Assignment 7

Due on: October 27^{th}, Sunday 6pm
No interview grading

Task 1: Working with data files in MATLAB

Large amounts of data can become useful if the information is organized and presented visually. Following are the goals of this assignment:

1. Read in data files. Use the MATLAB built-in function `xlsread` to read in an Excel spreadsheet. Perform some initial cleaning on the imported data, and then save the cleaned data in MATLAB's native data format. (Related links: xlsread, save)
   Note: As you can see from this link (xlsread), MathWorks do not recommend to use `xlsread`. Therefore, we also accept to use the MATLAB built-in function `readcell`. In other words, you can choose the function `xlsread` or `readcell`.
2. Write a program that loads the cleaned data and uses it to calculate the grade for each student. (Related link: load)

Note that you should end up with 2 separate m-files to meet the objectives of this problem.

During the Fall 2015 semester, 101 CU Boulder students took a quiz. The quiz consisted of 17 multiple-choice questions, and 1 essay question. Your task is to compute the final quiz scores and to plot a histogram of the scores.

What is given?

You are given one file named *quiz_results.xlsx* which contains (see figure below):

I. The answer key for the multiple-choice part of the quiz (cells G2:W2)

II. The student ID numbers (column B)

III. The score the students received for the Essay portion of the quiz (column E)

IV. The section number (column F)

V. The student answers to each of the 17 multiple-choice questions (column G through W)

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
1		Student ID No.	Score	# Multiple-choice	Essay	Section	Q 1	Q 2	Q 3	Q 4	Q 5	Q 6	Q 7	Q 8	Q 9	Q 10	Q 11	Q 12	Q 13	Q 14	Q 15	Q 16	Q 17
2	Key		100	17	15		D	B	C	B	A	C	B	D	D	B	B	A	B	B	A	D	C
3		CSCI1320000'3			0	9	A	A	B	C	C	B	A	B	A	B	B	A	B	B	A	B	B
4		CSCI1320000'4			0	9	D	A	D	C	A	A	D	B	A	B	A	B	A	B	B	D	B
5		CSCI1320000'17			2	9	D	B	B	A	C	C	A	B	C	B	A	A	B	A	B	D	C
6		CSCI1320000'18			2	9	D	B	B	B	A	B	A	D	D	B	A	B	A	A	B	D	B
7		CSCI1320000'59			7	9	A	A	B	C	A	A	A	C	C	B	A	B	A	A	B	B	D
8		CSCI1320000'60			7	9	D	B	B	B	A	C	B	D	D	D	B	B	B	B	A	D	C
9		CSCI1320000'61			7	9	C	A	C	C	A	B	D	B	C	B	A	B	A	B	B	B	D
10		CSCI1320000'90			9	9	D	B	C	D	A	E	B	D	D	B	A	B	B	B	A	D	C
11		CSCI1320000'91			9	9	D	A	C	C	A	B	D	C	D	A	A	A	B	B	B	B	C
12		CSCI1320000'92			9	9	D	B	C	B	A	C	A	D	D	D	B	A	B	B	A	D	C
13		CSCI1320000'114			10	9	D	B	C	B	A	C	B	D	D	B	A	B	A	B	A	D	C
14		CSCI1320000'115			10	9	D	A	D	C	A	B	D	B	B	B	A	B	B	B	B	B	D

Figure 1: Part of *quiz_results.xlsx*

What you need to do:

1. **Read in the spreadsheet and clean up the data.**

   Make a script file named *task1_read.m*. In the script,

   (a) Read in the data using the MATLAB built-in function `xlsread` or `readcell`.

   (b) We only need the I, II, III, and V parts of data. We do not need to keep any of the column titles. Create separate cell arrays that contain only those parts of the data. Even though you are separating your data into multiple arrays, make sure the row numbers will still correspond to the correct student ID numbers, the answer key column numbers correspond to the correct question numbers, etc.. (*You do not need to come up with an algorithm for detecting where the real data starts/ends. You can hard code these indices.*)
   Hint: **If you use the function** `xlsread`, use the syntax [nums,txt,raw] = xlsread(filename);.
   Then, get the I, II, and V parts of data from txt, and get the III part of data from nums.
   If you use the function `readcell`, use the syntax M = readcell(filename);.
   Then get the I, II, III, and V parts of data from M. Additionally, use the function `cell2mat` to get the III part of data. (related link: link)

   (c) Save **only** the data that is needed in the MATLAB native file format.

2. **Load cleaned data and calculate grades.**

   Make a script file named *task1_load.m*. In the script,

   (a) Read in the MATLAB data file you created in previous part. (Related link: link)

   (b) Calculate each student's grade. The multiple choice part of the quiz should be worth 60% of the grade and the essay should be worth 40%. Use the same letter-grade-to-score mapping as used in our CSCI1320 syllabus. (The percentage at the boundary of each grade is assigned a high grade. For example, if a student's grade percentage is 90%, the student receives 'A-', not 'B+')

   (c) Display each student's ID along with the grade they received on the quiz.

GRADES follow the standard percentage breakdown for the College of Engineering:

93%-100%	A
90%-93%	A-
87%-90%	B+
83%-87%	B
80%-83%	B-
77%-80%	C+
73%-77%	C
70%-73%	C-
67%-70%	D+
63%-67%	D
60%-63%	D-
0%-60%	F

Figure 2: The letter-grade-to-score mapping as used in our CSCI1320 syllabus

Task 2: Intro to C++

Earlier on in the semester we wrote simple calculator script in MATLAB. Now let's do the same in C++. Problem description: write a program that prompts the user to enter 2 numbers. The two numbers should be stored in variables of floating point type. The results of each of the following operations should be printed to the screen as a part of a meaningful message:

1. adding them together
2. subtracting the second from the first
3. multiplying them together
4. dividing the first by the second

(You are welcome to experiment with writing the result of each operation to a variable, or you can just print the results directly.)

NOTE: If your C++ file does not compile with standard g++ compiler you will get a zero on the C++ portion of the assignment (no partial credit). This will be the case with all C++ assignments going forward.

Instructions to submit your Assignment

Please zip all your script/function files, make the file name <first_name>_<last_name>.zip and upload it on Moodle. Remember that your file must be free of errors upon running it in MATLAB or compiling it with standard g++ compiler. Failing to do so will result in deduction of points from your total score for this assignment. Keep in mind the Honor code and ensure that you do not violate any of the rules it entails.