CECS 220
Summer 2021
Assignment 4
Due Date: By 9:00 pm Wednesday, July 14, 2021

Instructions about assignments and reports:

Your presentation in your report reflects great deal about you, your understanding of the assignment and on how much this course means to you. I try very hard to look at the substance of the report but I will be lying if I said that presentation does not influence my judgment. It would be wise on your part to assume that this true in every course at school and in real life/work. I expect your reports to be well formed and conform to the following rules:

1. All reports have to be submitted as a PDF report that contains:
   1.1. Title page with your name, assignment number and the day you are actually submitting this report (Not the assignment due date)
   1.2. A brief description of the assignment or the statement of the problem which ever is appropriate.
   1.3. A brief description of the logic employed and the needed input and expected output.
   1.4. A comprehensive set of snapshots showing the inputs submitted, outputs obtained in the case of a successful output or a failure.
   1.5. Any conclusions, analysis, or answers to any questions I as you as part of the assignment.
2. All source code of all your classes interface and implementation and not the whole project (.java files only). You may combine these in one text file.
3. Please zip both the PDF document with the source code and submit one zip file.
4. Name the final zipped file using your initials and assignment number, e.g. my file should be named: ini_assignment_4.zip

CECS 220
Summer 2021
Assignment 4
Due Date: By 9:00 pm Wednesday, July 14, 2021

Please do the following:

1. Programming project 7.3 of Chapter 7:
   Write a class called Course that represents a course taken at a school. Represent each student using the modified Student class from the previous programming project. Use an ArrayList in the Course to store the students taking that course. The constructor of the Course class should accept only the name of the course. Provide a method called addStudent that accepts one Student parameter. Provide a method called average that computes and returns the average of all students' test score averages. Provide a method called roll that prints all students in the course. Create a driver class with a main method that creates a course, adds several students, prints a roll, and prints the overall course test average.
2. Programming project 7.5 of Chapter 7:
   Write a Java interface called Priority that includes two methods: setPriority and getPriority. The interface should define a way to establish numeric priority among a set of objects. Design and implement a class called Task that represents a task (such as on a to-do list) that implements the Priority interface. Create a driver class to exercise some Task objects.
3. Programming project 8.1 of Chapter 8:
   Write a program that reads an arbitrary number of integers that are in the range 0 to 50 inclusive and counts how many occurrences of each are entered. Indicate the end of the input by a value outside of the range. After all input has been processed, print all of the values (with the number of occurrences) that were entered one or more times.
4. Programming project 8.13 of Chapter 8:
   Define a class called Car that extends Group and presents the drawing of a car (side view). Use polygons, poly-lines, and other shapes to present the car. Write a JavaFX application that displays the car.

You may use any of the code in the source files posted on Blackboard.