IC152: Assignment 2
String Methods and Reading/Writing Text Files

In this assignment, we will apply string methods, operations and learn how to read, modify, and write the text files using python.

Text or strings form an important part of online content. Tasks (sometimes based on machine learning) such as transliteration, translation, text correction, question generation involve modifying text. Some tasks like speech recognition, or optical character recognition involve post text correction steps. Some advertisement companies might be using the text we type on social media platforms to display advertisements of interest. You might have heard or read articles on people posting sensitive texts on such platforms and affecting the feelings of many. The effect of even a single line of text can go beyond our imaginations, which is generally seen on twitter which limits tweet length to 280 characters. Nowadays we can even generate poems using Artificial Intelligence, but we need to be careful while posting the created/modified text online!

Download the assignment from moodle, unzip and paste folder assignment2/ on Desktop. Please save a separate python file for each question in this folder (~/Desktop/assignment2/). Give appropriate names to the files e.g., q1Problem1a.py. Make sure that you use the meaningful variable names and add proper comments to your code. Here, we go:

Important: Save all your python files in this folder assignment2/ you've placed on Desktop.

Problem 1:

1) Write a Python script that rotates a string to the left by a given number of positions. For example, if the input string is "abcdef" and the number of positions is 2, the output should be "cdefab".
2) Write a Python script to determine if one string is a substring of another. Specifically, the script should perform the following tasks:
    i) Prompt the user to input two strings, string_1 and string_2.
    ii) Check whether string_2 is a substring of string_1.
    iii) Output a message indicating whether string_2 is a substring of string_1
3) Write a Python script that replaces all vowels in a given string with the question mark character (?). The script should follow these specifications:

Input:

Prompt the user to input a single string.

Processing:

Replace each vowel (both uppercase and lowercase) in the input string with the character “?”.

Output:

Print the modified string with all vowels replaced by “?”.

4) Write a Python script to check if a string is palindrome or not. Prompt the user to input a string (using input() function), and then print if it is a palindrome or not.

(Example: "abcde" is not a palindrome but "abcba" is a palindrome.)

Problem 2: You are given two strings, str1 and str2, for example: “programming” and “prototyping”, of the same length. Write a program that compares the characters at each index of both strings. If the characters at a particular even index in str1 and str2 are the same, store that character in a new string called AnsStr. Finally, print the value of AnsStr.

Problem 3:

1. Write a program in python that takes a string and a character input from the user as InputStr and InputChr and finds the first index, last index and total number of occurrences of the InputChar between the first and the last occurrences of the InputChr in the InputStr inclusive. The input character is case **insensitive** i.e uppercase and lowercase letters are to be treated the same. In case where the InputChr is not present in the InputStr print -1.

   E.g InputStr = “Coding is amazing, everyone should code” InputChr = ‘o’, then the program should print first index = 1, last index = 36 and total occurrences of InputChr = 4.

   a. Use index()/find(), rindex()/rfind() methods and slicing for the above problem.

   b. Solve the problem without using inbuilt functions like index() or rindex() in a single loop. (**<u>you can use .lower() or .upper() for case insensitivity</u>**)

Hint code: If you want to repeat “a = a + 1” n/2 times, then code is:

```
for i in range(n):
        If (i&1):
                a = a+1
                print(“a is appended when i is”, i)
```

c. Try to reason which approach is faster.

Problem 4: Given the expression, “In a Democracy, the government is of the people, by the people, and for the people, still people don’t go out and vote”.

1) Replace the first occurrence of “people” with “PEOPLE”, the second and third ones with “pEOPLE” (the second and third occurrence should be replaced together using the replace function once) and the fourth one with “peoPLE”. Use the print command to show successive replacements.
2) Repeat 1 using string slicing (Don’t count the indices manually!!).
3) Try to reason which program is faster.

Problem 5: Auto correct the text in file ‘inputWithErrors.txt’. Parts 1-4 below are for understanding and implementing as it is. You only have to write code in the 5th part:-

1. Save the folder 'assignment2/' on the Desktop. If you haven’t, you will get the “file not found” error in the next parts.
2. Read the file ‘inputWithErrors.txt’ in a list named 'lines' using following statements:

```
import os
with open(os.path.expanduser('~') + '/Desktop/assignment2/inputWithErrors.txt') as f:
        lines= [line.rstrip() for line in f]
```

3. Ensure that the variable named 'lines' have the content from '~/Desktop/assignment2/inputWithErrors.txt':

   ```
   print(lines)
   ```

4. Read the lines, and spot following types of errors (no coding, only observe):
   a. Incorrect cases. E.g., Heading is 'features of Python' and not 'Features Of Python', and some of the sub-headings/sentences start with lowercase.
   b. An extra space before characters like '.', ',', ')'
   c. Missing space before '('
5. Write the code using string methods like replace(), upper(), lower(), title(), etc. to auto correct the content of '~/Desktop/assignment2/inputWithErrors.txt'. Save the final output in '~/Desktop/assignment2/outputClean.txt'
   a. You can loop over the lines loaded in 1.1 and save the modified lines in output file with following commands:

   ```
   import os
   fOutput = open(os.path.expanduser('~') + '/Desktop/assignment2/outputClean.txt', 'w')
   for line in lines:
           newline = line.replace(' .', '.') #correcting one of the 4.b
           #write statements here to correct other errors:-
           #to correct case of title, use firstIter = True flag, as discussed in class
           #to correct headings in each line, use find() method (find ':') and slicing
           #to correct case for start of each line find full stop and use slicing
           #to correct 4c., replace '(' with ' (', then think & perform one more replace
           #save final output in newlineFinal
           fOutput.write(newlineFinal  + '\n')
   fOutput.close()
   ```

   Important:

   - you can use len(lines) and if condition to avoid printing last "\n"
   - Similarly, use if condition to avoid changing the case in the second line

Save the python file with a meaningful file name corresponding to the whole program you have written.

Problem 6 (Optional/Bonus): Write a Python script that analyzes a given string to determine how many times each vowel appears in that string. The program should be case-insensitive, meaning it should treat uppercase and lowercase vowels (a, e, i, o, u) as the same. The output should provide a count for each vowel.
String_1- "IIT Mandi Computer Science" .

Problem7 (Optional/Bonus): Transliterate the text in '~/Desktop/assignment2/hindiInEnglish.txt' to Hindi using string methods discussed in class. Save the output in '~/Desktop/assignment2/hindiPure.txt'

1. Do not worry if you do not know Hindi, take help from TAs and your friends to understand how different characters look and sound in Hindi.

2. OR use these links:
    a. This one is just for reference, do not follow it as it is: https://en.wikipedia.org/wiki/SLP1
    b. https://www.easyhindityping.com/
    c. https://www.lexilogos.com/keyboard/sanskrit_devanagari.html

3. Hint1: write your own dictionary for substring transliterations:
    a. e.g. engToHin = {'m': 'म्', 'e': 'े', 'aa': 'ा', 'a': 'अ', 'r': '' ,'n': '' }
    For simplification: we refer to ma as 'म', and m as 'म्' i.e. ma without the sound of a.
    chr(2381) will print '्' on the shell/console.
    ord('्') will print 2381 on the shell/console.
    b. Use following command in your shell/console to print all characters you need:

```
for i in range(2300,2400):
        print(chr(i))
```

c. If needed copy paste the output of previous command in a new file

4. Write a program to transliterate and save the output in '~/Desktop/assignment2/hindiPure.txt'
    a. You need to pass through the text multiple times before saving.
    b. You need to loop carefully for vowels, e.g. engToHin dict should be applied on 'aa' and not individually on two 'a' if they occur together two or more times.
    c. You need to carefully work with vowels which appear at the beginning of the words or with other vowels. E.g. 'isalie' should be transliterated to 'इसलिए' and not ' िसलिे'
    Hint: Write another dictionary for correcting such text.

Save the python file with a meaningful file name corresponding to the whole program you have written.

Rename the folder “assignment2” with roll number of first student in group excel followed by “_assignment2” (do not use inverted commas in folder name), compress the folder and submit it on moodle. During lab evaluations (later) you will need to be extra careful to follow such naming conventions as some of the assignments will be auto-evaluated based on filenames or folder names.

<u>Make sure that you delete all your files from the lab PC/Laptop, and shut it down before you leave.</u>