ISTA 130: Fall 2017
Programming Assignment 8
(100 points)
Lists and File I/O

Due : November 9th at 11:59 pm

(submit via D2L dropbox)

Please read the instructions below carefully and follow them closely.   All problems (and sub-parts of the problems) are required except as noted in the instructions below. If you have any questions, please email the instructor or one of the section leaders.

Important: **You will lose 5 points if you use the tab character for indentation.**

Important: Your filenames must be identical to the filenames given below. For any functions you are asked to write, the function signature (header) must be exactly as described in the instructions. That is, you must use the exact function names given in the instructions, you must have the parameters the instructions ask for, and the parameters must be in the order the instructions give.

Important: Make sure you always save a backup of your work somewhere safe (such as your D2L locker, Dropbox, UA Box, etc).

Important: You are not just graded on whether your code produces the same result as the examples show. You should be using what you've learned to do your best to write good code. For example, things like poorly named variables/functions, duplicating code where you could instead use a loop, and missing documentation will cost you points.

1.0) TERRESTRIAL MAMMAL BRAIN AND BODY WEIGHS (100 POINTS)

A ".csv" file is a text file containing many rows of data in which each row consists of comma-separated values. Example row: `'123.7,64.5,98.98,101.23\n'`.

Download the file named "BrainBodyWeightKilos.csv" from D2L. Open the file with a text editor (not Excel).   This file contains data collected about the weights of the bodies and brains of various terrestrial mammals, from http://mste.illinois.edu/malcz/DATA/BIOLOGY/Animals.html. Each row in the file

represents a single mammal with three values: name, body weight in kilograms, brain weight in grams.

Here are the first two lines and the last line of the file:

African elephant,6654.0,5712.0

African giant pouched rat,1.0,6.6

...

Yellow-bellied marmot,4.05,17.0

The data shows that the African elephant weighed 6654.0 kilograms and its brain weight was 5712.0 grams. The African giant pouched rat weighed 1.0 kilogram and its brain weight was 6.6 grams.

Your program will read the data file, collecting the values into lists. It will then allow the user to view information about an animal, add new animals and/or remove animals. Once the user is finished modifying the data, the program will convert all of the weights from kilograms and grams to pounds and write the data out to a new file.

1.1) DETAILS

Create a new file called "brainbodyweight.py". In the file:

1.) Write a function called **find_insert_position** that takes two parameters. The first parameter is a string representing a terrestrial mammal's name. The second parameter is a list of strings in alphabetical order.

    i.) The function returns an integer indicating the position (index) in the list of names such that if you inserted the given mammal's name into the list at that index the list would still be in alphabetical order.

NOTE: the function does not insert the name, it does not change the names list in any way. It simply finds and returns the appropriate integer index.

2.) Write a function called **populate_lists** takes three parameters. The first parameter is a list to hold the mammal names. The second parameter is a list to hold the mammal

body weights (these should be floats). The last list will hold the mammal brain weights (these should also be floats). This function must open the "BrainBodyWeightKilos.csv" file and then populate each list accordingly. The mammal names **MUST** be in title case.

· when you finish reading the file your name list should contain all of the names, the body weight list should contain all of the body weights and the brain weights list should contain all of the brain weights.

· The first element in each list will be the values for the African elephant, the second element in each list will be values for the African giant pouched rat, and so on...

3.) Write a function called **write_converted_csv** that takes four parameters. The first is the name of a file to write. The second is a list of strings (the names of mammals). The third is a list of floats (weights of mammals in kilograms). The fourth is a list of floats (brain weights of mammals in grams.)

i.) The function writes a new .csv file (with the given filename) in the same format as the original "BrainBodyWeightKilos.csv" using the values in the three lists, but all **kilogram** values and all **gram** values should first be converted to **pounds**, and names should be in title case. Your conversion factor for converting from kilograms should be 2.205; from grams it should be 0.0022.

(Read steps 3 and 4i below for further clarification on the lists that will eventually be passed to this function)

ii.) The floats written to the file should be rounded to 2 decimal places.

4.) In **main**:

i.) Create three empty lists to hold mammal names, body weights, and brain weights.

ii.) Populate the three empty lists you just created.

iii.) Next you will repeatedly ask the user to (note that the prompt is preceded by a blank line):

Convert the user's response to title case. Depending on the user's response:

- If the name entered by the user is not in your list of names (case insensitive search):

    1.i. Print a message indicating the animal was not found.

      – For example, assuming the user entered "Bugbear" you would print:

        File does not contain "Bugbear".

    1.ii. Ask the user if he would like to add it. (You can assume the user will answer either 'y' or 'n'.)

      – For example, assuming the user previously entered "Bugbear":

        Add "Bugbear" to file?   (y|n)

  – If the user enters 'n', do nothing (i.e. you'll go back to asking the user to enter an animal name)
  – If the user enters 'y', ask for the body weight and brain weight:
  - Continuing the previous example you would ask the user for each of the two following:

Enter body weight for "Bugbear" in kilograms:
Enter brain weight for "Bugbear" in grams:

  ∗ Add the new animal's data to your three lists. Put the values in the appropriate positions in the lists so that the animals are still all alphabetized.

- If the name entered by the user is already in your list of names:

i. Print the animal's data:

- For example, assuming the user entered 'Cat'

<p style="color:green; text-align:center;">Cat: body = 3.3, brain = 25.6</p>

ii. Ask the user if he would like to delete the animal. (You can assume the user will answer either 'y' or 'n'.)

- For example, assuming the user previously entered "Cat":

<p style="color:green; text-align:center;">Delete "Cat"?   (y|n)</p>

- If the user enters 'n', do nothing (i.e. you'll go back to asking the user to enter an animal name)
- If the user enters 'y' delete the mammal's data from your lists.

• If the user entered 'q' you should:

- use the function you wrote for part 2 above to write the data out to a new file called "BrainBodyWeightPounds.csv" and your program will end. If at any point you accidentally open this file with Excel, make sure you close it, because Excel will put a lock on it and you will not be able to write to it again until you close it.

5.) Verify that your documentation makes sense and that you've added documentation to each of your functions.

6.) **Verify that your program works**

7.) Upload your file to the Program 8 dropbox folder on D2L

1.2) RUNNING EXAMPLE

Below is an example of what you might see in a terminal when running this program (output is in green, user input is in turquoise):

Enter animal name (or "q" to quit): Bugbear

File does not contain "Bugbear".

Add "Bugbear" to file? (y|n) y

Enter body weight for "Bugbear" in kilograms: 1000

Enter brain weight for "Bugbear" in grams: 1

Enter animal name (or "q" to quit): Goat

Goat: body = 27.66, brain = 115.0

Delete "Goat"? (y|n) n

Enter animal name (or "q" to quit): Chickenshrimp

File does not contain "Chickenshrimp".

Add "Chickenshrimp" to file? (y|n) n

Enter animal name (or "q" to quit): Bugbear

Bugbear: body = 1000.0, brain = 1.0

Delete "Bugbear"? (y|n) y

Enter animal name (or "q" to quit): q