W1005 - Fall 2014
Homework 4

- Due by Friday 4pm (Nov. 7th).
- See submission instructions.
- Always include your name and UNI at the top of your submitted files.

1. Write a function '**add_grades2**' which takes as input a cell array (G), followed by three numbers (hw1,hw2, hw3).
   The function prompts the user to enter the name of one student (string), and a row number (integer).

   Your function should return a single variable which is the updated cell array with the name and grades entered in the appropriate row. The name should be stored in column 1, grades in columns 2-4.

2. Write function '**add_grades3**' which is similar to 'add_grades2' but takes as input and returns an *array of structs* rather than cell array. Each struct (in your array) records the name and grades for one student in the fields {name,hw1,hw2,hw3}.

3. Write a function '**stat_grades**' that takes as input a cell array (G), and returns two variables: a vector A which contains the averages over all students for each grade (hw1/hw2/hw3), and a sorted cell array G (see below).

   The variable 'G' is the cell array from problem 1. You need to sort in a consistent fashion (hint: sortrows) in descending order according to the hw1 (column 2) grade.

   Calling 'add_grades2' can help with generating a cell array to test your code.

   Note: to compute average over elements of a cell array it is OK here to use a for loop. There is a better way however (help cell2mat).

4. Write a function '**isitequal**' which takes as input two strings (S1, S2) which consist of letters mixed with digits and other punctuation marks. The function compares the two strings as follows:
    1. Extract only the digits from each string (in order)
    2. Extract only the letters from each string (in order)
    3. Compare the extracted digits and the extracted letters (case-insensitive) from the strings, return 1 if both are identical, 0 if not (in variable T).

    For example, given two strings:
    S1 = 'abc246T8!?'; S2 = '24ab68ct';
    Extract digits only:
    S1 --> 2468 S2 --> 2468
    Extract letters only:
    S1 --> abcT S2 --> abct

    Compare: digits are equal, letters are equal (ignore case).
    Return T = 1.

    Note: you can solve this problem without using regular expressions

5. Write a function named '**read_my_data'** which has four input parameters {file_dir, N, delim, subM}. Your function should do the following:

    a) Suppose you have a directory whose name is stored in 'file_dir' (string). The directory contains 100 files all named 'data_#.txt' (data_1.txt, … , data_100.txt).
    Each file has strictly numeric data separated by the delimiter 'delim'.
    b) Your function should open each file (from 1 to N), read the data into any variable, where the range of the data is specified in 'subM' and compute the total sum over all N files.
    For example, say N = 2, and you read data into D1 (for file 1), and D2 (for file 2). Then the total sum should be D = D1+D2;

c) Finally save the "total sum" variable to a new file named 'ndata_#.mat', (hint: notice file extension) in the same directory. You should substitute N for '#' e.g. 'ndata_2.mat'.

   You can assume that there are no errors in the data or the input parameters.

6. Write a function named '**class_plot**' with 2 input parameters (M, T). Your function should do the following:

   a) Suppose M is a matrix with 3 columns. Columns {1,2} of M are the {x,y} coordinates of points in the plane. Column 3 is the class to which the point belongs. The values of column 3 are some unknown integers. T is a struct with one field: 'c1'.

   b) Your function should check that T is indeed a struct and that the field c1 exists. Use the '*error*' function to indicate an error if your checks above fail.

   c) Your function should choose only the points (rows) of M whose class (column 3) matches the value of the field c1 (in struct T).

   d) Your function should plot the selected points (from part c) as circles in blue.

      Note: you can test your function by generating some random data (using 'rand' for example)

Your zip folder should include the following files:

add_grades2.m
add_grades3.m
stat_grades.m
isitequal.m
read_my_data.m
class_plot.m