At least 1 letter between [a-z]**

**2. At least 1 number between [0-9]**

**1. At least 1 letter between [A-Z]**

**3. At least 1 character from [$#@]**

**4. Minimum length of transaction password: 6**

**5. Maximum length of transaction password: 12**

**Your program should accept a sequence of comma separated passwords and will check them according to the above criteria. Passwords that match the criteria are to be printed, each separated by a comma.'''**

**Solution**

import re

s = []

input = (n for n in input("Enter password: ").split(','))

for password in input:

if not re.search('[a-z]', password):

pass

elif not re.search('[0-9]', password):

pass

elif not re.search('[A-Z]', password):

pass

elif not re.search('[$#@]', password):

pass

elif (len(password) < 6):

pass

elif (len(password) > 12):

pass

else:

s.append(password)

print("The correct password is:")

print(",".join(s)).

**Q.2 You are required to write a program to sort the (name, age, height) tuples by ascending order where name is string, age and height are numbers. The tuples are input by console. The sort criteria is:**

**1: Sort based on name;**

**2: Then sort based on age;**

**3: Then sort by score.**

**The priority is that name > age > score.**

**Solution:-**

s = 'Tom,19,80 John,20,90 Jony,17,91 Jony,17,93 Json,21,85'

lst = [tuple(x.split(',')) for x in s.split()]

print(sorted(lst, key=lambda x: (x[0], x[1], x[2])))

**Q.3 Write a program to compute the frequency of the words from the input. The output should output after sorting the key alphanumerically.**

def word\_count(str):

counts = dict()

words = str.split()

for word in words:

if word in counts:

counts[word] += 1

else:

counts[word] = 1

return counts

print( word\_count('New to Python or choosing between Python 2 and Python 3? Read Python 2 or Python 3..'))

**Q.4 Write a program which accepts a string from console and print the characters that have even indexes.**

a=input("Enter the string :")

list(a)

print(a[::2])

**Q.5 With a given list [12,24,35,24,88,120,155,88,120,155], write a program to print this list after removing all duplicate values with original order reserved.**

def Remove(duplicate):

final\_list = []

for num in duplicate:

if num not in final\_list:

final\_list.append(num)

return final\_list

duplicate = [12,24,35,24,88,120,155,88,120,155]

print(Remove(duplicate))

**Q6 Given a string find a substring based on the following conditions:**

**The substring must be the longest one of all the possible substring in the given string .**

**There must not be any repeating characters in the substring .If there is more than one substring satisfying the above two conditions then print the substring which occurs first if there is no substring then print -1.**

def areDistinct(strr, i, j):

visited = [0] \* (26)

for k in range(i, j + 1):

if (visited[ord(strr[k]) -

ord('a')] == True):

return -1

visited[ord(strr[k]) -

ord('a')] = True

return True

def longestUniqueSubsttr(strr):

n = len(strr)

res = 0

for i in range(n):

for j in range(i, n):

if (areDistinct(strr, i, j)):

res = max(res, j - i + 1)

return res

if \_\_name\_\_ == '\_\_main\_\_':

strr = "thedogisthewise"

print("The input is ", strr)

len = longestUniqueSubsttr(strr)

print("The length of the longest "

"non-repeating character substring is ", len).

THANK YOU!