CSCI A201/A597 - Introduction to Programming I

Fall 2017

Homework 13

Due Date

- Thursday, November 16, 11:59pm on IU Canvas.

Work Policy

Homework for A201/A597 has to be completed individually: no group solutions or cooperative work. You may discuss the homework with other A201/A597 students, but you have to write the solutions and implement programming answers by yourself. If you discuss the homework with other A201/A597 students (or anyone else, for that matter!), you must acknowledge them by including their names in a comment at the top of the Python files you turn in for this homework. **Do not share any source code or homework answers with any students**.

Homework Tasks [total: 100 points]

Topics: refer to the second part in this week's Lecture 13 notes, and Reading Assignment 13, about working with files in Python. When you've completed and run your programming tasks on IDLE, save a transcript of your IDLE session to a file named `hw13-transcript.py`, and include that with your submission. Make sure that your submitted IDLE transcript includes interactions with running *all* of your homework scripts, otherwise your homework will be rejected.

A. ***Written Questions***
[40 points]
In answering these questions, it'll be helpful to read the material listed in parts A. and B. this week's Reading Assignment 13.

1. What are some of the most common Python file access modes?
(name at least two distinct file access modes in Python, and explain their difference)
[6 points]

2. What happens if your Python program tries to read past the end of a text file?
[7 points]

3. Why is it important that a Python program closes a file when it's done working with it?
(provide at least two distinct valid reasons for closing a file after working with it)
[6 points]

4. What is the difference between `read()` and `readline()` in Python?
[7 points]

5. How can you read the entire content of a text file with just one method call?
(provide an example of such a method call, i.e. a one-line Python script that would read the entire content of a text file that has been correctly opened)
[7 points]

6. How can you use a `for` loop to read the entire content of a text file, and how does this differ from the method at the previous question (5.)?
   [7 points]

Write your answers to the above questions in a plain-text file named `hw13-answers.txt`.
Make sure to number your answers appropriately, so that we know which answer belongs to which question.

B. [60 points]
For Task B you have to implement the functions listed below.

For Homework 13, it is *required* to include properly formatted docstrings with purpose statements and signatures for all functions you define. If you don't include properly formatted docstrings with purpose statements and signatures for all functions you define for Homework 13, your homework will be rejected.

1. [30 points]
   Write a script called `hw13-fileprinter.py` that asks the user to input the name of a file, opens the file, reads its contents, prints the contents of the file on the screen, and closes the file. If no such file exists, it's okay if the script crashes. You may organize this your into function(s), or not, as you see fit: either choice is fine for Task B.1.

2. [30 points]
   Define the function `dict_from_file()` which takes a string representing a filename, opens the file, reads its contents, closes the file*, and returns a dictionary based on the letter frequencies in the given file. If no such file exists, it's okay if the function crashes. So for example, if the file `poem.txt` was in the same directory as `hw13-filehistogram.py` and had the following contents:

   ```
   I have never seen a purple cow,
   And I never hope to see one.
   But I can tell you anyhow,
   I'd rather see than be one.
   ```

   Then `dict_from_file("poem.txt")` would return the following dictionary: `{'r': 5, 'w': 2, 't': 5, ',': 2, 'u': 3, 'o': 7, 'B': 1, 'b': 1, 'p': 3, 's': 3, 'd': 2, 'l': 3, '\n': 4, 'A': 1, 'h': 5, '.': 2, 'n': 9, 'a': 6, 'v': 3, 'y': 2, ' ': 22, 'I': 4, "'": 1, 'c': 2, 'e': 18}`.

   ***Note*: remember to close the file *before* returning the dictionary. If you don't, you won't be able to catch the bug during testing, so you'll have to check your code by hand to make sure that the file gets closed.

   Hint: you may review Task B in Programming Assignment 12 -- you're welcome to reuse any code that **you wrote** to solve that task, in this week's Homework 13: if you do so, you also need to provide a reference to your previously written code, in a Python comment at the top of the solution file you turn in for Homework 13.

   Put all your code for Task B.2 into a Python script named `hw13-filehistogram.py` and attach that to your Canvas submission.

Rejection Warning: any function you define that *displays* a value (using `print()`) when it was supposed to *return* it (or vice versa) will be rejected (meaning you will receive zero points for that problem, and you will have to redo the assignment to get any credit for that problem). Similarly, any function you write that gets *user* input (using `input()`) instead of taking arguments (or vice versa) will also be rejected.

Homework 13 Submission Instructions:

1. Your submission has to contain the files as listed above:
   `hw13-transcript.py`,
   `hw13-answers.txt`,
   `hw13-fileprinter.py`, and
   `hw13-filehistogram.py`.
    1. At the top of your files, include "*A201 / Fall 2017*" (or "*A597 / Fall 2017*"), "*Homework 13*", your *full name* and your *IU username* (i.e. your IU account) *in* the homework text. Note: in your Python files (file extension .py), kindly include this information at the top of the file, within Python `# comments`.
    2. Make sure that your plain-text documents (including all your `.py` files) are all clearly readable by anyone; plain-text documents need to use either:
        - ASCII encoding (7-bit, allows no accents/smart quotes/etc.), ← preferred for Python source code submissions that only contain English-alphabet characters & symbols.
        - UTF-8 encoding (capable of encoding all sorts of characters) ← only for any submissions that *require* non-English-alphabet characters & symbols.

       (...otherwise your output may end up garbled...)
    3. Do not use any other document file formats: submitted files *other* than plain text (file formats such as .pdf, .doc, .docx, .html, .xml, etc...) will be *rejected*. In other words, all submitted `.txt` and `.py` files need to be plain text.
2. Turn in your Homework 13 files by 11:59PM on Thursday, November 16 2017, on the IU Canvas *A201/A597 Fall 2017* page.
3. When you turn in your homework on IU Canvas, it is your responsibility to verify that your files have been uploaded, and that their content is visible on the IU Canvas server.
4. Please do not omit your IU username from the content.

Last updated: November 15, 2017
mitja⊛indiana·edu