

# CS 421 – Fall 2017

## Project #0

**50 Points**

**Due: Monday, 09/25/17 at 11:59PM in D2L**

The objective of this assignment is to help you get familiar with basic C, text and binary file processing and using `structs`; you will perform file processing to compute the grades for students in a course.

The grading scheme for a course is described by the following template:

<u>Category</u>	<u>Category percent</u>	<u>#items in category</u>	<u>Points/item in category</u>
Quizzes	10	3	10
Labs	20	2	20
Assignments	30	1	50
Exams	40	2	100

The first row is a set of column header names describing the different assessment item categories (e.g. quizzes, exams, etc), and for each category, the total percent this category comprises of the final overall grade (e.g. quizzes will account for 10% of your total grade), the total number of items in that category (e.g. there were three quizzes), and the total number of points per item in that category (e.g. each quiz is worth 10 points).

This grading scheme is used in conjunction with a data file to compute the grades. The data file is a binary file of student data that has the scores for each data item, for each student. The file is composed of multiple records, with each record having the following format:

`id(4-int)First(20-char)Last(20-char)Q1(4-int)Q2(4-int)Q3(4-int)L1(4-int)L2(4-int)A1(4-int)E1(4-int)E2(4-int)`

So each record has a size of 76 bytes.

Your program must process the data file and generate an output CSV file with all of the data in the input file, along with the following additional data:

- The 'Grade' that computes the final grade, out of 100, for each student, based on the weights for each item.
- In addition, there are four new data rows, at the end of the file, as follows: the minimum score, maximum score, average score and standard deviation for each numeric column.

Here is a sample output file:

```
Id,First Name,Last Name,Quiz1,Quiz2,Quiz3,Lab1,Lab2,Assignment1,Exam1,Exam2,Grade
58136,Wade,Andrews,5,8,2,2,20,47,20,40,56.20
59759,Jacqueline,Simmons,9,4,1,16,13,2,82,41,44.97
28056,Frederick,Castro,8,0,6,3,6,29,53,63,49.77
15532,Annette,Myers,2,6,0,14,13,4,77,43,42.57
,,Min,2,0,0,2,6,2,20,40,42.57
```

```

,,Max,9,8,6,16,20,47,82,63,56.2
,,Avg,6.00,4.50,2.25,8.75,13.00,20.50,58.00,46.75,48.38
,,Std Dev,3.16,3.42,2.63,7.27,5.72,21.52,28.32,10.90,6.01

```

Please note the following:

- a) You should write a C program, that runs on the lab Linux machines, to implement this project. Your program should read the input and output file names as command line arguments, *in that order*.
- b) You should define a `struct` to capture a student record, and dynamically allocate space for an array of such `struct`. You will need to figure out how many records are in the input file. (*Hint*: you can use the calls `fseek`, `ftell` and `rewind` on your file pointer.) Read the input file data into this array and do your processing on the array. The output file is generated from the contents of this processed array.
- c) Use the standard algorithm to compute the grade for a student: compute a category percent by summing up the scores for all items in a category, divide that by the sum of the maximum score for all items in that category and multiply that by the percent for that category. Sum up all category percents to get the final grade.
- d) Use the following formula to compute the standard deviation:
$$\sqrt{\frac{(x_1 - avg)^2 + (x_2 - avg)^2 + \dots + (x_N - avg)^2}{N - 1}}$$
- e) Please break up your code into multiple methods, and split the methods among at least two C files. Document each method with sufficient comments.
- f) Provide a make file, called `makefile`, that builds your project.
- g) Sample input and output files are available in the zip file `P0.zip` on D2L.

## Submission

To complete this assignment, simply submit your C code files and your make file, zipped up in a single folder, to the D2L submission dropbox folder **Project 0** by the deadline specified.