Lab Assignment 5
If Schrödinger's Cat Played Tic-Tac-Toe

Lab Objectives

- Understand and apply collections appropriate to solving a problem.
- Creating effective abstractions to model a concrete problem.
- See that a solution to a larger problem can be developed in terms of smaller solutions to smaller problems.
- Continue to apply your personal software development strategy.
- Challenge you to think and apply yourself more than you have so far in this course.

Problem Description

I've tried to make every assignment challenging in some respect because that is so vitally important to your development as a (future) computing professional. When you're challenged and stretched and forced to really apply yourself, that's when fundamental, long-lasting learning takes place. This is the most difficult assignment so far and it will certainly challenge you to think and to apply yourself. This assignment ties together most of the course topics so far and it introduces you to the (possibly) Next Big Thing in computing. It brings together algorithmic complexity and efficiency, various data structures, recursion, game playing strategies, rudimentary artificial intelligence heuristics, a child's game, advanced physics, quantum computing, about half the episodes of Star Trek, the entire Lost series, and – of course – Schrödinger's cat.

Your assignment is to write a program in Java that allows a human to play against the machine in a game of *Quantum Tic-Tac-Toe*. Your program must always try to never lose against its opponent, and it must assume that it's opponent will try to play a perfect game.

If you can't remember how to play classic tic-tac-toe, do the following.

1. Read: http://en.wikipedia.org/wiki/Tic-tac-toe
2. Play: http://www.atksolutions.com/games/tictactoedeluxe.html
3. Improve: http://ostermiller.org/tictactoeexpert.html

Quantum Tic-Tac-Toe is a combination of classic tic-tac-toe and quantum physics. That might sound bizarre, but it makes for a very interesting game of strategy. Your first step in completing this assignment will be to understand and be able to play a game of Quantum Tic-Tac-Toe. To that end, you need to do the following.

1. Read: http://chaos.swarthmore.edu/courses/pdg07/ajp/ajp000962.pdf
2. Play: http://www.paradigmpuzzles.com/QT3Play.htm
3. Improve: http://www.ParadigmPuzzles.com/

Once you understand the game, you can then start thinking about writing the software to play it. Here are some useful resources to that end.

1. Basic information on game trees: http://en.wikipedia.org/wiki/Game_tree
2. A lecture from Berkeley's version of COMP 2210: http://www.youtube.com/watch?v=Unh51VnD-hA
3. Basic information on minimax: http://en.wikipedia.org/wiki/Minimax
4. Basic information on alpha-beta pruning: http://www.ocf.berkeley.edu/~yosenl/extras/alphabeta/alphabeta.html

5. Basic information on transposition tables: http://en.wikipedia.org/wiki/Transposition_table

There are many decisions to be made: How will you represent the board and its positions? How will you represent and record moves? What strategy will you use to select the next move for the machine? What data structures and collections will you need to support the game play and all associated strategies? How will you design the overall architecture of your software (classes/methods/interfaces, etc.)? What type of user interface will your program have?

The requirements for the problem are intentionally broad. Your solution must, however, satisfy the following.

1. Your program must play against a human user.
2. Your program must be able to play in real time. Computer moves can't take too long to choose and execute.
3. Your program must play to win.
4. Your program must provide enough information at runtime so that the state of the game is clear to the human opponent.

Make sure that the work you present and the work that you turn in is your work and yours alone. You can use ideas from other people and other resources, but the solution itself – and especially the source code – must be yours alone.

Substantial partial credit will be given for submissions that follow criteria 1-4 (above) for *classic* tic-tac-toe.

Lab Turn-In

You must turn in your submission(s) as specified in Canvas.