22C:1110 Final Project: *Quiz Administrator*

Due Friday, May 6, 2016 by 11:30PM
(Late submissions will not be accepted)

For this project, you will get some more practice working with files and data. Suppose that you are given a file containing questions for a multiple-choice quiz. The data is organized in the following manner:

- The first two lines contain information about the quiz:
  - Number of questions (this will be an integer, k)
  - Number of choices for each question (this will be an integer, n)

- The rest of the file contains the exam questions:
  - Question statement
  - Choice 1
  - .....
  - Choice n
  - Correct answer choice (an integer between 1 and n)

Here is an example of the contents of such a file (which may be useful in testing your program):

```
3
5
How many states are there in the United States?
30
49
50
51
52
3
Which of the following cities is the capital of Iowa?
Springfield
Tallahassee
Cedar Rapids
Cedar Falls
Des Moines
5
Which of the following television shows had a character named “Hawkeye”?
The Brady Bunch
M*A*S*H
Doctor Who
The X Files
Lost in Space
2
```

Notice that in this example, there are 3 questions, every question has 5 choices, and only one of the choices is correct. Your job is to write a python program, called administerQuiz(filename), that takes, as an argument, the name of a file containing a quiz in the previously described format. It will open the file and administer the quiz. It will instruct the user on how to answer the questions (do they enter the letter corresponding to their choice? A number? What are the limitations of using letters?) It will use a loop to do the following:

- ask the user the question,
- present the answer choices in whatever format matches the instructions,
- get the user input from the keyboard,
- determine whether or not the user answered the question correctly,
- give feedback to the user after every question.
    - The feedback for each question should be something simple, like "Correct!" or "Incorrect!"...you do not have to use these words, but you must let the user know whether or not they answered the question correctly.

At the end of the quiz, the program should **return** a summary of the results (not print…this will signify the end of the program), with a description that indicates how well they did. Something along the lines of "You answered ____ questions correctly out of ____ questions. Great job!" if the person was in Group 1. There will be 5 categories for the user evaluation, based on the percentage answered correctly according to the following ranges:

Group 1 = A: [90%, 100%] correct
Group 2 = B: [75%, 90%) correct
Group 3 = C: [60%, 75%) correct
Group 4 = D: [45%, 60%) correct
Group 5 = F: [0%, 45%) correct

You should come up with the feedback, like "Good job!" or whatever you decide to use. Make sure that the user understands that if they are in the lower grade scales, they did not do well!
The grading rubric for this project is:

- The program runs without errors
- The program can open, read, and close a file
- The program loops through the questions
- The program keeps track of whether the user answered the question correctly or not
- The program provides feedback after each question is answered
- The program provides a set of instructions for how to answer the questions
- The program presents a summary of their performance based on the groupings specified

The projects are to be worked on individually. You may discuss your ideas for writing your program with the TAs and with your instructor, but you may not discuss this project with other students or tutors. Questions related to this project should be directed primarily to Chris, and he will be scheduling office hours to specifically discuss this project. You may not ask for suggestions through discussion boards. You may not request help on this project through online discussion boards. Any violation of this policy will be considered to be an academic integrity violation. You may use

resources (python.org, etc…) but you must cite your sources and provide the URL of any website visited. Start this project early, and ask questions as they come up. Be sure to comment your code, and start with an outline of your thought process. Submit your project to the Dropbox folder: Project: Quiz (through ICON).