**Digital Assignment 1**

**Minimum four test cases should be there for each question. Give input which test all constraints of the question**.

1. Write a program to accept an array of strings and an original string that is to be compared, Display an array of boolean values - True if the word can be formed from the original word by swapping two letters only once and False otherwise.

Examples

Array : ["BACDE", "EBCDA", "BCDEA", "ACBED"]

Original string : "ABCDE"

➞ [True, True, False, False]

# Swap "A" and "B" from "ABCDE" to get "BACDE".

# Swap "A" and "E" from "ABCDE" to get "EBCDA".

# Both "BCDEA" and "ACBED" cannot be formed from "ABCDE" using only a single swap

1. Write program to encrypt and decrypt a message. For encryption accept a string form user and encrypt it as a list of numbers which can be formed by finding the difference between the adjacent characters . For decryption , accept a list of numbers from user and decrypt as a string.

Example:

Encryption:

"Hello" ➞ [72, 29, 7, 0, 3]

# H = 72, the difference between the H and e is 29 (upper- and lowercase).

# The difference between the two l's is obviously 0.

"Sunshine" ➞ [83, 34, -7, 5, -11, 1, 5, -9]

Decryption :

[72, 29, 7, 0, 3] ➞ "Hello”

72=H

72+29=101=>e

…….

1. Accept two strings form the user . check whether the first string can be completed ( same as a second string ) by adding some characters to the first string without removing any character and without the changing the order of the character in the first string. If it so print “True” else “False”

Example :

("butl", "beautiful") ➞ True

# We can add "ea" between "b" and "u", and "ifu" between "t" and "l".

"tulb", "beautiful" ➞ False

# Although "t", "u", "l" and "b" all exist in "beautiful", they are incorrectly ordered.

1. Write a program that takes one string ( which is a one date ) and a list of date dictionaries. Count how many consecutive dates are there including the given date and print .

Eg:

String ="2019-09-23",

List = [ { "date": "2019-09-18" }, { "date": "2019-09-19" }, { "date": "2019-09-21" }, { "date": "2019-09-22" } , { "date": "2019-09-23" }])

➞ 3

String ="2019-09-25",

List = [ { "date": "2019-09-18" }, { "date": "2019-09-19" }, { "date": "2019-09-21" }, { "date": "2019-09-22" } , { "date": "2019-09-23" }])

➞ 0

Because 2019-09-25 is not there in the list.

Even though 21,22,23 is consecutive.

1. Write a program to accept the string form user and check for the validity. The string is valid if the string all the character at the same number of times or if we remove one character one time , the string become valid . print Yes or no.

Eg :

"abc" = Yes

“abcc”= yes => we can remove one ‘c’

“abccc”= No => even if we remove one ‘c’ it is not a valid string.

"abcdefghhgfedecba" =yes => All characters occur twice except for e which occurs 3 times.

# We can delete one instance of e to have a valid string.

"aabbccddeefghi"= NO => we need to remove 5 characters.