CSCI 2300, Fall 2019

Team Project

Phase 3: Core Logic Implementation with Unit Tests [100 points]

Overview

In the previous phase of the project you designed your class interfaces, just enough to make them compile. In this part of the team project you need to implement the details of your **model** classes and test them.

Details

Create a test class for each class in the **model** directory of your project. If you class is in file `A.java`, the corresponding test should be in file `ATest.java`. Implement test cases for each method of each class. Keep the tests independent of each other. Each test case needs to focus on one method of the class you are testing. Initially, the tests will fail (if you have not yet implemented the details of your class methods).

Once your test cases are in place, implement the details of your model classes and rerun your tests. You should see some of your tests passing now. Continue working on implementing the details until all tests pass.

At the end of this phase you should have a complete implementation of all classes in your **model** directory and unit tests for them.

Submit

Submit your code (changes to your model classes and the unit tests you added) to your project team git repo. All the submitted code must compile.

Grading

Your grade will be based on the following criteria:

[50 points] Model implementation:

- Is all the necessary code present and working correctly (30 points)?
- Does your solution avoid code duplication (5 points)?
- Does your code follow Single Responsibility Principle, Open Closed Principle, and Liskov Substitution Principle (10 points)?
- Does the code follow CSCI 2300 coding standards (5 points)?

[30 points] Unit tests:

- Did you provide at least one unit test per method of each class in the model directory?
- Did you provide several unit tests for more involved methods?

[20 points] Peer evaluation results: how well did you work with your teammates and how much did you contribute to this phase of the project (as evaluated by your teammates).