Homework Assignment #2

CS 3753/5163

Individual work

Submit one Jupyter notebook named as yourLastName_HW02 with appropriate code cell to solve the following questions. Make sure that the question is included in your notebook as a markdown cell above your answer. You must use only the basic Python and no external module or library is allowed other than math or os modules. Include any specific direction/instruction to run your script in comments. For all questions, the main functionality of your Python script should be implemented as a **Python function**.

Note: if you want to include your script as separate .py files, you may submit all source files with a Jupyter notebook in a zipped file (compressed). In the notebook, inside the code cell of a question, write %run Qx. Where x is the number of the question.

Undergraduate: Each question worth 15 points.
Graduate: Each question worth 10 points.

Q1) Write a Python script to mimic a trivial text editor in search and replace functionality. Your program accepts a text file named “test.txt” and, using a menu-like, can either search for all occurrences of a given string or replace one or all occurrences of a given string. Your program should write back on the original text (update it) when replace option is selected. No need to submit your test.txt you used for debugging. The grader will use his own test.txt.

Q2) Write a Python script that finds the largest difference between elements of a given list. For example, the largest difference in [2, 5, -3, 4, 7, 9] is 12. The list is entered as one single comma-separated string.

Q3) Write a Python script that simplifies a given string. It removes any tabs, newline, or multiple white spaces. For example: the string “   This is   a nice   day\n” will be simplified to “This is a nice day”. The string should be read from a text file called “test2.txt”. The string could be on one or more lines. No need to submit your test2.txt. We will use our own test2.txt to grade your program.

Q4) Write a Python script that contains a recursive function to calculate the nth Fibonacci, where f(n) = 1, n=1,2 and f(n) = f(n-1)+f(n-2), n>2. Your program will accept the value of n.

Q5) Write a Python script that contains your own implementation of the functional tool “filter”. Your function should be named as myFilter with header (foo, iterableObject). For simplicity, the iterable object could be a numerical list, set, or a tuple. Your program accepts it as one single line with the appropriate brackets/braces/parenthesis to denote the type of the object. For example, if the user would like to enter a list, s/he will type in something like: [2, 4, 2, 9].

Q6) Write a Python script that contains a function called foo that accepts a variable number of numerical parameters and returns these numbers sorted (in ascending order) and saved in a list. We will update your code with our invocation to your function.

Q7) Write a Python script to accept a text file called “test3.txt” and returns some statistics about it. Statistics include: #lines, #words, #characters, #printable special characters (anything that is not alphanumeric).

Graduate Students Only

Q8) Write a Python script that contains a function lensort to sort a list of strings based on length. The user will enter the list of strings as one comma-separated line.

Q9) Write a Python script that contains a function “triplets” that takes a number n as argument and returns a list of triplets such that the sum of the first two elements of the triplet equals the third element using numbers below n. Please note that (a, b, c) and (b, a, c) represent the same triplet. Example: triplets (5) produces [(1 , 1 , 2) , (1 , 2 , 3) , (1 , 3 , 4) , (2 , 2 , 4)]

Q10) Write a Python script that contains a function to calculate the angle between 3 given 3d points. Each point is accepted as a comma-separated string as x, y, x and you process it.

Bonus Question: (10 points)

Q11) Write a Python program to explore a given directory in the local machine. Your program accepts a path to a directory from “test4.txt” and does the following (in a menu-like): search for a file, list all sub-directories, list all files with a given extension, remove a given file, and show all files sorted by date/time modified. “test4.txt” will have a single line that contains the full path of the directory we are interested in. For example: the only line in the text could be c:\Program files to denote to the program files directory in c drive. You can use os module.

Due date: 02/21/2019 at 11:59PM

How to submit

Through blackboard. No hard copy is accepted. The system will close after 11:59PM and you will not be able to turn it in. No late submission is accepted unless you receive instructor’s approval no less than two days before the due date.