

Homework #5 Introduction to Computing

Due: Oct 31, 2009 (Saturday at 11:59 PM)

Chapter 7 Programming Challenge 12 - Exam Grader (text below)

One of your professors has asked you to write a program to grade her final exams, which consist of only 20 multiple-choice questions. Each question has one of four possible answers: A, B, C, or D. The file `CorrectAnswers.txt`, which is on the Student CD, contains the correct answers for all of the questions, each answer written on a separate line. The first line contains the answer to the first question, the second line contains the answer to the second question, and so forth.

Write a program that reads the contents of the `CorrectAnswers.txt` file into a one-dimensional char array, and then reads the contents of another file, containing a student's answers, into a second char array. The Student CD has a file named `StudentAnswers.txt` that you can use for testing purposes. The program should determine the number of questions that the student missed, and then display the following:

- A list of the questions missed by the student, showing the correct answer and the incorrect answer provided by the student for each missed question
- The total number of questions missed
- The percentage of questions answered correctly. This can be calculated as

$$\text{Correctly Answered Questions} \div \text{Total Number of Questions}$$

- If the percentage of correctly answered questions is 70% or greater, the program should indicate that the student passed the exam. Otherwise, it should indicate that the student failed the exam.

Additional Requirements:

Be sure to divide the program into functions that perform each major task.

When you write your program, be sure to use general conventions of good style:

- use meaningful variable names with good style, i.e., use CamelCase (or use underscores) (I like to declare them one per line with a following comment if necessary)
- use meaningful named constants (e.g., `PI`, `STATE_SALES_TAX`) where appropriate with good style (`ALL_CAPS_AND_UNDERSCORES`). Put your global constants where they can be found and changed easily in future versions of your program, e.g., after your initial comments describing the program and before your main function definition.
- use comments at the start of the program, before each function, and before any especially difficult section of code to understand (I like to label the closing set bracket, `}`, with some indication of what's being closed)
- place the main function near the top of the program with user-defined functions below it
- use *white space* (spaces, indentation, blank lines) to make you program more readable by:
 - aligning the opening set/curly brace, `{`, with the corresponding closing `}` one (I like to put the `{` on the same line with the programming construct (e.g., main function definition) with the closing `}` aligned with the start of the construct)
 - indent all the lines inside a set of of braces
 - you blank lines to separate logical units of the code, e.g., between variable declarations and executable statements

The data files for the homework #5 are at:

<http://www.cs.uni.edu/~fienup/cs051f09/homework/CorrectAnswers.txt>

<http://www.cs.uni.edu/~fienup/cs051f09/homework/StudentAnswers.txt>

Submit your homework electronically at http://math-cs.cns.uni.edu/~schafer/submit/which_course.cgi

The steps for the homework submission system are:

1. Zip your project folder, say hw5, by right-clicking on it and selecting Send To | Compressed Zipped Folder which will create a file called hw5.zip. This hw5.zip file is what you want to submit via the on-line submission system.
2. Log on to the submission system at: http://www.cs.uni.edu/~schafer/submit/which_course.cgi
Use the same AD-ITS User name and password you use to log on the lab computers.
3. Select the course and section number of "810:051, Intro to Computer, Fienup". Click the "Continue" button.
4. Select the homework that you wish to submit: "HW 5 Exam Grader". Click the "Continue" button.
5. Specify how many extra files you want to submit. Just leave it at 0. Click the "Continue" button.
6. Upload the hw5.zip file containing your project by Browsing and selecting hw5.zip. Click the "Continue" button.
7. The next page reports on the status of the upload(s). You can always continue to upload a newer version until the deadline. A newer file of the same name will replacing the older one.