PRESIDENCY UNIVERSITY

COURSE TITLE & CODE: Programming in Python & CSE 317

SEMESTER/YEAR: VI/III

COURSE CREDIT STRUCTURE: 1- 0- 4-3

Name : Sai Ram.K

ID : 20181CSE0621

Exercise - 4

Strings

Question 1

1. Write a program that accepts a sentence and calculate the number of upper case letters and lower case letters. Suppose the following input is supplied to the program: Hello world! Then, the output should be: UPPER CASE 1 LOWER CASE 9

```
string = input("Enter a sentence : ")
up=0 ; low = 0
for letter in string:
  if (letter.isupper()):
    up+=1
  else :
    low+=1
print("Uppercase letters = ",up,"Lowercase letters = ",low)
```

```
Enter a sentence : ABcdEF
Uppercase letters =  4 Lowercase letters =  2
```

Question 2

2. Write a program that accepts a sentence and calculate the number of letters and digits. Suppose the following input is supplied to the program: Hello world! 123 Then, the output should be: LETTERS 10 DIGITS 3

```
string = input("Enter a sentence : ")
let=0 ; dig=0
for letter in string:
  if (letter.isdigit()):
    dig+=1
  else : let+=1
print("Letters = ",let," Digits = ",dig)
```

```
Enter a sentence : Mobile number is 9876543210
Letters =  17  Digits =  10
```

Question 3

3. Arrange String characters such that lowercase letters should come first. Given input String of combination of the lower and upper case arrange characters in such a way that all lowercase letters should come first. Expected Output:
   Input String: PyNaTive
   Output_String: yaivePNT

```
string = input("Enter a string : ")
up=[] ; low=[]
for letter in string:
  if (letter.isupper()):
    up.append(letter)
  else :
    low.append(letter)
final = ''.join(low + up)
print(final)
```

```
Enter a string : AnTIBioTicS
nioicATIBTS
```

Question 4

4. Write a Python program to count repeated characters in a string.

```
 Sample string: 'thequickbrownfoxjumpsoverthelazydog'
Expected output:
 o 4
```

```
e 3
u 2
h 2
r 2
t 2
```

```
import collections
string = input("Enter the string : ")
count = collections.defaultdict(int)
for c in string:
    count[c] += 1

for c in sorted(count, key=count.get, reverse=True):
  if count[c] > 1:
     print('%s %d' % (c, count[c]))
```

```
Enter the string : An apple a day keeps a doctor away
  7
a 6
p 3
e 3
d 2
y 2
o 2
```

Question 5

5. Write a program to find the first occurrence of word in a string. Input
   Input string: I love programming!
   Input word to search: love
   Output
   'love' is found at index 2.

```
string = input("Enter a string : ")
word = input('Enter search word : ')
print(f"{word} is found at index ",string.find(word))
```

```
Enter a string : I love programming!
Enter search word : love
love is found at index  2
```

Question 6

6. Write a program to count frequency of each character in a string using loop. How to find frequency of each characters in a string in C programming. Logic to count frequency of each character in a given string in C program. Input string: Codeforwin Output

Frequency of all characters in the given string:
'c' = 1
'd' = 1
'e' = 1
'f' = 1
'i' = 1
'n' = 1
'o' = 2
'r' = 1
'w' = 1

```
count = {}
string = input("Enter the string : ")
for i in string:
  if i in count:
    count[i]+=1
  else :
    count[i]=1
#print(count.items())
for i in count:
    print (i,'=',count[i])
```

```
Enter the string : anappleadaykeepsadoctoraway
a = 7
n = 1
p = 3
l = 1
e = 3
d = 2
y = 2
k = 1
s = 1
o = 2
c = 1
t = 1
r = 1
w = 1
```

Question 7

7. Write a program to read any string from user and remove last occurrence of a given character from the string.
   Input string: I love programming. I love Codeforwin.
   Input character to remove: I
   Output
   String after removing last 'I': I love programming. love Codeforwin.

```
string = input("Enter a string : ")
rem = input("Enter character to be removed : ")
x = string.replace(rem,'')
x
```

```
Enter a string : I love programming. I love Codeforwin.
Enter character to be removed : love
'I  programming. I  Codeforwin.'
```

Exercise - 5

Lists

Question 1

1. Create an empty list and read the BMI of ‘n’ people. Check each value and display the category as shown.

BMI	Category
0-18.5	Underweight
18.5 to 23	Normal Range
23-25	Overweight

1. Read the list of BMI
2. Compare each BMI, as per the BMI and category, display the list of BMI in each category

INPUT and OUTPUT
[16.5, 17, 20, 21, 25]
Underweight [16.5, 17]
Normal Range [20, 21]
Overweight [25]

```
lst =[float(lst) for lst in input(f"Enter BMI of 'n' persons : ").split(',')]
uw =[] ; nr=[] ; ow=[]
for bmi in lst:
  if (bmi>=0 and bmi<=18.5):
    uw.append(bmi)
  elif (bmi>18.5 and bmi<23):
    nr.append(bmi)
  elif (bmi>=23 and bmi<=25):
    ow.append(bmi)
print(f"Underweight = {uw} \n Normal Range = {nr} \n Overweight= {ow}" )
```

```
Enter BMI of 'n' persons : 16.5,17,20,21,25
Underweight = [16.5, 17.0]
 Normal Range = [20.0, 21.0]
 Overweight= [25.0]
```

Question 2

2. Given a list of numbers, write a Python program to count Even and Odd numbers in a List.

Input: list1 = [2, 7, 5, 64, 14] Output: Even = 3, odd = 2

Input: list2 = [12, 14, 95, 3] Output: Even = 2, odd = 2

```
lst = [int(lst) for lst in input("enter n integers : ").split(',')]
even=0; odd=0
for num in lst:
  if (num%2==0):
    even+=1
  else : odd+=1
print("even = ",even,"odd = ",odd)
```

```
enter n integers : 12,14,95,3
even =  2 odd =  2
```

Question 3

3. Write a program to print index at which particular value exists. If the value exists at multiple location in the list, then print all the indices. Also count the number of times that value is repeated in the list. INPUT:
   L1 0, 1,2,4,5,5,6,4
   Enter the value to be searched: 4
   OUTPUT:
   4 found at location 3
   4 found at location 7
   4 appears 2 times in the list

```
lst = [int(lst) for lst in input('Enter the list of integers : ').split(',')]
key = int(input('Enter the value to be searched for : '))
res = [i for i in range(len(lst)) if lst[i] == key]
for i in res:
  print(f"{key} found at location {i}")
print(f'{key} appears {len(res)} times in the list')
```

```
Enter the list of integers : 1,2,4,6,5,4,7
Enter the value to be searched for : 4
4 found at location 2
4 found at location 5
4 appears 2 times in the list
```

4. Create an empty list to insert ‘n’ string values given by the user. Later, print all the elements of the list starting from some specific item from the list mentioned by the user. If the specified item is not found in the list, then print all the elements of the list. OUTPUT: Test Case 1: L= ['bus', 'car', 'train', 'flight', 'boat']
   Enter the list item to print from: car
   ['car', 'train', 'flight', 'boat']
   Test Case 2:
   Enter the list item to print from: bike
   bike Not Found
   ['bus',”car”, 'train', 'flight', 'boat']

```
lst = [lst for lst in input("Enter strings : ").split(',')]
key = input("Enter item to print from : ")
if key in lst:
  x = lst.index(key)
  print(lst[x:])
else:
  print("Not Found")
```

```
Enter strings : car,train,bus,flight,boat
Enter item to print from : bus
['bus', 'flight', 'boat']
```

Question 5

5. Write a program that computes the net amount of a bank account based a transaction log from console input (read the list of transaction from the users and compute the net amount). The transaction log format is shown as following: D 100
   W 200

D means deposit while W means withdrawal.
Suppose the following input is supplied to the program:
D 300
D 300
W 200
D 100
Then, the output should be:
500

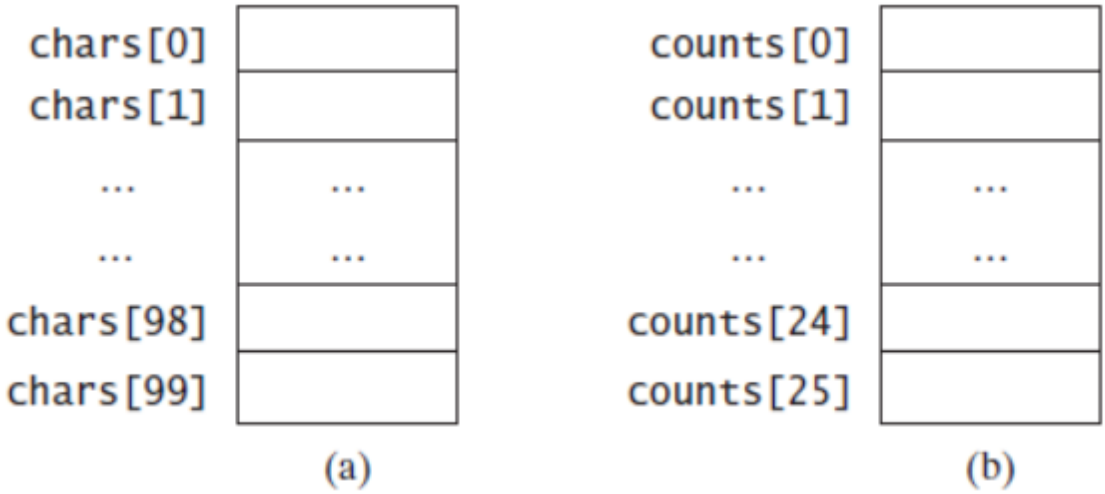
```
amt = [ amt for amt in input("Enter complete transaction details (D amt/ W amt) : ").split
```

```
amt = [ amt for amt in input( Enter complete transaction details (D amt/ W amt) :  ).split
amt = sorted(amt)
dep=[];wit=[]
for x in amt :
  if x[0]=='D':
    dep.append(x)
  else :
    wit.append(x)
dsum = 0 ; wsum = 0
for i in dep :
  i=i.split(' ')
  dsum = int(i[1]) + dsum
for j in wit :
  j=j.split(' ')
  wsum = int(j[1]) + wsum
print("Net amount = ",(int(dsum) - int(wsum)))
```

```
Enter complete transaction details (D amt/ W amt) : D 300,D 300,W 200,D 100
Net amount =  500
```

Question 6

6. Generates 100 lowercase letters randomly and assigns them to a list of characters, Named chars, as shown figure (a).



The chars list stores 100 characters, and the counts list stores 26 counts shown in the figure (b), each of which counts the occurrences of a letter. Write a python program to achieve the above operation.

Sample Input and Output:

The lowercase letters are:
e y l s r i b k j v j h a b z n w b t v
s c c k r d w a m p w v u n q a m p l o
a z g d e g f i n d x m z o u l o z j v

h w i w n t g x w c d o t x h y v z y z
q e a m f w p g u q t r e n n w f c r f

The occurrences of each letter are:
5 a 3 b 4 c 4 d 4 e 4 f 4 g 3 h 3 i 3 j
2 k 3 l 4 m 6 n 4 o 3 p 3 q 4 r 2 s 4 t

```
import random as rd
chars=[]
for i in range(100):
  x = rd.randint(97,122)
  chars.append(chr(x))
print("lower case letters are : ")
#for x in range(len(chars)):
print(chars)
import collections
counter=collections.Counter(chars)
print("Count of occurences is ")
print(counter)
```

```
lower case letters are :
['e', 't', 'f', 'n', 'o', 'g', 'm', 'e', 'b', 'k', 'f', 'j', 'e', 's', 't', 'u', 'a',
Count of occurences is
Counter({'n': 8, 'e': 7, 'l': 6, 'd': 6, 'm': 5, 'j': 5, 's': 5, 'u': 5, 't': 4, 'f'
```

Question 7

7. CIT College need the library automation application which should perform the following functionalities.

   I. The following details must be there for each Book
   i. bookTitle
   ii. bookAuthor
   iii. bookNoOfCopies
   iv. bookAvailability
   v. bookEdition
   vi. bookPublisher
   II. Add the Book details into library automation application
   III. Update the Book details in library automation application
   IV. Display the Book list with all details in a proper and neat format.
   V. Display the list of books of Ashok Nandev author

```
lst = input("Enter title,author,no.of copies,availability,edition & publisher : ").split('
lib =[]
for book in lst:
  lib.append(book)
```

```
for i in range(len(lib)):
  print("Title \t\t | ",lib[0]) ; print("author \t\t | ",lib[1]) ; print("no.of copies \t\
  print("edition  \t\t | ",lib[4]) ; print("publisher \t\t | ",lib[5])
  break
```

```
Enter title,author,no.of copies,availability,edition & publisher : HarryPotter JK.Row
Title            |  HarryPotter
author           |  JK.Rowling
no.of copies             |  2
availability             |  Yes
edition                  |  edition.12.2
publisher                |  Marcos
```

Question 8

8. Given a list of integers with duplicate elements in it. The task to generate another list, which contains only the duplicate elements. In simple words, the new list should contain the elements which appear more than one.

Input: list = [10, 20, 30, 20, 20, 30, 40, 50, -20, 60, 60, -20, -20]

Output: output_list = [20, 30, -20, 60]

Input: list = [-1, 1, -1, 8]

Output: output_list = [-1]

```
lst = [lst for lst in input("Enter values : ").split(',')]
dup=[]
for i in range(0, len(lst)):
    for j in range(i+1, len(lst)):
        if(lst[i] == lst[j]):
            print(lst[i])
```

```
Enter values : 10,20,30,20,20,30,40,50,-20,60,60,-20,-20
20
20
30
20
-20
-20
60
-20
```

Exercise - 6

Dictionaries

Question 1

1. Write python program to read name of the day and the corresponding temperature of days in a week and store it in dictionary. Find the temperature for any day of the week using a dictionary Input and Output:
   Enter the day: SUN

Enter the temperature: 31
Enter the day: MON
Enter the temperature: 27
Enter the day: TUE
Enter the temperature: 26
Enter the day: WED
Enter the temperature: 31
Enter the day: THUR
Enter the temperature: 25
Enter the day: FRI
Enter the temperature: 27
Enter the day: SAT
Enter the temperature: 31

Enter a day - SUN, MON, TUE, WED, THUR, FRI or SAT:
SUN
The temperature 31.0

```
days = [days for days in input("Enter days : ").split(',')]
temp = [temp for temp in input(f"Enter temperature : ").split(',')]
d = dict(zip(days,temp))
find = input("Enter day to know temp : ")
print(d[find],'^C is the temperature on ',find)
```

```
Enter days : sun,mon,tue
Enter temperature : 84,77,66
Enter day to know temp : mon
77 is the temperature on  mon
```

Question 2

2. Given an array of names of candidates in an election. A candidate name in array represents a vote casted to the candidate. Print the name of candidates received Max vote. If there is tie, print lexicographically smaller name.
   Examples:
   Input: votes = ["Raja", "Raja", "Elango",

```
        "Elango", "Elango", "Elango",
        "Elango", "john", "john",
        "Alfred", "Raja", "Raja",
        "Raja"];
```

Output: Elango
We have four Candidates with name as 'Raja',
'Elango', 'john', 'Alfred'. The candidates
Raja and Elango get maximum votes. Since Elango is alphabetically smaller, we print it.

```
from collections import  Counter
votes = [votes for votes in input("Enter votes : ").split(',')] ; cnt=Counter(votes)
max_votes = max(cnt.values())
lst=[i for i in cnt.keys() if cnt[i]==max_votes]
print(sorted(lst)[0])
```

```
Enter votes : "Raja", "Raja", "Elango", "Elango", "Elango", "Elango", "Elango", "johr
 "Elango"
```

Question 3

3. Write a program performing the following tasks over the dictionary: phonebook= {’John’: "00903234561",’Mary’: "02359332865",’Bill’: "004934784530"}
   I. return all names of the phonebook (use function keys(), i.e. tel.keys())
   II. return all phone numbers of the phonebook (use function values())
   III. return all names and phone numbers in tuples of type (name,tel) (use function items())
   IV. ask for an alphabetically sorted telephone book
   V. ask for the phone number for Mary (you can access values by using dict[key])
   VI. add new entries in the phonebook (use the function update(key=value) and tel[key] = value)
   VII. delete an entry (you can use: del dict[key]))
   VIII. check whether there is "John" on the phonebook (use: key in dict)

```
name=[name for name in input("Enter the names of contact : ").split(",")]
num =[num for num in input("ENter number of the contact : ").split(',')]
tel=dict(zip(name,num))
print("Names in phonebook :- \n",tel.keys())
print("Numbers in phonebook :- \n",tel.values())
print("Names with numbers :-\n",tel.items())
print("Sorted contacts :-\n",sorted(tel))
x=input("Enter contact name to get number : ")
print(f"{x}'s number is {tel[x]}")
```

```
y=input("Enter contact name to add : ")
z=input(f"Enter {y}`s number : ")
tel.update({y:z})
print(tel)
a=input("Enter name to delete : ")
del tel[a] ; print(tel)
key = input("Enter name to search : ")
if key in tel:
  print(f"{key}'s contact is in the phonebook")
```

```
Enter the names of contact : ram,mary,john,chand
ENter number of the contact : 2001,8861,8762,1601
Names in phonebook :-
 dict_keys(['ram', 'mary', 'john', 'chand'])
Numbers in phonebook :-
 dict_values(['2001', '8861', '8762', '1601'])
Names with numbers :-
 dict_items([('ram', '2001'), ('mary', '8861'), ('john', '8762'), ('chand', '1601')])
Sorted contacts :-
 ['chand', 'john', 'mary', 'ram']
Enter contact name to get number : mary
mary's number is 8861
Enter contact name to add : sai
Enter sai`s number : 9481
{'ram': '2001', 'mary': '8861', 'john': '8762', 'chand': '1601', 'sai': '9481'}
Enter name to delete : sai
{'ram': '2001', 'mary': '8861', 'john': '8762', 'chand': '1601'}
Enter name to search : john
john's contact is in the phonebook
```

Question 4

4. Given a sentence containing n words/strings. Remove all duplicates words/strings which are similar to each other's.

Input: Python is great and Java is also great
Output: Python is great and Java also

```
s = input("enter a string : ")
l = s.split()
k = []
for i in l:
  if (s.count(i)>1 and (i not in k)or s.count(i)==1):
    k.append(i)
print(' '.join(k))
```

```
enter a string : Python is great and Java is also great
Python is great and Java also
```

Question 5

5. Write a function called wordcount () that accepts a list of strings returns a dictionary where the keys are the words and the values are the number of times that word appears.
   Input and Output:
   Enter the String Python is great and Java is also great
   {'Python': 1, 'is': 2, 'great': 2, 'and': 1, 'Java': 1, 'also': 1}

```
from collections import  Counter
def wordcount():
  lst=[lst for lst in input("Enter strings : ").split(' ')]
  count=Counter(lst)
  print("Count and Occurences are :-\n",count)
wordcount()
```

```
Enter strings : Python is great and Java is also great.
Count and Occurences are :-
 Counter({'is': 2, 'Python': 1, 'great': 1, 'and': 1, 'Java': 1, 'also': 1, 'great.'
```

Question 6

6. Kannan stationery store saves the price details in the price dictionary details are as follwows.

PEN 10
NOTEBOOK 20
PENCIL 5
SCALE 10
STAPLER 40
ERASER 2
HIGHLIGHTER 15

   Write a python program to solve the following task.
   1 Read the list of items and the number of the item bought by the customer.
   2 Compute the bill for the customer.
   SAMPLE INPUT:
   Enter the customer name:
   Elango
   Enter the list of item and number item bought by customers:
   PEN 2, NOTEBOOK 4, STAPLER 2, HIGHLIGHTER 4
   SAMPLE OUTPUT:
   Name: Elango

Item Number of item Price
PEN 2 20
NOTEBOOK 4 80

STAPLER 2 80

HIGHLIGHTER 4 60

Total amount to be paid: 240

```
price = {'pen':10,'note':20,'pencil':5,'scale':10,'eraser':2,'high':15}
name = input("Enter customer name : ")
item = input("Enter the item and quantity : ").split(',')
tot=0;pay=0
for i in item:
  i=i.split(" ")
  quantity = int(i[1])
  pay=price[i[0]]*quantity
  print(i[0],'\t',quantity,'\t x',pay)
  tot=tot+pay
print("Price to pay = Rs.",tot)
```

```
Enter customer name : ram
Enter the item and quantity : pen 2,note 5,pencil 3
pen      2       x 20
note     5       x 100
pencil   3       x 15
Price to pay = Rs. 135
```