420-LCU-05 Programming in Python - Assignment 3
Due May 03rd, 2018

1- **Identification section**: This section must be either in a comment, with a '#' preceding each line, or enclosed within triple quotes ('''). The grader and I need this section for the accurate processing of your assignment. Assignments missing this may lose up to 5% of the mark.

"""
Your Name and ID
420-LCU Computer Programming, Section #
Thursday, May 03rd
S. Hilal, instructor
Assignment 3
"""

2- **Submission**: Submit your assignment in 1 Python file, with the extension .py. No need to create a ZIP file. Be sure to respect other instructions specified in the assignment. An important part of each assignment is to correctly follow the instructions as closely as possible.

3- **Late assignments** are accepted up to 1 week from deadline. But late penalty will be applied.

For this assignment, you will develop an application that is very similar to Assignment 2 but will be using some different concepts and data structures that we have been learning.

<u>Learning Objectives:</u>

- Defining and using a simple class, class & instance attributes, and methods
- Understanding the concepts of class and instance(s) of a class.
- Using dictionaries as a temporary storage and for arbitrary access (similar to database)
- Using files as an alternative to user input and to store program output.
- Using structured formatting to print information

The program will enable a teacher to **<u>input</u>**, **<u>analyze</u>** and **<u>report</u>** on the grades of the students in one class based on their marks in different components of the course. It can calculate a student's **<u>total grade</u>**, **<u>letter grade</u>**, **<u>class average, grade distribution, and give a useful visualization of the stored data as requested</u>**. Your program will be using the **<u>Student</u>** class that we have seen in lecture notes. You will be modifying the basic class definition to add attributes and methods. You may also add other functions to your program.

It is strongly advisable to use the basic structure of your program in assignment 1 to build this program. You need to determine what to keep and what needs to be removed, slightly changed or replaced.

You need to start working on your program early and seek help as needed to be able to complete your program by the defined **deadline which will be strictly enforced**.

1- Program Input and Description of Data:

A student can be identified by a name (first name only) and ID (3-digits). For each student, there are 6 grades based on 2 tests (20% each) and 4 assignments (15% each). The grades can be integers or floats. Test marks given /20 and assignments are /15. The program should be able to process input for any number of students but you can limit your tests to 10 students. Input file provided **students.txt**

The input to your program (the students' data) will be read from a text file (studentInfo.txt) that will be provided to you. For each student record, your program (class) calculates and stores the total grade and corresponding letter grade.

Total Grade	Letter Grade
87 or above	A
From 75 to 86 inclusive	B
From 65 to 74 inclusive	C
Below 65	F

2- Storing students' data:

Each student record will be stored in an instance of the Student class. A **dictionary** will be used as temporary storage manager that will store and keep track of all the Student instances. The dictionary will be the entry point to each student record and will be used to access the data for all the required tasks. Each dictionary item will store the record of 1 student. The student's id will be used as the key and the reference to the instance will be stored as the value.

Sample code will be provided to you to setup the dictionary.

Running Your Program:

The program starts by giving the user a menu of options to select from. When the program completes the processing of a given option, the menu will be displayed again and user prompted to make another selection and continues until the user selects option 5, Exit.

Sample Run:

Welcome to the Teacher's Simple Class Calculator. Here's the list of options:

1- Read and process Students' records
2- Display All student records
3- Display the record of a particular student
4- Display the class average and grade distribution.
5- Exit

Select an option by entering its number or 4 to exit:

Description of the Different Options (above):

1- **Option 1**: Read the text file provided and store student data. Calculate and store total grade and letter grade for each student. You can assume that all records are valid and that there are no duplicates. Duplicate names may exist but no duplicate IDs.
2- **Option 2**: Display all the information stored for each student in a table like format.
3- **Option 3**: The program will ask the user to enter the name and ID of the student. The program will print the student's record. Same format as in option 2. If the name and ID do not match a stored record, an error message is displayed.
4- **Option 4**: The class average and grades distribution will be displayed.

The program will display the main menu following the completion of each option and until the user selects option 5.

Testing Your Program:

The data in the file will be of the same format as the input for assignment 2; Separated by commas and no spaces. Each record will be on a separate line.