ECS170 Programming Assignment 2 Written

Russell Chien

February 2024

Evaluation Function

$Score = (W_1)(X_1 + 4X_2 + 10X_3) - (W_2)(Y_1 - 4Y_2 - 8Y_3)$

$$\text{Positional Weight Matrix} = \begin{matrix} 1 & 2 & 3 & 4 & 3 & 2 & 1 \\ 1 & 2 & 3 & 4 & 3 & 2 & 1 \\ 1 & 2 & 3 & 4 & 3 & 2 & 1 \\ 1 & 2 & 3 & 4 & 3 & 2 & 1 \\ 1 & 2 & 3 & 4 & 3 & 2 & 1 \\ 1 & 2 & 3 & 4 & 3 & 2 & 1 \end{matrix} \tag{1}$$

Motivation

For the evaluation function, we are considering the number of 3's in a row very heavily, since multiple 3's in a row almost always is a win. 2's and 1's in a row are weighted less as they can potentially have no way to improve, i.e. no more possible moves from that sequence. An opponent 3's in a row is weighted slightly less to incentive winning moves. Keep in mind that we only consider pieces in a row that have an empty spot on either side that can form a winning move. Finally, we consider the position of the board, weighting the middle column more heavily since it has the most possible sequences, and keeping the rows of the board uniform in weight.

Worked Example

$$\text{Midgame Board} = \begin{matrix} 1 & 2 & 3 & 4 & 3 & 2 & 1 \\ 1 & 2 & 3 & 4 & 3 & 2 & 1 \\ 1 & 2 & 3 & 4 & 3 & 2 & 1 \\ 1 & Y & X & Y & 3 & 2 & 1 \\ 1 & X & X & X & Y & X & 1 \\ X & Y & Y & X & Y & Y & 1 \end{matrix} \tag{2}$$

Assume player 1 is X and player 2 is Y.
Player 1 has two 3's in a row, one 2's in a row, and one 1's in a row.

Player 2 has one 3's in a row, two 2's in a row, and one 1's in a row.
Then $score = 1(2) + 4(1)(3+3) + 10(2)(1+2+3+2+3+4) - 1(4) - 4(2)(4+3+3+2) - 8(1)(4+3+2) = 154$

Alpha Beta Pruning Successor Function

The successor function takes in the current game state and returns all possible legal moves, recursively searches the game tree and uses the alpha and beta values to prune branches of the tree that do not need to be explored.

Testing

	Wins	Ties	Losses
Vs stupidAI	10	0	0
Vs randomAI	10	0	0
Vs monteCarloAI	20	0	0

alphaBetaAI vs stupidAI Tests

```
alphaBetaAI vs stupidAI Seed 1: W
        [0 0 0 2 0 0 0]
        [0 0 0 1 0 0 0]
        [0 0 0 2 0 0 0]
        [0 0 0 2 0 0 0]
        [0 0 0 2 0 0 0]
        [0 1 1 1 1 0 0]

alphaBetaAI vs stupidAI Seed 2: W
        [0 0 0 2 0 0 0]
        [0 0 0 1 0 0 0]
        [0 0 0 2 0 0 0]
        [0 0 0 2 0 0 0]
        [0 0 0 2 0 0 0]
        [0 1 1 1 1 0 0]

alphaBetaAI vs stupidAI Seed 3: W
        [0 0 0 2 0 0 0]
        [0 0 0 1 0 0 0]
        [0 0 0 2 0 0 0]
        [0 0 0 2 0 0 0]
        [0 0 0 2 0 0 0]
        [0 1 1 1 1 0 0]
```

alphaBetaAI vs stupidAI Seed 4: W

```
[0 0 0 2 0 0 0]
[0 0 0 1 0 0 0]
[0 0 0 2 0 0 0]
[0 0 0 2 0 0 0]
[0 0 0 2 0 0 0]
[0 1 1 1 1 0 0]
```

alphaBetaAI vs stupidAI Seed 5: W

```
[0 0 0 2 0 0 0]
[0 0 0 1 0 0 0]
[0 0 0 2 0 0 0]
[0 0 0 2 0 0 0]
[0 0 0 2 0 0 0]
[0 1 1 1 1 0 0]
```

stupidAI vs alphaBetaAI Tests

stupidAI vs alphaBetaAI Seed 1: W

```
[0 0 0 1 0 0 0]
[0 0 0 1 0 0 0]
[0 0 2 2 0 0 0]
[0 0 2 1 0 0 0]
[0 0 2 1 0 0 0]
[0 0 2 1 0 0 0]
```

stupidAI vs alphaBetaAI Seed 2: W

```
[0 0 0 1 0 0 0]
[0 0 0 1 0 0 0]
[0 0 2 2 0 0 0]
[0 0 2 1 0 0 0]
[0 0 2 1 0 0 0]
[0 0 2 1 0 0 0]
```

stupidAI vs alphaBetaAI Seed 3: W

```
[0 0 0 1 0 0 0]
[0 0 0 1 0 0 0]
[0 0 2 2 0 0 0]
[0 0 2 1 0 0 0]
[0 0 2 1 0 0 0]
[0 0 2 1 0 0 0]
```

stupidAI vs alphaBetaAI Seed 4: W

```
[0 0 0 1 0 0 0]
[0 0 0 1 0 0 0]
[0 0 2 2 0 0 0]
```

```
[0 0 2 1 0 0 0]
[0 0 2 1 0 0 0]
[0 0 2 1 0 0 0]
```

stupidAI vs alphaBetaAI Seed 5: W

```
[0 0 0 1 0 0 0]
[0 0 0 1 0 0 0]
[0 0 2 2 0 0 0]
[0 0 2 1 0 0 0]
[0 0 2 1 0 0 0]
[0 0 2 1 0 0 0]
```

alphaBetaAI vs randomAI Tests

alphaBetaAI vs randomAI Seed 1: W

```
[0 0 0 0 0 0 0]
[0 0 0 0 0 0 0]
[0 0 0 0 0 0 0]
[0 0 0 0 0 0 0]
[0 0 0 2 0 0 0]
[0 2 1 1 1 1 2]
```

alphaBetaAI vs randomAI Seed 2: W

```
[0 0 0 0 0 0 0]
[0 0 0 0 0 0 0]
[0 0 0 0 0 0 0]
[0 0 0 0 2 0 0]
[0 0 0 0 2 0 0]
[2 0 1 1 1 1 0]
```

alphaBetaAI vs randomAI Seed 3: W

```
[0 0 0 0 0 0 0]
[0 0 0 0 0 0 0]
[0 0 0 0 0 0 0]
[0 0 0 0 0 0 0]
[0 2 0 2 0 0 0]
[1 1 1 1 0 2 0]
```

alphaBetaAI vs randomAI Seed 4: W

```
[0 0 0 0 0 0 0]
[0 0 0 0 1 0 0]
[0 2 0 0 1 0 0]
[0 1 0 1 1 0 0]
[0 2 0 2 1 2 0]
[0 2 0 1 2 1 2]
```

alphaBetaAI vs randomAI Seed 5: W

```
[0 0 0 0 0 0 0]
[0 0 0 0 0 0 0]
[0 0 0 0 0 0 0]
[0 0 0 2 0 0 0]
[0 0 0 2 1 0 0]
[2 1 1 1 1 2 0]
```

randomAI vs alphaBetaAI Tests

randomAI vs alphaBetaAI Seed 1: W

```
[0 0 0 0 0 0 0]
[0 0 0 1 2 0 0]
[0 0 0 2 2 0 0]
[0 0 2 2 1 0 0]
[0 2 1 1 2 0 0]
[0 1 1 2 1 1 0]
```

randomAI vs alphaBetaAI Seed 2: W

```
[0 0 0 0 0 0 0]
[0 0 0 0 0 0 0]
[0 0 0 1 0 0 0]
[0 0 0 2 0 0 0]
[0 0 0 1 0 0 0]
[1 2 2 2 2 1 1]
```

randomAI vs alphaBetaAI Seed 3: W

```
[0 0 0 0 1 0 0]
[0 0 0 0 2 0 0]
[2 0 0 2 2 0 0]
[1 0 0 2 1 0 0]
[2 0 0 2 1 0 1]
[1 0 0 2 1 0 1]
```

randomAI vs alphaBetaAI Seed 4: W

```
[0 0 1 0 0 0 0]
[0 0 2 0 0 2 0]
[0 0 1 0 0 2 0]
[0 2 2 0 0 2 0]
[0 1 2 0 0 2 0]
[1 1 1 2 1 1 1]
```

randomAI vs alphaBetaAI Seed 5: W

```
[0 0 0 0 0 0 0]
[0 0 2 2 2 0 0]
[0 0 1 2 2 0 0]
```

```
          [0 0 2 1 1 0 0]
          [0 2 1 2 2 0 1]
          [0 1 1 2 1 1 1]
```

alphaBetaAI vs monteCarloAI Tests

```
alphaBetaAI vs monteCarloAI Seed 1: W
          [0 0 0 1 0 0 0]
          [0 0 0 2 1 0 0]
          [0 0 0 1 1 0 0]
          [0 0 1 1 2 0 0]
          [0 1 2 1 1 2 2]
          [2 2 1 2 2 1 2]

alphaBetaAI vs monteCarloAI Seed 2: W
          [0 0 1 0 0 0 0]
          [0 0 1 2 0 0 0]
          [0 0 1 1 0 0 0]
          [0 0 1 1 0 0 0]
          [0 0 2 2 0 0 0]
          [0 0 1 2 2 2 0]

alphaBetaAI vs monteCarloAI Seed 3: W
          [0 0 0 2 0 0 0]
          [0 1 0 1 1 0 0]
          [0 1 1 1 1 0 0]
          [0 2 1 1 2 0 0]
          [1 2 2 2 1 0 2]
          [2 2 1 1 2 2 2]

alphaBetaAI vs monteCarloAI Seed 4: W
          [0 0 2 0 0 0 0]
          [0 0 2 1 2 0 2]
          [0 0 1 1 1 0 1]
          [0 1 2 2 2 0 2]
          [1 2 2 1 1 0 2]
          [1 1 2 1 1 1 2]

alphaBetaAI vs monteCarloAI Seed 5: W
          [0 0 2 1 0 0 0]
          [0 0 2 1 0 0 0]
          [0 0 1 1 1 0 0]
          [0 1 2 2 2 1 0]
          [1 2 2 1 1 2 0]
          [1 1 2 1 2 2 2]
```

```
alphaBetaAI vs monteCarloAI Seed 6: W
           [0 0 0 0 0 0 0]
           [0 0 0 0 0 0 0]
           [0 0 0 0 0 0 0]
           [0 0 0 0 0 0 0]
           [0 0 0 2 0 0 0]
           [2 1 1 1 1 2 0]

alphaBetaAI vs monteCarloAI Seed 7: W
           [0 0 2 0 2 0 0]
           [0 0 1 1 1 0 0]
           [1 0 1 1 1 0 0]
           [2 0 1 2 2 0 0]
           [2 1 2 1 2 2 0]
           [2 1 2 1 2 1 0]

alphaBetaAI vs monteCarloAI Seed 8: W
           [0 0 0 0 0 0 0]
           [0 0 0 0 0 0 0]
           [0 0 0 0 0 0 1]
           [0 0 0 0 0 0 2]
           [0 0 2 0 0 0 2]
           [1 1 1 1 0 0 2]

alphaBetaAI vs monteCarloAI Seed 9: W
           [0 0 0 0 0 0 0]
           [0 0 0 0 0 0 0]
           [0 0 1 0 0 0 0]
           [0 0 1 2 0 0 0]
           [0 0 1 2 0 0 0]
           [0 2 1 1 2 1 2]

alphaBetaAI vs monteCarloAI Seed 10: W
           [0 0 0 0 0 0 0]
           [0 0 0 0 0 0 0]
           [0 0 0 0 0 0 0]
           [0 0 0 0 0 0 0]
           [0 0 0 2 0 0 0]
           [0 2 1 1 1 1 2]
```

monteCarloAI vs alphaBetaAI Tests

```
monteCarloAI vs alphaBetaAI Seed 1: W
           [2 0 2 1 0 1 2]
           [1 0 2 1 0 2 1]
           [2 2 2 2 0 2 2]
```

```
          [1 1 1 2 0 1 1]
          [1 2 2 1 2 1 1]
          [1 2 2 1 2 1 1]

monteCarloAI vs alphaBetaAI Seed 2: W
          [0 0 1 0 2 0 0]
          [0 0 1 2 2 0 0]
          [0 0 2 2 2 0 0]
          [0 2 2 1 1 0 0]
          [0 1 1 2 1 0 0]
          [0 1 1 2 1 0 0]

monteCarloAI vs alphaBetaAI Seed 3: W
          [0 0 0 2 1 0 0]
          [0 0 2 1 2 0 0]
          [0 0 1 2 1 0 2]
          [0 0 2 1 2 2 1]
          [0 2 2 1 2 1 1]
          [0 1 2 1 1 2 1]

monteCarloAI vs alphaBetaAI Seed 4: W
          [0 0 0 2 0 0 0]
          [0 0 2 2 0 0 0]
          [0 0 1 2 2 0 0]
          [0 2 1 1 1 0 0]
          [0 1 2 2 2 2 1]
          [0 1 1 2 1 1 1]

monteCarloAI vs alphaBetaAI Seed 5: W
          [0 0 0 1 0 0 0]
          [0 0 2 2 2 0 0]
          [0 0 1 2 1 0 0]
          [0 0 1 2 2 0 0]
          [0 0 2 1 1 2 0]
          [0 0 1 2 1 1 0]

monteCarloAI vs alphaBetaAI Seed 6: W
          [0 0 0 2 2 0 0]
          [0 0 0 2 1 0 0]
          [0 0 2 2 2 0 0]
          [0 0 1 1 1 2 0]
          [0 0 1 2 2 1 2]
          [0 1 1 2 1 1 1]

monteCarloAI vs alphaBetaAI Seed 7: W
          [0 0 0 0 2 0 0]
```

```
[0 0 0 2 2 0 1]
[0 0 0 2 2 2 2]
[0 2 0 1 1 2 1]
[0 1 0 2 1 1 1]
[0 1 2 1 1 2 1]
```

monteCarloAI vs alphaBetaAI Seed 8: W

```
[0 0 0 0 0 0 0]
[0 0 0 0 0 0 0]
[0 0 2 1 0 0 0]
[1 0 2 2 0 0 0]
[1 0 2 2 0 0 0]
[1 1 2 1 0 0 0]
```

monteCarloAI vs alphaBetaAI Seed 9: W

```
[0 0 0 0 0 0 0]
[1 0 0 1 0 0 0]
[2 0 0 2 2 0 0]
[1 2 1 2 2 0 0]
[1 1 1 2 2 0 0]
[1 2 1 1 2 0 0]
```

monteCarloAI vs alphaBetaAI Seed 10: W

```
[0 0 0 0 0 0 0]
[0 0 0 2 0 0 0]
[0 0 0 2 0 0 0]
[0 0 0 2 0 0 0]
[1 0 0 2 0 0 0]
[1 1 2 1 0 0 1]
```