CS 3331 - Assignment 3
UML Class Models
Due Wednesday, October 8th by end of day on BBlearn

In this assignment, you will practice developing UML Class diagrams to describe system elements and structure, and will practice writing code that conforms to the class diagram. You can draw UML Class Diagrams by hand, or by using any software tool of your choosing.

The Battle Ship game is played with two players. Each game has a number of properties, such as start time, end time, type of game (two human players, or one human and one computer player). A player, can be either a human being or a computer. Each game must have at least one human player. Every player has exactly 45 torpedoes to throw at enemy waters. Each torpedo has two coordinates, x and y.

Each player has 6 ships to position on the grid. Ships have different size and value. The game is composed of many rounds. At each round, a player throws one torpedo and can view the results. If the torpedo hits a ship, then the player can throw a second torpedo. This goes on until there are no torpedoes left, or when one player has lost all his ships. The game has one of two possible outcomes; one of the two players can win, or it can be a tie.

Players collects points. For every game won, the player collects two points, and for a tie the player collects only one point.

Part 1 (5 points):
Draw a UML class diagram that covers all the elements of the Battle Ship game as described above. A complete class diagram should depict the following:
   a) All classes, their attributes and methods, if any.
   b) All associations and their multiplicities
   c) Association names
   d) Role names

A good class diagram is also one that has a good layout and is easy to understand.

Part 2 (5 points):
Use your design in part one and implement it in Java. Your implementation should cover the following elements at minimum:
   a) Implementation of all classes and associations.
   b) Each class should include all required attributes.
   c) Each class should implement two methods; add and remove. Add method creates a new object and add it to the association, and remove method removes objects from the association. These two methods must not violate the association multiplicities. If a violation occurs, the method should report an error to the user.
   d) Each class should implement a method to list all objects participating in the association.

Evaluation Criteria

Your assignment will be evaluated using the following criteria.

- _Correct_: Proper use of modeling notation, correct identification of key system elements.
- _Consistent_: elements of the model and implementation are consistent.
- _Unambiguous_: Are there any ambiguities, or aspects that can have multiple interpretations?
- _Complete_: Does the model cover in sufficient depth the battle ship game?

Submission Guideline

Submit a single PDF document for the class diagram.
Submit the code and sample runs to demonstrate your code functions as expected.
Submit by the due date, no late submissions are accepted.