ISA 281, Business Programming: Homework 5 (25 Points)

Due on: April 17, 2019, Wednesday, 11:59 PM

Chaitanya "Chai" Sambhara

Q1. (5 points)

Extract all the numbers in the file and compute the sum of the numbers. You **MUST** use regular expression to extract the numbers. You are given two files, one small file called sample.txt to test the accuracy of your code and one large file called regex_large.txt that you will use to work on this question. Make sure you place the text files in the same folder as your python file. The numbers can appear anywhere in the line. There can be any number of numbers in each line (including none). You should use re.findall() function. The sum for the small file is 25238. Remember, your code should compute the sum of numbers in the large file regex_large.txt. No credit will be given if your code computes the sum of the sample.txt file and not the regex_large.txt file. No credit will be given if you do not use regular expressions.

Q2. (10 points)
Consider the text file long.txt. Although you can open it in notepad or WordPad on a PC, I recommend opening it with WordPad. On a Mac, you can open the file with TextEdit. Write a program in Python that opens the file long.txt, reads through the file looking for lines that start with "Expected_Probability:" and look something like this:

```
Expected_Probability: 0.8475
```

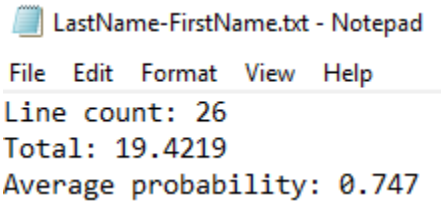
This file has many such single lines that look like it. Count how many such lines exist in the file. Extract the floating point values (for example, the floating point value in the example is 0.8475) from each line and compute the sum (i.e., total) and average of those values. Do not use the inbuilt Python function sum() or any variable called sum. You should show the output on the screen and write the output in a new file called "LastName-FirstName.txt". I have also provided you a file called "short.txt" that you can use to verify the accuracy of your results. The output for short.txt is shown below but please remember that the homework will be graded for long.txt instead. Round the results to 4 digits. Make sure your code is for long.txt. Will deduct 5 points if I must fix your code to change short.txt to long.txt.

Hint: Consider only those lines that start with the string "`Expected_Probability:`". Use string extraction techniques or regular expressions to extract the floating-point values. Remember to write code to close the files (file that you read and the file that you created). Remember, you can only write strings to a file. You cannot write integers or floating-point numbers. You may or may not chose to use regular expressions for this question.

Output for short.txt below:

```
Line count:  26
Total:  19.4219
Average probability:  0.747
Data written to LastName-FirstName.txt
```

Data written to file LastName-FirstName.txt in case of short.txt is shown below:



Q3. (10 points)
Write a program to read through the file long.txt and find who has sent the most number of email messages. Look for "From: " lines to extract emails. You must create a dictionary and map the sender's email address to the number of times the email address appears in the file. After the completion of your dictionary, loop through it to find out the most prolific email sender (i.e., the key with the highest value) and the number of emails sent by him/her.

You can check the accuracy of your code from the file short.txt. The most prolific email sender and the number of emails he/she sent based on short.txt is listed below:
cwen@iupui.edu 5

Submission Instructions (Must read carefully and follow instructions):

1. You will submit only python files (extension .py). Name your files in the following format:
   LastName-FirstName-HW5-Q1.py
   LastName-FirstName-HW5-Q2.py
   Example: Sambhara-Chai-HW2-Q2.py
2. You will get zero points for a question if the python file for that question has an error of any kind. Even if it is a minor error, if your code does not run, you do not get any credit.
3. You are **NOT** allowed to collaborate with any student on this homework.
4. Do not submit any files of any other format. Submit only python files.
5. You can make comments in the file. Non-code text should be commented out
6. You must include your identifier on the first line of each python file
   a. Example: The first line of your python file will look something like this
      #Sambhara-Chai-HW5
   b. 2 points will be deducted for failure to include your identifier in the python file.