CS611 A1 Object-oriented Software Principles and Design (Fall 2020) Assignments

Upload Assignment: Tic Tac Toe - Take II

Upload Assignment: Tic Tac Toe - Take II

ASSIGNMENT INFORMATION

Due Date	Points Possible
Monday, September 28, 2020 11:59 PM	**100**

Design and Develop an **Object Oriented** class structure that allows you to play *continuing* rounds of two different types of turn based board games, tic-tac-toe (see attached program specification), and a close variant order and chaos.

The specification for the tic-tac-toe game has not changed. However, your tic-tac-toe game should be able to ***seamlessly*** scale from a 3x3 board to an nxn board as specified.

Care should be given to your object design. Your class design should center on the logical objects that are needed to play these games and variants of. Your design will be evaluated not simply on correctness of play, but on ***scalability*** and ***extendability***. Concepts we will more specifically discuss Tuesday.

To test for correctness. Consider the following scenarios:

1. Is it possible to create a board with a different number of rows and columns?

2. Is it possible to select a position on the board that has previously been selected or not even on the board?

3. Is it possible for the game to never end?

To test for scalability, consider the following scenarios:

1. **Board scalability:** Should your game be played with different size boards and run seamlessly? **This is a reasonable expectation that we will test for in the game of tic tac toe.** Keep in mind though that board *scalability is not infinite* and ceilings should be placed on the size of the tic-tac-toe board. Additionally not all games should allow for board scalability. The implementation of the game of *order and chaos* should only allow for play on a 6x6 board, following the original specifed winning rules. Consider carefully though, ***how*** you place and enforce these ceilings.

2. **Player Scalability:** It is not unreasonable to consider playing with teams. Playing turn based games with teams can be accomplished multiple ways, here are a couple:

   - Each team can alternate players during each round.

   - Each round of the game can be played by two specific team members, and new team members can be randomly selected for each subsequent round.

This is not an expectation of this submission, however, the ability to do so will earn bonus points.

Scalability is not limited to the state of play. Scalability can also imply learning and

intelligence. What if (at some point in the future) you wanted to make a smart game that learned from each round. Specifically, given the size of the board, which is the best starting point, or which is the most popular starting point, etc. This would require information to be maintained by each round, the board, and even each cell of the board. This is not an expectation of the current submission, but the ability to allow for this type of scalability is not unreasonable and your design will be evaluated accordingly.

To check for extendability, consider if your class design would be easily extendible to other **turn based games** like *sorry*, *monopoly or card games*, or other *cell based* **board games** like *connect four*, or *chess*, etc.

Due Date: Monday, September 29th @ 11:59 p.m.

Please note that this is not a group assignment. Your design and your implementation should be your own.

This submission will be scored in two parts: part I, a re-evaluation of your object structure for TTT and part II, how it was extended to allow for variations to be played. Your score on this assignment will be combination of the two, using a 30% (part I) 70% (part II) split.

Your program will be evaluated as follows:

- **Object Design, with an emphasis on:**
  - **scalability**
  - **extendibility**
- Implementation, with a specific emphasis on:
  - readability
  - best practices
- Usability
- Program (and game) Correctness

Deliverables:

- Read me text file
- Java source code

A word advice. The most successful submissions will be from those of you who ***limit*** coding this weekend to simple exercises ensuring your understanding of the principles of Encapsulation and Inheritance (as needed). Your focus as related to the specifics of this assignment this weekend should be to think through the class and object design. ***What is*** and ***What has*** should be the questions you should be asking yourselves. We will discuss more during Tuesday's lecture.

Have fun!

TicTacToe.docx.pdf

ASSIGNMENT SUBMISSION

Text
Submission

Write Submission

Attach Files

Browse My Computer

Browse Content Collection

Browse Cloud Storage

ADD COMMENTS

Comments