

**Name:** Siddarth Venkateswaran

**Matriculation No:** 11011753

## Exercise 1: Finding the Longest Common Subsequence of two strings

The whole idea behind solving this problem was to take two strings, find the pattern in which similar characters appeared with incremental index values across the strings, and finally display that pattern as a third string value. This was done using a hard-coded logic with fixed input values. In the code executed, the input strings S1 and S2 were assigned with values "ABAZDC" and "BACBAD" respectively. Through incremental indexing, the expected outcome was A = "ABAD", however the actual outcome was A = "ABADC" which is incorrect.

### Pseudocode:

Assign a value of "ABAZDC" for string S1

Assign a value of "BACBAD" for string S2

Assign a blank value for string A

//Checking for the earliest instance of the first index value of S1 in S2

If the first index value in S1 is equal to the first index value in S2:

Append A with the first index value of S2

Else if the first index value in S1 is equal to the second index value of S2:

Append A with the second index value of S2

Else if the first index value of S1 is equal to the third index value of S2:

Append A with the third index value of S2

Else if the first index value of S1 is equal to the fourth index value of S2:

Append A with the fourth index value of S2

Else if the first index value of S1 is equal to the fifth index value of S2:

Append A with the fifth index value of S2

Else if the first index value of S1 is equal to the sixth index value of S2:

Append A with the sixth index value of S2

Else:

Make no changes to the value of A

Print the current value of A

//Checking for the earliest instance of the second index value of S1 in S2

If the second index value of S1 is equal to the first index value of S2, and the first index value of S1 is neither equal to the first nor the second index values of S2:

Append A with the first index value of S2

Else if the second index value of S1 is equal to the second index value of S2, and the first index value of S1 is neither equal to the second nor the third index values of S2:

Append A with the second index value of S2

Else if the second index value of S1 is equal to the third index value of S2, and the first index value of S1 is neither equal to the third nor the fourth index values of S2:

Append A with the third index value of S2

Else if the second index value of S1 is equal to the fourth index value of S2, and the first index value of S1 is neither equal to the fourth nor the fifth index values of S2:

Append A with the fourth index value of S2

Else if the second index value of S1 is equal to the fifth index value of S2, and the first index value of S1 is neither equal to the fifth nor the sixth index values of S2:

Append A with the fifth index value of S2

Else if the second index value of S1 is equal to the sixth index value of S2, and the first index value of S1 is not equal to the sixth index value of S2:

Append A with the sixth index value of S2

Else:

Make no changes to A

Print the current value of A

//Checking for the earliest instance of the third index value of S1 in S2

If the third index value of S1 is equal to the first index value of S2, and neither the first nor the second index values of S2 are not equal to either the first or the second index values of S1:

Append A with the first index value of S2

Else if the third index value of S1 is equal to the second index value of S2, and neither the second nor the third index values of S2 are not equal to either the second or the third index values of S1:

Append A with the second index value of S2

Else if the third index value of S1 is equal to the third index value of S2, and neither the third nor the fourth index values of S2 are not equal to either the third or the fourth index values of S1:

Append A with the third index value of S2

Else if the third index value of S1 is equal to the fourth index value of S2, and neither the fourth nor the fifth index values of S2 are not equal to either the fourth or the fifth index values of S1:

Append A with the fourth index value of S2

Else if the third index value of S1 is equal to the fifth index value of S2, and neither the fifth nor the sixth index values of S2 are not equal to either the fifth or the sixth index values of S1:

Append A with the fifth index value of S2

Else if the third index value of S1 is equal to the sixth index value of S2, and the sixth index value of S2 is not equal to the sixth index value of S1:

Append A with the sixth index value of S2

Else:

Make no changes to A

Print the current value of A

//Checking for the earliest instance of the fourth index value of S1 in S2

If the fourth index value of S1 is equal to the first index value of S2, and the first 3 index values of S1 are not equal to the first 3 index values of S2:

Append A with the first index value of S2

Else if the fourth index value of S1 is equal to the second index value of S2, and the second, third or fourth index values of S1 are not equal to the second, third or fourth index values of S2:

Append A with the second index value of S2

Else if the fourth index value of S1 is equal to the third index value of S2, and the third, fourth or fifth index values of S1 are not equal to the third, fourth or fifth index values of S2:

Append A with the third index value of S2

Else if the fourth index value of S1 is equal to the fourth index value of S2, and the fourth, fifth or sixth index values of S1 are not equal to the fourth, fifth or sixth index values of S2:

Append A with the fourth index value of S2

Else if the fourth index value of S1 is equal to the fifth index value of S2, and the fifth or sixth index values of S1 are not equal to the fifth or sixth index values of S2:

Append A with the fifth index value of S2

Else if the fourth index value of S1 is equal to the sixth index value of S2, and sixth index value of S1 is not equal to the sixth index value of S2:

Append A with the sixth index value of S2

Else:

Make no changes to A

Print the current value of A

//Checking for the earliest instance of the fifth index value of S1 in S2

If the fifth index value of S1 is equal to the first index value of S2, and the first 4 index values of S1 are not equal to the first 4 index values of S2:

Append A with the first index value of S2

Else if the fifth index value of S1 is equal to the second index value of S2, and the second, third, fourth or fifth index values of S1 are not equal to the second, third, fourth or fifth index values of S2:

Append A with the second index value of S2

Else if the fifth index value of S1 is equal to the third index value of S2, and the third, fourth, fifth or sixth index values of S1 are not equal to the third, fourth, fifth or sixth index values of S2:

Append A with the third index value of S2

Else if the fifth index value of S1 is equal to the fourth index value of S2, and the fourth, fifth or sixth index values of S1 are not equal to the fourth, fifth or sixth index values of S2:

Append A with the fourth index value of S2

Else if the fifth index value of S1 is equal to the fifth index value of S2, and the fifth or sixth index values of S1 are not equal to the fifth or sixth index values of S2:

Append A with the fifth index value of S2

Else if the fifth index value of S1 is equal to the sixth index value of S2, and the sixth index value of S1 is not equal to the sixth index value of S2:

Append A with the sixth index value of S2

Else:

Make no changes to A

Print the current value of A

//Checking for the earliest instance of the sixth index value of S1 in S2

If the first index value of S1 is equal to the sixth index value of S2, and the first 5 index values of S1 are not equal to the first 5 index values of S2:

Append A with the first index value of S2

Else if the first index value of S1 is equal to the second index value of S2, and the second, third, fourth, fifth or sixth index values of S1 are not equal to the second, third, fourth, fifth or sixth index values of S2:

Append A with the second index value of S2

Else if the first index value of S1 is equal to the third index value of S2, and the third, fourth, fifth or sixth index values of S1 are not equal to the third, fourth, fifth or sixth index values of S2:

Append A with the third index value of S2

Else if the first index value of S1 is equal to the fourth index value of S2, and the fourth, fifth or sixth index values of S1 are not equal to the fourth, fifth or sixth index values of S2:

Append A with the fourth index value of S2

Else if the first index value of S1 is equal to the fifth index value of S2, and the fifth or sixth index values of S1 are not equal to the fifth or sixth index values of S2:

Append A with the fifth index value of S2

Else if the first index value of S1 is equal to the sixth index value of S2, and the sixth index value of S1 is not equal to the sixth index value of S2:

Append A with the sixth index value of S2

Else:

Make no changes to A

Print the final value of A

GitHub Link:

<https://github.com/sidvenkat1988/Big-Data-Programming/blob/master/Exercise1.py>

## Exercise 2: Pattern Matching

This problem involved entering values a-z in string A and entering values a-z along with symbols "." and "\*" in string B. The code was conditioned as follows:

- 1) If the length of string A is greater than the length of string B, then "\*" would be replaced by multiple instances of its single previous character until the lengths of both the strings matched
- 2) If the first index value of string B is ".", it would be replaced by the first index value of string A
- 3) If strings A and B were of the same length, or if B was longer than A, then "\*" would be replaced by a single instance of its single previous character in the string B and
- 4) If "." didn't occupy the first index value in the string B, then it would be replaced by a single instance of its single previous character in string B.

Once these conditions were put in place, tests were performed to check the presence of different values of A in B and the output would be a Boolean value of either True or False.

For eg. If string A had a value of "AAB" and B had a value of "C.A\*B", the value of B would be changed to "CCAAB" and the output would be True.

Pseudocode:

Enter a value with characters a-z for string A

Enter a value comprising of a-z with symbols '.' and '\*' for string B

Calculate the absolute difference in lengths between strings A and B and assign it to a variable C

Print the value of C

//Replacing \* with single or multiple instances of previous single character

For iterator i between a range of 0 and the length of string B:

    If an indexed value of string B is "\*":

        If the length of string A is greater than that of B:

            Replace "\*" with the multiple instances of its previous single character until the length of string B becomes equal to that of A

        Else:

            Replace "\*" with a single instance of its previous single character

Print the current value of B

//Replacing "." with a single instance of it's previous single character

For iterator i between a range of 0 and the length of string B:

    If an indexed value of string B is ".":

        Replace it with a single instance of it's previous single character

Print the current value of B.

//Replacing "." with the first indexed value of A, if it is the first indexed value of B

If the first indexed value of string B is ".":

    Replace it with the first indexed value of string A

Print the current value of B

Check if the pattern in string A is available in string B.

GitHub Link:

<https://github.com/sidvenkat1988/Big-Data-Programming/blob/master/Exercise2.py>

References:

Though these were executed using my limited programming knowledge, all the basic syntax were taken done with the help of w3schools (<https://www.w3schools.com/python/default.asp>)

Replacing "." and "\*" were done by referring to this link in w3schools:

[https://www.w3schools.com/python/ref\\_string\\_replace.asp](https://www.w3schools.com/python/ref_string_replace.asp)