CSCI 1133, Fall 2019
Programming Assignment 8
Due: 11:55pm, Wednesday October 30, 2019

Due Date: Submit your solutions to GitHub by 11:55 p.m., Wednesday, October 30th. We will do a pull from this time point. Do not upload anything to Canvas and PLEASE be sure to use proper naming conventions for the file, classes, and functions. We will NOT change anything to run it using our scripts.

Unlike the computer lab exercises, this is not a collaborative assignment. You must design, implement, and test your code on your own without the assistance of anyone other than the course instructor or TAs. In addition, you may not include solutions or portions of solutions obtained from any source other than those provided in class (so you are ONLY allowed to reuse examples from the textbook, lectures, or code you and your partner write to solve lab problems). Otherwise obtaining or providing solutions to any homework problem for this class is considered Academic Misconduct. See the syllabus and read section "Academic Dishonesty" for information concerning cheating. Always feel free to ask the instructor or the TAs if you are unsure of something. They will be more than glad to answer any questions that you have. We want you to be successful and learn so give us the chance to help you.

Instructions: This assignment consists of 1 problem, worth 40 points. Solve the problem below by yourself, and put all functions in a single file called `hw8.py`. Use the signatures given for each class and function. We will be calling your functions with our test cases so you must use the information provided. If you have questions, ask!

Because your homework file is submitted and tested electronically, the following are very important:

- You follow all naming conventions mentioned in this homework description.
- You submit the correct file, hw8.py, through Github by the due date deadline.
- You follow the example input and output formats shown.
- Regardless of how or where you develop your solutions, your programs should execute using the python3 command on CSELabs computers running the Linux operating system.

Push your work into Github under your own repo. The specific hosting directory should be: repo-<username>/hw8, where you replace <username> with your U of M user name. For instance, if your email address is bondx007@umn.edu, you should push your hw8 to this directory: repo-bondx007/hw8

The following will result in a score reduction equal to a percentage of the total possible points:

- Incorrectly named/submitted source file, functions, or classes (20%)
- Constraints not followed (40%)
- Failure to execute due to syntax errors (30%)

Problem A. (40 *points*) **Hacking the Gradebook**

This problem is too hard. It's so hard that you're hallucinating these words in place of the actual problem description (which you're pretty sure was written in Klingon). You have only one chance: you must use your File I/O skills to write a series of functions to edit the gradebook CSV file and give yourself full credit.

Luckily, you overheard the TAs discussing how the gradebook is formatted, so you know the following:

- The file you want to access is in the same folder that your program will be run in.
- The first row(i.e. the first line) of the CSV file contains the column titles.
- The file has columns called `'Full Name'`, and `'hw8 Grade'`, along with others. You do not know what order these columns appear in nor how many other columns are in the file.
- The entry you want to change will be in the row that contains your full name as it appears on Canvas, and the column titled `'hw8 Grade'`.
- You do not know how the students in the class are ordered in the spreadsheet, so you can't figure out what number row your entry will be in ahead of time.
- Your full name will only appear in the row corresponding to your grades.
- The hw8 Grade column is not the farthest to the right, so there's no need to check for `'\n'`.
- There are no commas in the CSV file entries themselves; commas are only used to separate values.
- This problem is out of 40 points, so you need to find the row containing your name and change the score in that row's 'hw8 Grade' column to 40.

In order to accomplish your mission, download the top-secret template `hw8.py` file and CSV files from Canvas (They are all in the compressed folder `hw8.zip`). One of the TAs accidentally uploaded a useful outline for your program. You must write four functions by following the instructions left for you in the comments. Do NOT change the function names.

Hints:

- Don't try to do this all at once. Make sure to test each function before moving to the next one.
- Parts 2 and 3 will require turning rows of a CSV file into lists to access the columns.
- You can turn a line of a CSV file into a list with the statement `listName = stringName.split(',')`.
- You can turn this list back into a CSV string with the statement `stringName = ','.join(listName)`.
- Until you're certain that your `haxx` function is working correctly, you should avoid overwriting the original grade files, as redownloading them every time you make a mistake will get annoying. Temporarily write to a dummy file ('dummy.csv') until you know for certain that `haxx` does what you want it to.
- Please do not take the contents of the randomly generated CSV files seriously.

Constraints:

- Do not import/use any Python modules, <u>especially the csv module.</u>

- You may use any string or list method that is appropriate to solving this problem.
- All files must be properly opened and closed.
- You must not change the function names.
- You must use the provided function arguments.
- Don't use the input() function, as this will break our grading scripts.
- Follow the instructions in the `hw8.py` template.
- Your submission should have no code outside of the function definitions (comments are fine).
- You are not required to push any of the CSV files to github: only the `hw8.py` file will be graded.

Examples (assuming that you are running this in the same folder containing the example CSV files that you downloaded from Canvas, and that the file `'fakefile.notreal'` doesn't exist):

```
>>> get_data_list('smallfile.csv')
['This,file,exists,\n', 'but,does,not,contain\n', 'hw8
Grade,as,a,\n', 'column,title,so,haxx\n', 'should,return,FALSE,\n']
>>> get_data_list('fakefile.notreal')
-1

>>> hw8_index("hw1 Grade,hw2 Grade,hw3 Grade,hw4 Grade,hw5 Grade,hw6
Grade,hw7 Grade,hw8 Grade,hw9 Grade\n")
7
>>> hw8_index("HW8 grade,HW8 Graid,hw8 Grade,HW 8grade,Homework8
grade\n")
2
>>> hw8_index('This,file,exists,,\n')
-1

>>> alter_grade("30,20,0,0,\n",2)
'30,20,40,0,\n'
>>> alter_grade("5,2,5,6,0,0,minutes,\n",1)
'5,40,5,6,0,0,minutes,\n'
>>> alter_grade("Bulbasaur,Charmander,Squirtle,Weedle,Pidgey\n",3)
'Bulbasaur,Charmander,Squirtle,40,Pidgey\n'
```

See Canvas for examples of how the files would look after these calls to `haxx` if your name was 'Joe Bob'. Keep in mind that your name is <u>not</u> Joe Bob, so your program should be editing your own row, not Joe Bob's. The examples are in the folder named:
`hw8_template_examples_Joe_Bob`

```
>>> haxx('fakefile.notreal')
False
>>> haxx('smallfile.csv')
False
```

```
>>> haxx('grades1.csv')
True
>>> haxx('grades2.csv')
True
>>> haxx('grades3.csv')
True
```