

CSC 520, Advanced Object Oriented Programming, Fall 2010, Dr. Dale E. Parson

Assignment 3, implementing the basic logic of Class ScrabbleAI, due October 22, 2010

Per my email to the class of October 2, by October 22 you will need to have a working AI that makes a legal move within one minute integrated into the ScrabbleAI method makeMove().

You cannot change the public method signatures of my ScrabbleAI class in any way that affects its relationships to the class that Associates to it (ScrabbleGame) or the classes to which it associates (ScrabblePlayer, ScrabbleBoard, ScrabbleScoreSheet).

You are free to add any fields that are necessary to implement method makeMove(). We will discuss the likely need to make these fields transient in class. In fact, we will discuss all of my design decisions on this project in class Thursday October 7. I do not expect your solutions to match mine -- as usual, I made discoveries as part of implementing this.

You must use my class with your enhancements to makeMove() because the associations with other classes have some dimensions that we need to discuss. I ***STRONGLY*** advise you not to try to use multiple threads at this time. We will discuss that issue. I have generated my Javadoc on bill.

It turns out that having more than one way to inject moves into a game (from a human/UI or from an automatic agent) triggers some interruption complexities that I have seen in industrial projects, but that we have not gone over in class. This is good -- it's a solid example. If you have to miss class on October 7, please try to get notes from someone.

You can now run a MIDI-less game by clicking the JAR file in games2010rev5, or a MIDI game via the JAR file in gamestomidi2010rev5. You enable or disable an AI for a given player with the AI On|Off button for that player, and you ask the AI to make a move via the AI RUN button. My AI swaps all 7 tiles for the net gain of 0 points. I plan to do better.

If you check the AI Auto checkbox for a player, its AI runs automatically on its turn without requiring the AI Run button to be pressed. If you check AI Auto boxes for all players, the GUI could become starved with no way to interrupt, so the GUI automatically unchecks one after each round of completely automatic play. This is related to threading and GUI interruption issues that we will discuss in class on Thursday.

Have fun. If you encounter bugs, please write them down.

Instead of a README file, I have put fairly copious notes into ScrabbleAI.java's comments. Don't worry if your design does not look close to my ScrabbleAI -- mine has a few serious differences from what I anticipated.

Please use my gamestomidi2010rev5.oct02mac.zip and models2010rev5.oct03.zip distributions as a starting point. You are not working on UML models in this

assignment. When your solution is ready, please run **gmake turnitin** from directory **games2010rev5**. You can use **gmake clean build** to build your JAR files for testing, or you can build under the Eclipse Project menu selection. I will show you how to run the executable from Eclipse in class.