CS 480 Programming Assignment

Connect Five Game Playing

Connect Five is a harder version of the well-known game [Connect Four](https://en.wikipedia.org/wiki/Connect_Four). It is played on an 8x9 grid, and the objective is to get five pieces in a row. In the attached zip file you will find code implementing the game including a rudimentary AI player that uses the Minimax algorithm with a simple evaluation function. Your job will be to create a subclass of the AI player and improve its play. You can test your program by playing against it interactively, and by having it play against the simple AI player provided. (See the README file for more information about the program.)

1. Create a subclass of AIPlayer whose name is your CWID (see the example in A123456.py) – the file should have the same name as the object.
2. Implement Alpha Beta pruning by modifying the Minimax object into an AlphaBeta object, which will be used by your new AIPlayer subclass. Have it print the number of evaluated and the number of pruned nodes into a log file; evaluate the average percentage savings for searching to a depth of 4 (one game’s worth is sufficient).
3. Improve the evaluation function. This is totally open ended, but your new evaluation function cannot do any search and should compute a fixed computation time function of the board position. Document what the function computes and why.
4. Play several games against your player, at different search depths, and alternately as the first or second player. What can you say about your player’s strengths and weaknesses? How much does an extra ply of game-tree search improve its play? How might you think about improving the player further?
5. EXTRA CREDIT: Implement a form of dynamic search depth, searching deeper under some positions than others, if warranted, so that search time is limited but results are improved. Fully document your method and your code.

There will be an in-class tournament of your players against each other. The exact tournament protocol is to be determined. Note that if you do not implement your code entirely in the file named <yourCWID>.py and your player in an object of that name, your code might not run in the tournament in which case it will not be able to participate.

The tournament winner will receive a certificate and bragging rights, but the tournament will affect no one’s grades.