Computer Science CS134 (Fall 2020)
Daniel Aalberts, Duane Bailey, & Molly Feldman
Laboratory 09
Trivia! (due Thursday at 5pm)

Objective. To analyze a CSV database.

This week we're going to take a closer look at the faculty of Williams. We have a (mostly) up-to-date set of data about the faculty, presented here as a *comma separated values* (CSV) file.[1]

You will build two scripts: `faculty.py` and `trivia.py`. In `faculty.py` you will complete a small collection of methods to help organize the data. Then, in `trivia.py` you'll write a Python program to answer a number of trivia questions.

Getting Started. As usual, you should clone the starter repository for this week's lab into your `cs134` directory:

```
git clone https://evolene.cs.williams.edu/cs134-labs/22xyz3/lab09.git ~/cs134/lab09
```

where your CS username replaces `22xyz3`. The resulting directory contains a CSV file that describes the faculty, two Python starter files (`faculty.py` and `trivia.py`), and a plain text file (`trivia.txt`) for submitting trivia answers.

Reading CSV files.
As we've discussed, Python provides a module `csv` that helps read and write CSV files. We will use this module throughout this lab to analyze the faculty database. Remember you can type `pydoc3 csv` for more details about the `csv` module.

This week's tasks. We have two Python scripts that need to be fleshed out. The first is a module, `faculty`, that contains a pair of classes—`Degree` and `Instructor`—and a few tools that will help you build a database of faculty. Your job is to complete `faculty.py` as outlined by the steps below. The second script—`trivia.py`—will contain routines that you use to compute the answers to the questions in `trivia.txt`.

First, let's complete some functions we believe will be helpful as we build our database:

1. The `Degree` class's `__init__(self,info)` method. This method takes a string (`info`) and breaks it down into the components that describe a degree: a year (an integer), a degree "kind" (a string), and a granting institution (a string).

   ```
   >>> d = Degree('2000, B.A., Williams College')
   >>> d.year
   2000
   >>> d.kind
   'B.A.'
   >>> d.institution
   'Williams College'
   ```

   Remember to strip any leading or trailing spaces from any strings you build.

[1] We apologize in advance for any mistakes regarding your favorite faculty member; web scrapping and data cleaning, as you may encounter here and in the future, can be an error-prone task.

2. The Instructor class's __init__(self,name,dept,title,degrees). Given the name, home department, and title (all strings) as well as a list of Degree objects, save these items in the state appropriate state variables for the Instructor.

```
>>> d1 = Degree("1987, B.A., Bowdoin College")
>>> d2 = Degree("1993, Ph.D., Wesleyan University")
>>> i = Instructor("Robert M. Savage","Biology","Professor of Biology", [d1, d2])
>>> i.name
'Robert M. Savage'
>>> i.dept
'Biology'
>>> i.title
'Professor of Biology'
>>> isinstance(i.degrees[0], Degree)
True
```

3. readDB(). Reads the database file faculty.csv and returns a list of Instructor objects.

4. uniqCount(itemList). Given a list of values, possibly containing adjacent duplicates. The uniqCount function returns a list of pairs, [value,count]. Each pair corresponds to a value that occurs in the list and the number of times it occurs, adjacently. For example:

```
>>> uniqCount([1988,1988,1988,1989,1990,1990,1991,1993,1993,1988])
[[1988,3],[1989,1],[1990,2],[1991,1],[1993,2],[1988,1]]
```

After building the toolbox in the faculty module, you should turn your attention to completing some of the trivia problems in trivia.py. You must answer **4 trivia questions in total** and **at least 1** of the questions must be Q5, Q6, or Q7. Make sure to consider how to use the functions in faculty.py in your trivia solutions.

Each of your solutions to the trivia questions should be functions which print out the problem's answer, as shown in the starter code. For example:

```
def q1(fac):
    """
    1. How many members are on the faculty at Williams?
    """
    print("Q1: There are {} faculty.".format(YOUR ANSWER))
```

As you solve the trivia questions, copy the answers into trivia.txt.

Submitting your work.

When you're finished, add, commit, and push your work to the server as you did in previous labs. Remember that you must certify that your work is your own, by signing the honorcode.txt file, committing and pushing it along with your work. Make sure you have: (a) implemented all of the methods asked for in faculty.py and (b) answered at least 4 trivia questions, one or more of which is Q5, Q6, or Q7.

Extra credit. Work on more than 3 of the trivia questions or invent additional questions and find the answer!

⋆