CS380 HW Assignment 4
Due: May 20th 11:59pm

Part 1: Programming Assignment (100 points)

In this assignment you will program an agent that uses Minimax to play the game of Othello. You do not need to implement the game of Othello this time if you do not want. You can use the provided Java implementation (java-othello-src.zip), which is a very simple implementation of the game of Othello. You can test it by running the file "Test.java", which should run a game where both players generate moves at random.

Note:

- The zip file is expanded in a folder “java-othello-src”, of which subdirectory “cs380” is a child.
- To compile, “cd” to directory “java-othello-src” and run

  javac cs380/othello/Test.java
- To run, “cd” to the same directory and run

  java cs380.othello.Test

The Othello implementation is structured as follows:

- OthelloMove.java: this class contains stores a "move" (which player made the move and the coordinates of the move)
- OthelloState.java: this is the core class, which implements most of the functionality of the game. The functions you should be aware of, for implementing minimax are:
  1. public OthelloState(int a_boardSize): the constructor, you can create boards of whatever size you want, as long as it's 2 or bigger (Othello is typically played in a 8x8 board though).
  2. public boolean gameOver(): determines whether the game has finished or not.
  3. public int score(): returns the score (positive means player O is winning, negative means player X is winning).
  4. public List generateMoves(): returns the list of moves for the next player to move.
  5. public List generateMoves(int player): same as before, but you can specify which player you want to generate moves for.
  6. public OthelloState applyMoveCloning(OthelloMove move): creates a new game state that has the result of applying move ‘move'
  7. OthelloPlayer.java: this is an abstract class defining an agent that player Othello. Your agent should be implemented as a class that extends this one.
- OthelloRandomPlayer.java: an example agent that player Othello, but just by choosing moves at random.
- Test.java: an example of how to use all the above classes to play a game of Othello.

Specifically, what we are asking you to do is the following:

- Create a new class that extends OthelloPlayer (call it something original, to prevent any name clashes with the classes that your classmates will create, since we MAY put all the classes together at some point, to create a tournament!).
- Within this new class, implement an agent that player Othello using the standard minimax algorithm, as we studied it in class.
- As the evaluation function, just use the "score" function that is provided to you in the OthelloState class (make sure that your bot can play both as the first or second player).

- Your agent's constructor should accept the depth up to which we want to search.
- To make sure your agent works, make it play against the OthelloRandomPlayer we provide. Your agent should defeat it easily!

What to Submit

All homework for this course must be submitted electronically using Bb Learn. Do not e-mail your assignment to a TA or Instructor! If you are having difficulty with your Bb Learn account, you are responsible for resolving these problems with a TA, an Instructor, or someone from IRT, before the assignment it due. It is suggested you complete your work early so that a TA can help you if you have difficulty with this process.

For this assignment, you must submit:

- Your source code, written documentation for your program, and results of your testing.
- Use a compression utility to compress your files into a single file (with a .zip extension) and upload it to the assignment page.

Academic Honesty

You must compose all program and written material yourself, including answers to book questions. All material taken from outside sources must be appropriately cited. If you need assistance with this aspect of the assignment, see a consultant during consulting hours.