# Homework 7
# IT-210

**INSTRUCTIONS:** You must work on this homework individually. Write all your solutions in a single .py file. Add a substantial amount of comments for each solution. All applications must be developed using Python 3.x. Submit your assignment in the Homework 7 Dropbox in D2L by 11.59 pm on the due date to receive credit. Late Homework will not be accepted. Email attachments will be ignored.

1. Provide a class for authoring a simple letter. In the constructor, supply the names of the sender and the recipient:

   ```
   def _ _init_ _(self, letterFrom, letterTo)
   ```

   Supply a method

   ```
   def addLine(self, line)
   ```

   to add a line of text to the body of the letter. Supply a method

   ```
   def getText(self)
   ```

   that returns the entire text of the letter. The text has the form:

   ```
   Dear recipient name:
   blank line
   first line of the body
   second line of the body
   . . .
   last line of the body
   blank line
   Sincerely,
   blank line
   sender name
   ```

Also supply a driver program that prints the following letter:

```
Dear John: \\
I am sorry we must part.
I wish you all the best.\\
Sincerely,
Mary \\
```

Construct an object of the Letter class and call addLine twice.

2. (a) Implement a class Student. For the purpose of this exercise, a student has a name and a total quiz score. Supply an appropriate constructor and methods getName(), addQuiz(score), getTotalScore(), and getAverageScore(). To compute the latter, you also need to store the number of quizzes that the student took.

   (b) Modify the Student class to compute grade point averages. Methods are needed to add a grade and get the current GPA. Specify grades as elements of a class Grade. Supply a constructor that constructs a grade from a string, such as "B+". You will also need a method that translates grades into their numeric values (for example, "B+" becomes 3.3). [You may obtain traditional grading scales from here `http://en.wikipedia.org/wiki/Academic_grading_in_the_United_States`]

3. Implement a class Car with the following properties. A car has a cer tain fuel efficiency (measured in miles/gallon) and a certain amount of fuel in the gas tank. The efficiency is specified in the constructor, and the initial fuel level is 0. Supply a method drive that simulates driving the car for a certain distance, reducing the fuel level in the gas tank, and methods getGasLevel, to return the current fuel level, and addGas, to tank up. Sample usage:

```
myHybrid = Car(50)              # 50 miles per gallon
myHybrid.addGas(20)             # Tank 20 gallons
myHybrid.drive(100)             # Drive 100 miles
print(myHybrid.getGasLevel()) # Print fuel remaining
```

4. Consider a hospital scenario. Design classes for:

   - Patients. The class might have fields like unique ID for the patient, name, male or female, age, address, phone number, date of birth, height, and weight.
   - Doctor. Fields might include the doctor's name, unique registration number, qualification (DO or MD), specialization (surgeon, pediatrician, etc.), phone number, office hours, and office location.
   - Patients' records in a hospital. The record might have fields like last date of checkup, doctor's unique ID, patient's unique ID, list of health problems in the patient, list of medicines prescribed, cost of the checkup, final report, and so on.

Methods of these classes would be mostly get and set methods for the corresponding fields.

5. Design a class called Color. The fields of the class are three decimals for Red, Green, and Blue components in the range 0 to 1, inclusive (0 indicates Black and 1 indi- cates White). Add checks to ensure that the values are always in the given range. Provide addition and subtraction operators for the color class. Include saturation in the addition and subtraction: if any component goes less than 0 or greater than 1, assign them 0 or 1, respectively.