**Othello Report**

Eckel AI Period 2

**Strategies Implemented:**

* Piece stability: detects if a piece can be changed; if it cannot be changed, it is called stable and therefore weighted more in the score function
* Opening book: a hard-coded set of moves that either black or white should make given an opening position

**Original Strategy:**

* corners are weighted the highest
* squares adjacent to corners have negative weight if there is no piece in the corner, otherwise it has positive weight if the piece in that square is the same as the piece in the corner
* mobility (number of o-moves subtracted from number of x-moves)
* weight from highest to lowest: corners, squares, mobility

**Implementation of New Strategies**:

Piece stability:

* for every piece, to calculate if it is stable, at least one of the following must be true
  + all the squares to the top and left each have the same piece as the current square
  + all the squares to the bottom and left each have the same piece as the current square
  + all the squares to the top and right each have the same piece as the current square
  + all the squares to the bottom and right each have the same piece as the current square
* this heuristic isn’t perfect, since the piece circled red could still be sandwiched between two white pieces in a diagonal, but it gives a pretty good idea of which pieces are more valuable

Opening Book:

* two dictionaries called x\_opening and o\_opening, keys are boards and values are integers that dictate which space x or o should play on
* in the find\_best\_move method, check if the board states exists in the opening book first, if it does, then just return that move

**Improvements:**

The main improvement came from the piece stability strategy. On my computer, at least, my AI could take 100% of the pieces most of the time, and other times rarely dipping below 90%. Strangely enough, my AI performed significantly better when it played as x compared to when it played as o. In terms of percentages, the opening book strategy didn’t improve much, but it did result in much faster and more optimal decisions made in the beginning of the game.