CSCE 1040 Homework 4

For this assignment we are going to implement the Mean Green Airlines Application that we designed in Homework 3.

For the purposes of this assignment you will need to create a user interface menu that matches your design as well as implement each of the entities as classes in C++. Note we will be using C++ and you will need to use the g++ version of the compiler.

You **may use any** of the C++ STL at this time. You may NOT use Inheritance at this time. It is STRONGLY recommended, in fact it is required, that you use a pattern-based solution such as the transaction pattern we have discussed, just as you created in your design. Other methods or designs will significantly increase your development time, stress level, as well as the
degree of difficulty for this assignment. And you will lose a portion of the points since it is required. Be sure that your implementation reflects the information in your updated design; based on any feedback you received from the grader or instructor.

You may need to modify your design from Homework 3 based on grader comments and class discussion. You will need to turn this updated design in as well as use it for the basis of your program
Be sure to attend class lectures, as we will discuss many of the topics you will need to complete the assignment!

Program Requirements
Your program will be written in C++ not any other computer language. You will include the steps in your algorithm in your code. You may, of course, paraphrase them if you like.

Your program will be graded based largely upon whether it works correctly on a CSE Department Linux machine.

Your program will also be graded upon your program style. At the very least your program should include: A consistent indentation style as recommended in the textbook and in class. It should also use meaningful variable names. A block header comment section that

includes: Your Name and Email Address, and a brief description of the program.

Your program's output should initially display the department and course number, program number, your name, and your email address – as in Homework 1.

Be sure to create appropriate test data and execute tests for proper and improper data on all functions. You will submit your program source file to the Canvas website under the **"Homework 4"** drop box. Make sure you submit your program before the due date and time. You must also submit your updated Design file, and a short report about your efforts. Each class should have its own header (.h) and code (.cpp) file as well as a .cpp for main and, if needed, a .h file for main if any other functions of constants are needed. This means you will be turning in 2 PDF's (updated design document and report) , 7 cpp files (6 classes and 1 main) and at least 6 header files (1 for each class). If you create a .h for main then you will have 7 header files. You should also create a makefile to make it easier to compile your code for both you and the grader. Finally zip all of these files into a single zip file named hw4xxx.zip, where xxx represents your initials.

Please be sure and test your program to make sure it is working properly. You can either do this by hand (calculating some test values on paper to see if they match what your program says), or temporarily display various intermediate values you're calculating in the process and desk check the results to make sure they are correct. The more test cases you try and you get correct answers, the more certain you will be that when the grader uses his or her own test cases that your program will produce the correct result.

UPDATE REGARDING TIME

You may consider all times to be GMT (i.e. 0 offset for time zone) You should include this in your user prompts that all times should be entered as GMT. This will greatly simplify the time part of the problem.