"""

Write a python program to compute following operations on String:

a) To display word with the longest length

b) To determines the frequency of occurrence of particular character in the string

c) To check whether given string is palindrome or not

d) To display index of first appearance of the substring

e) To count the occurrences of each word in a given string

"""

def Display\_word\_with\_longest\_length() :

Str = input("Enter the main string : ") # Its assumed that string contain only characters and spaces (multile spaces are allowed)

M\_str = ""

i = 0

while( i < len(Str)) :

word = ""

while(Str[i] != ' ') :

word += Str[i]

i = i + 1

if( i == len(Str)) :

break

if(i != len(Str)) :

while(Str[i] == ' ') :

i = i + 1

if(len(M\_str) < len(word)) :

M\_str = word

print("\tWord with longest length is %s having lenght %d\n\n"%(M\_str,len(M\_str)))

def Determine\_frequency\_of\_occurrence\_of\_particular\_character\_in\_string() :

Str = input("Enter the string : ")

C = input("Enter the character : ")

print("\tString : %s"%Str)

print("\tCharacter : %s"%C)

count = 0

for i in range(len(Str)) :

if(Str[i] == C) :

count += 1

print("\tFrequency of occurrence of character(%s) in string(%s) is %d\n\n"%(C,Str,count))

def Check\_for\_palindrome() :

Str = input("Enter the string to be checked : ")

b = 0

e = len(Str) - 1

while( b < e) :

if(Str[b] != Str[e]) :

break

b += 1

e -= 1

if(b < e) :

print("\t%s string is not an palindrome string\n\n"%Str)

else :

print("\t%s string is an palindrome string\n\n"%Str)

def display\_index\_of\_first\_appearance\_of\_the\_substring() :

M = input("Enter the main string : ")

S = input("Enter the sub string to check : ")

print("Main String : %s"%M)

print("Substring String : %s"%S)

L1 = len(M)

L2 = len(S)

if(L1 >= L2) :

for i in range((L1 - L2 + 1)) :

flag = 1

for j in range(L2):

if(M[i+j] != S[j]) :

flag = 0

break

if(flag == 1) :

print("Substring %s found at index %d\n\n"%(S,i))

break;

if(flag == 0) :

print("Substring not found in the main string\n\n")

else :

print("Substring is greater than main string\n\n")

def Count\_\_occurrences\_of\_each\_word\_in\_given\_string() :

Str = input("Enter the main string : ") # Its assumed that string contain only characters and spaces (multile spaces are allowed)

i = 0

Word\_array = []

Count = []

while( i < len(Str)) :

word = ""

while(Str[i] != ' ') :

word += Str[i]

i = i + 1

if( i == len(Str)) :

break

if(i != len(Str)) :

while(Str[i] == ' ') :

i = i + 1

if(len(Word\_array) == 0) :

Word\_array.append(word)

Count.append(1)

else :

flag = 1

for j in range(len(Word\_array)) :

if(Word\_array[j] == word) :

Count[j] += 1

flag = 0

break

if (flag == 1) :

Word\_array.append(word)

Count.append(1)

for i in range(len(Word\_array)) :

print("\t%15s : %d "%(Word\_array[i],Count[i]))

def main():

while True :

print ("\t\t \*\*\*\* STRING OPERATIONS \*\*\*\*")

print ("\t\t1 : Display word with longest length")

print ("\t\t2 : Determine the frequency of occurrence of particular character in the string")

print ("\t\t3 : Check whether given string is palindrome or not ")

print ("\t\t4 : Display index of first appearance of the substring")

print ("\t\t5 : Count the occurrences of each word in a given string")

print ("\t\t6 : Exit")

ch = int(input("Enter your choice : "))

if (ch == 6):

print ("End of Program")

quit()

elif (ch == 1) :

Display\_word\_with\_longest\_length()

elif (ch == 2) :

Determine\_frequency\_of\_occurrence\_of\_particular\_character\_in\_string()

elif (ch == 3) :

Check\_for\_palindrome()

elif (ch == 4) :

display\_index\_of\_first\_appearance\_of\_the\_substring()

elif (ch == 5) :

Count\_\_occurrences\_of\_each\_word\_in\_given\_string()

else :

print ("Wrong choice entered !! Try again")

main()