**Object-Oriented Programming CSCE A222 Final Project: Monopoly**

**IMPORTANT**: This project is is prepared with GUI design on mind. Therefore if your code does not compile and no GUIs, you will lose 50% of the grade.

Implement the game of Monopoly using object-oriented programming techniques and Java. At a minimum, your solutions should have the following classes: Square, TaxSquare, BuyableSquare, PropertySquare, UtilitySquare, RailroadSquare, player, Monopoly, and a class with a main method. In addition to these classes, you must also create and use at least one interface or abstract class. It is up to you to figure out what should go in each of the classes and their relationship with one another. A GUI is required for the project with a minimum of a roll, buy property, and buy house buttons. Your GUI should display the game board, the players moving across the board, and any relevant information about what is happening during the game. **NOT ALL** monopoly rules will be implemented or are original. **DO NOT** upload all the classes in one java file. Avoid duplicate code as much as possible. You should follow good programming practice and comment/document your code.

Implement the game however you wish as long as you meet the requirements noted on this document.

A detailed UML diagram must be submitted and briefly explain why you chose to implement the game in the way that you did. A minimum of three paragraphs of text is required.

**Class Specific Notes:**

*Monopoly*

- Decide how to store your gameboard internally within this class (array, linked list, etc.)

*Square*

*RailroadSquare*

- Railroad rent is based off of the number of railroads owned ($25, $50, $100, $200).

*UtilitySquare*

- Rent should be determined by the formula: (dice roll \* multiplier).

**Text File Format & Cards:**

*squares.txt*

Values are separated by commas. First number identifies what class it corresponds to.

* -  Square (1): Square name
* -  PropertySquare (2): Square name, color, price, rent, price w/ 1 house, price w/ 2

houses, price w/3 houses, price w/ 4 houses.

* -  TaxSquare (3): Square name, tax price
* -  RailroadSquare (4): Square name, rent
* -  UtilitySquare (5): Square name, price, multiplier value

- You can use this class for the four corners of the board & the community/chance squares (You do not need to split up *go, jail, free parking, go to jail, chance,* or *communit*y chest squares into 6 different classes).

*Community Cards*

* -  Advance to Go: Move player to go square.
* -  Doctor Fees: Subtract $100 from player’s money.
* -  Go to Jail: Move player to jail square & skip next turn.
* -  School Fees: Subtract $200 from player’s money.
* -  PFD: Add $350 to player’s money.

*Chance Cards*

* -  Take A Walk On The Boardwalk: Move player to Boardwalk.
* -  Advance to Illinois Ave.: Move player to Illinois Ave.
* -  Advance to Nearest Utility: Move player to nearest utility.
* -  Advance to Nearest Railroad: Move player to nearest railroad.
* -  You Won The Lottery: Add $1000 to player’s money.

**Additional Requirements & Information:**

* Read in the provided txt file to set up your gameboard.
* 4 players.
* Original win conditions do not need to be implemented. Allow the game to run for 10

turns and then display the winner (most money) instead.

* Community and Chance cards can be implemented however you want. Implement all

cards listed above.

* Trading/Selling property does not need to be implemented.
* Bonuses for owning all properties of the same color do not need to be implemented.
* Gain $200 if player lands on or passes go.
* Players start with $1500.
* Properties can have up to 4 houses (hotels not required).
* Houses cost $50.
* Jail time is 1 turn.
* Free parking gives player $20.
* All squares should be implemented with their appropriate action.
* Players traverse the game board using 2, 6-sided dice.
* Players can buy squares if they are not owned and buyable.
* Landing on a square owned by another player should deduct the correct value from the

current players money and add it to the owner of the property’s money.

**Submission Instructions:**

Zip ALL your .java files and .txt files and upload them to blackboard using the naming convention: ‘lastname\_monopoly.zip’.

**Your program should be set up in a way where the grader only has to compile and run your code. We should NOT have to go into your source code and edit anything.**

***Example***

