**Artificial Intelligence**

**Name:Abdul Moiz Asif**

**CMS:263802**

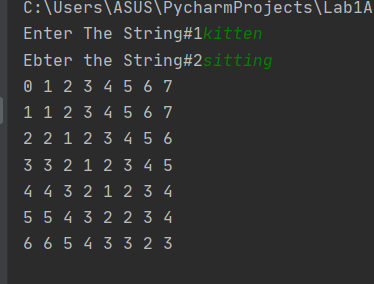
**Class: BSCS 8C**

**Lab#1**

**Task#1**

#using numpy for matrices  
import numpy as np  
#printing the matrix  
def printDistance():  
 for t1 in range(lengthstr1 + 1):  
 for t2 in range(lengthstr2 + 1):  
 print(int(matrix[t1][t2]), end=" ")  
 print()  
#the whole LevesheitDistance Algo  
def levensheit():  
 #printing the 1st row and column as indexes  
 for t1 in range(lengthstr1+1):  
 matrix[t1][0]=t1  
 for t2 in range(lengthstr2+1):  
 matrix[0][t2]=t2  
 a=0  
 b=0  
 c=0  
 for t1 in range(1,lengthstr1+1):  
 for t2 in range(1,lengthstr2+1):  
 #if two characters are equal then print the top left of the matrix  
 if(str1[t1-1]==str2[t2-1]):  
 matrix[t1][t2]=matrix[t1-1][t2-1]  
 #if not equal then find the min of the three enteries+1  
 else:  
 a=matrix[t1-1][t2]  
 b = matrix[t1][t2-1]  
 c = matrix[t1 - 1][t2-1]  
 if(a<=b and a<=c):  
 matrix[t1][t2]=a+1  
 elif (b <= c and b <= a):  
 matrix[t1][t2] = b + 1  
 else:  
 matrix[t1][t2] = c + 1  
  
  
str1=input("Enter The String#1")  
str2=input("Ebter the String#2")  
lengthstr1=len(str1)  
lengthstr2=len(str2)  
matrix=np.zeros((lengthstr1+1,lengthstr2+1))  
levensheit()  
printDistance()

**Output**

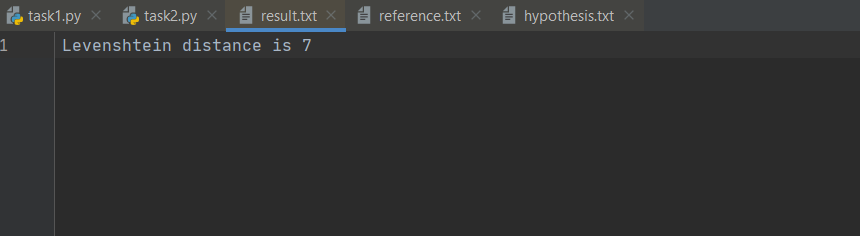
****

**Task#2**

#using numpy for matrices  
import numpy as np  
#printing the matrix  
def printDistance():  
 for t1 in range(lengthstr1 + 1):  
 for t2 in range(lengthstr2 + 1):  
 print(int(matrix[t1][t2]), end=" ")  
 print()  
#the whole LevesheitDistance Algo  
def levensheit():  
 #printing the 1st row and column as indexes  
 for t1 in range(lengthstr1+1):  
 matrix[t1][0]=t1  
 for t2 in range(lengthstr2+1):  
 matrix[0][t2]=t2  
 a=0  
 b=0  
 c=0  
 for t1 in range(1,lengthstr1+1):  
 for t2 in range(1,lengthstr2+1):  
 #if two characters are equal then print the top left of the matrix  
 if(str1[t1-1]==str2[t2-1]):  
 matrix[t1][t2]=matrix[t1-1][t2-1]  
 #if not equal then find the min of the three enteries+1  
 else:  
 a=matrix[t1-1][t2]  
 b = matrix[t1][t2-1]  
 c = matrix[t1 - 1][t2-1]  
 if(a<=b and a<=c):  
 matrix[t1][t2]=a+1  
 elif (b <= c and b <= a):  
 matrix[t1][t2] = b + 1  
 else:  
 matrix[t1][t2] = c + 1  
  
  
ref=open("reference.txt",'r')  
ref1=ref.read()  
str1=ref1.split()  
hypot=open("hypothesis.txt",'r')  
hypo1=hypot.read()  
str2=hypo1.split()  
lengthstr1=len(str1)  
lengthstr2=len(str2)  
matrix=np.zeros((lengthstr1+1,lengthstr2+1))  
levensheit()  
print()  
f = open("result.txt", "w")  
f.write("Levenshtein distance is "+ str(int(matrix[lengthstr1][lengthstr2])))  
f.close()  
printDistance()

**Output**

Result.txt

****