CS210 – Machine Problem 7

Endless Hoards of Zombies
30 Points
Assigned: April 10, 11, 2017
Due: April 24, 25, 2017

One of the most obvious signs of zombism is a marked decrease in intellectual ability. Since society has provided us with an ongoing measure of mental function (at least for the younger years), we will automate the monitoring of this measurement.

Implement a C grade book program. This program must do the following tasks:

1. Create a struct `student_t` that will hold all the information for a given student. That struct should have at least the following features:
    - A `name` variable holding the student's name. The maximum length of names is 50 characters.
    - An `id` variable for holding the student's id.
    - An array of type double that holds each individual score for the student. The maximum number of grades for each student is 20.
    - The number of grades stored in the grade array.
2. Print a greeting to show what this program is.
3. Ask the user for a filename to input.
4. Read the student data from the file
5. The data file will start with the number of students in the file. Then it will have an id, a name (on a single line), and a list of grades, terminated with a -99 for each student.
6. You should create a *dynamic* array that is exactly the right size to hold the students in the file.
7. Sort the array in ascending order by student name.
8. Output, on the screen, the student's name, id, their grade average, and their letter grade based on the 90-80-70-60 scale. (90=A, 80=B etc.) and a list of all their grades on the line.
9. Since the array is dynamically-allocated, you need to make sure that you free up the space when you are done with it.

Warning: We will throw a *really* big data file at it when we test it, so don't build any limits in.

Notes

- When you read in the student id, use the specifier `"%d\n"`. This will make it skip the newline character after the id.
- Read the name with the `fgets` function. One issue with this function is that it will read in the trailing `\n` character. You will need to remove it from the name after you've read it.
- The column widths for the output are: 20 for the name, 4 for the ID, 7 for the average, 5 for the grade and 3 for each of the scores. The columns are separated by a single space character.

Suggested Functions

You are expected to use good functional decomposition when designing your program. Here are some suggestions:

- A function that takes an already open file, an array of `student_t`, and a `count` and reads `count` students from the file into the array.
- A function that takes an open file, reads a single `student_t` from it and returns it.
- A function that takes a single `student_t` and returns the average grade
- A function that given an array of `student_t` and the number of elements in the array, sorts the array by name.
- A function that given two references to `student_t`, swaps them.
- A function that given an array of `student_t` and the number of elements, prints the entire report.

Sample Run

```
Welcome to the automated grade book program.
Enter student data filename:test.txt
Unable to open file test.txt. Try again.
Enter student data filename:grades2.txt
Name                 ID   Average Grade Scores
Barnes, Kristy       6642    78.2     C  66  77  82  84  82
Bickerstaff, Violet  5342    98.8     A 100  99  98  97 100
Carla                7421    75.2     C  88  34  89  92  73
Carosi, Stacey       8753    86.0     B  88  99 100  78  65
Coleman, Nikki       1234    84.2     B  80  90  92  87  79  77
Kapowski, Kelly      2442    92.2     A  87  99 100  87  88
Morris, Zack         8742    67.6     D  77  65  50  80  66
Ox                   1001    11.4     F  12  11  34   0   0
Parks, Wendy         8111    75.8     C  88  60  76  80  75
Peterson, Joanna     7723    74.0     C  70  78
Powers, Screech      9642   100.0     A 100 100 100 100 100
Scott, Tori          3412    63.8     D  56  77  60  62
Slater, A.C.         1254    95.0     A  90 100
Spano, Jesse         7625    73.7     C  88  85  92  78  76  23
Turtle, Lisa         9199    85.2     B  88  97  77  76  88
```