TYPE-C : STRING MANUPULATION-CH:10

1) Write a program to count the number of times a character occurs in the given string.

sol:

```
n=input('enter the string:')
to=input('enter chr to count:')
count=n.count(to)
print(count,'times',to,'occurs in the string',n)
```

output:

```
enter the string:python compiler
enter chr to count:o
2 time o occurs in the string python compile
```

2) Write a program which replaces all vowels in the string with '*'.

sol:

```
new=' '
vowels='AEIOUaeiou'
string=input('enter the string:')
for i in string:
    if i in vowels:
        new+='*'
    else:
        new+=i
print(new)
```

output:

```
enter the string:hello python
h*ll* pyth*n
```

3) Write a program which reverses a string and stores the reversed string in a new string.

sol:

```
a=input('enter the string:')
b=a[::-1]
print('the reversed string is:',b)
```

output:

```
enter the string:hello world
the reversed string is: dlrow olleh
```

4) Write a program that prompts for a phone number of 10 digits and two dashes, with dashes after the area code and the next three numbers. For example, 017-555-1212 is a legal input. Display if the phone number entered is valid format or not and display if the phone number is valid or not (i.e., contains just the digits and dash at specific places.)

sol:

```
phNo = input("Enter the phone number: ")
length = len(phNo)
if length == 12 \
    and phNo[3] == "-" \
    and phNo[7] == "-" \
    and phNo[:3].isdigit() \
    and phNo[4:7].isdigit() \
    and phNo[8:].isdigit() :
```

```
        print("Valid Phone Number")
  else :
        print("Invalid Phone Number")
```

output:

```
  Enter the phone number: 017-555-1212
  Valid Phone Number
```

5) Write a program that should do the following :

- prompt the user for a string
- extract all the digits from the string
- If there are digits:
  - sum the collected digits together
  - print out the original string, the digits, the sum of the digits
- If there are no digits:
  - print the original string and a message "has no digits"

Sample

- given the input : abc123
  prints abc123 has the digits 123 which sum to 6
- given the input : abcd
  prints abcd has no digits

sol:

```
  n=input('enter the string: ')
  num=0
  sum=0
  str1=''
  for i in n:
      if i.isdigit():
          str1+=i
          sum+=int(i)
  if n.isalpha():
      print(n,'has no digit')
  else:
      print(n,'has digits',str1,'sum is:',sum)
```

output:

```
  enter the string: py234
  py234 has digits 234 sum is: 9
```

6) Write a program that should prompt the user to type some sentence(s) followed by "enter". It should then print the original sentence(s) and the following statistics relating to the sentence(s) :

- Number of words
- Number of characters (including white-space and punctuation)
- Percentage of characters that are alphanumeric

Hints

- Assume any consecutive sequence of non-blank characters is a word.

sol:

```
n=input('enter the string: ')
print('the original sentence is: ',n)
b=len(n)
sum=0
print('the number of characters including whitespcae is: ',b )
for a in n:
    if a.isalnum():
        sum+=1
percent=(sum/len(n))*100
print('the percent of alnum character is:',percent)
```

output:

```
enter the string: python 123 hello 24 world
the original sentence is:  python 123 hello 24 world
the number of characters including whitespcae is:  25
the percent of alnum character is: 84.0
```

7) Write a Python program as per specifications given below:

- Repeatedly prompt for a sentence (string) or for 'q' to quit.
- Upon input of a sentence s, print the string produced from s by converting each lower case letter to upper case and each upper case letter to lower case.
- All other characters are left unchanged.

For example,

```
Please enter a sentence, or 'q' to quit : This is the Bomb!
tHIS IS THE bOMB!
Please enter a sentence, or 'q ' to quit : What's up Doc ???
wHAT'S UP dOC ???
Please enter a sentence, or 'q' to quit : q
```

sol:

```
while True :
    str = input("Please enter a sentence, or 'q' to quit : ")
    newStr = ""
    if str.lower() == "q" :
        break
    for ch in str :
        if ch.islower() :
            newStr += ch.upper()
        elif ch.isupper() :
            newStr += ch.lower()
        else :
            newStr += ch
    print(newStr)
```

output:

```
Please enter a sentence, or 'q' to quit : hello python
HELLO PYTHON
Please enter a sentence, or 'q' to quit : hi iam python
HI IAM PYTHON
Please enter a sentence, or 'q' to quit : HELLO WORLD
hello world
Please enter a sentence, or 'q' to quit : q
```

8) Write a program that does the following :

- takes two inputs : the first, an integer and the second, a string
- from the input string extract all the digits, in the order they occurred, from the string. if no digits occur, set the extracted digits to 0
- add the integer input and the digits extracted from the string together as integers
- print a string of the form : "integer_input + string_digits = sum"

For example :
For inputs 12, 'abc123' → '12 + 123 = 135'
For inputs 20, 'a5b6c7' → '20 + 567 =587'
For inputs 100, 'hi mom' → '100 + 0 = 100'

sol:

```
int1=int(input('enter the number: '))
int2=input('enter the string: ')
c=''
for i in int2:
    if i.isdigit():
        c+=i
sum=int1+int(c)
print('forinputs',int1,int2,'result is',int1,'+',c,'=',sum)
```

output:

```
enter the number: 65
enter the string: python 2468
forinputs 65 python 2468 result is 65 + 2468 = 2533
```

9) Write a program that takes two strings from the user and displays the smaller string in single line and the larger string as per this format :

sol:

```
str1 = input("Enter first string: ")
str2 = input("Enter second string: ")
small = str1
large = str2
if len(str1) > len(str2) :
    large = str1
    small = str2
print(small)
lenLarge = len(large)
for i in range(lenLarge // 2) :
    print(' ' * i, large[i], ' ' * (lenLarge - 2 * i), large[lenLarge - i - 1], sep='')
```

output:

```
Enter first string: Python
Enter second string: PANDA
PANDA
P      n
 y    o
  t  h
```

10) Write a program to convert a given number into equivalent Roman number (store its value as a string). You can use following guidelines to develop solution for it:

- From the given number, pick successive digits, using %10 and /10 to gather the digits from right to left.

- The rules for Roman Numerals involve using four pairs of symbols for ones and five, tens and fifties, hundreds and five hundreds. An additional symbol for thousands covers all the relevant bases.
- When a number is followed by the same or smaller number, it means addition. "II" is two 1's = 2. "VI" is 5 + 1 = 6.
- When one number is followed by a larger number, it means subtraction. "IX" is 1 before 10 = 9. "IIX isn't allowed, this would be "VIII". For numbers from 1 to 9, the symbols are "I" and "V", and the coding works like this. "I" , "II", "III", "IV", "V", "VI", "VII", "VIII", "IX".
- The same rules work for numbers from 10 to 90, using "X" and "L". For numbers from 100 to 900, using the symbols "C" and "D". For numbers between 1000 and 4000, using "M".

Here are some examples. 1994 = MCMXCIV, 1956 = MCMLVI, 3888= MMMDCCCLXXXVIII

sol:

```
n = int(input("Enter the number: "))
num = (1000, 900,  500, 400, 100,  90, 50,  40, 10,  9,   5,  4,   1)
rom = ('M',  'CM', 'D', 'CD','C', 'XC','L','XL','X','IX','V','IV','I')
result = ''
for i in range(len(num)) :
    count = int(n / num[i])
    result += str(rom[i] * count)
    n -= num[i] * count
print(result)
```

output:

```
Enter the number: 3888
MMMDCCCLXXXVIII
```

11) Write a program that asks the user for a string (only single space between words) and returns an estimate of how many words are in the string. (Hint. Count number of spaces)

sol:

```
count=0
word=0
string=input('enter the senence: ')
for i in string:
    if i.isspace():
        count+=1
    if i.isalnum:
        word+=1
print('no.of letters is:',word)
print("no.of space is:",count)
```

output:

```
enter the senence: twinkle twinkle little star
no.of letters is: 27
no.of space is: 3
```

12) Write a program to input a formula with some brackets and checks, and prints out if the formula has the same number of opening and closing parentheses

sol:

```
str=input('enter the formula:')
c1=c2=0
for i in str:
    if i=='(':
        c1+=1
```

```
        if i==')':
            c2+=1
    if c1==c2:
        print('the formula has both ending and closing')
    else:
        print('the formula does not have proper closing and opening')
```

output:

```
  enter the formula:(a+b)*(a-b)
  the formula has both ending and closing
```

13) Write a program that inputs a line of text and prints out the count of vowels in it.

sol:

```
  str=input('enter the string: ')
  vowels='AEIOUaeiou'
  count=0
  for i in str:
      if i in vowels:
          count+=1
  print(count,'vowels present in the given string')
```

output:

```
  enter the string: hello python world
  4 vowels present in the given string
```

14) Write a program to input a line of text and print the biggest word (length wise) from it.

sol:

```
  str=input('enter a sentence: ')
  a=str.split()
  l=''
  for i in a:
      if len(i)>len(l):
          l=i
  print('the longest word is: ',l)
```

output:

```
  enter a sentence: life is very precious
  the longest word is:  precious
```

15) Write a program to input a line of text and create a new line of text where each word of input line is reversed.

sol:

```
  str=input('enter a sentence: ')
  new=''
  for i in str.split():
      a=i[::-1]
      new+=a+' '
  print('the reversed sentence is:',new)
```

output:

```
  enter a sentence: life is very short
  the reversed sentence is: efil si yrev trohs
```