**CS231 Project# 2 - Ribopoly**

You may complete this project on your own, or in a team of two people.

You will implement a variant of the game of Monopoly. The rules will be similar to those stated in <http://en.wikibooks.org/wiki/Monopoly/Official_Rules> . Here are a few of the differences:

1. We will implement “Chance” and “Community Chest” differently. A player landing on either square will be charged/awarded a random value in the range [-$200 :+$200].
2. Rather than have human players make decisions, the game will be automated so that decisions are made by classes that implement the IPlayer interface.
3. The game will proceed from start to finish without human intervention.
4. A skeleton library is provided here: <http://ribler_r1.web.lynchburg.edu/cs231/Ribopoly.zip>

Here is a reminder about the cost of houses:

Costs of houses are:

* Browns/Purples and Light Blues - $50
* Pinks and Oranges - $100
* Reds and Yellow - $150
* Greens and Dark Blues - $200

You must use test-driven development to complete your project. Unlike the previous project, you will have to write the tests yourself. Please remember to keep the tests short.

Your development should proceed using the following set of goals in the order in which they are listed.

1. Test/Create a FairDie class that implements the IDie interface.
2. Test/Create a FairDice class that implements the IDice interface.
3. Test/Create a Player class
4. Test/Create a Game class. Have a single player move around on the board. Report the player’s position after each move.
5. Test/Create NothingSquare derived from IGameSquare
6. Test/Create PenaltySquare derived from IGameSquare that charges people for landing on a square.
7. Test/Create PlusOrMinusSquare classes derived from IGameSquare that implement Chance/Community Chest
8. Test/Create PropertySquare derived from IGameSquare that charging people rent and pays the rent to the owner.
9. Test/Create GoToJailSquare derived from IGameSquare that implement GoToJail
10. Test/Create RailRoadSquare derived from IGameSquare that implement Railroad squares.
11. Test/Create UtilitySquare derived from IGameSquare that implements Utility squares.
12. Test/Create a player that buys every available affordable property.
13. Test/Create players that build houses whenever they can afford to.
14. Continue on your own to complete the game.

We may not finish, but let’s see how far we can get.