## Three custom evaluation heuristics were implemented:

**AB_Custom:**
This evaluation heuristic uses the number of my moves minus two times the number of the opponent's moves (#my_moves - 2 * #opponents_moves) to evaluate the board score.
By using the number of my moves in the evaluation heuristic, it has the effect of maximizing my move options during game play. By using minus two times the number of the opponent's moves in the evaluation heuristic, it has the effect of minimizing the opponent's move options during game play and it does so more aggressively by using the weight of 2. I'm more likely to win if I have a lot of move options while the opponent has few move options left and the last player who has legal moves left wins.

**AB_Custom_2:**
This evaluation heuristic uses the minimum distance of my position to any of the 4 board edges minus two times the minimum distance of the opponent's position to any of the 4 board edges (my_min_edge_distance - 2 * opponents_min_edge_distance) to evaluate the board score.
Using the minimum distance of my position to any of the 4 board edges in the evaluation heuristic has the effect of trying to choose a move far away from all of the 4 board edges. By using minus two times the minimum distance of the opponent's position to any of the 4 board edges, it has the effect of trying to push the opponent close to the edge of board and it does so more aggressively by using the weight of 2. This heuristic comes from the observation that the knight has 8 move options in the center board and has fewer move options close to the edge of the board when there is nothing else in the way. So the game can lead to a win for me if I'm far away from any edge and the opponent is isolated to the edge of board and has no moves left.

**AB_Custom_3:**
In the beginning of the game (when the board is occupied by 12 or less knights), the evaluation heuristic uses the minimum distance of my position to any of the 4 board edges minus two times the minimum distance of the opponent's position to any of the 4 board edges (my_min_edge_distance - 2 * opponents_min_edge_distance) to evaluate the board score.
After the board is occupied by 12 more knights, it uses the number of my moves minus two times the number of the opponent's moves (#my_moves - 2 * #opponents_moves) to evaluate the board score.
This heuristic is a combination of AB_Custom and AB_Custom_2. This heuristic comes from the observation that in the beginning of the game if the two players are far apart, they can move in their own half of the game board oblivious to the other player and ignore the unoccupied center position on the first move. This heuristic was developed to overcome this by trying to move away from any edges in the beginning of the game when the game board is not heavily occupied and then switch to use AB_Custom when moves are narrowed down by more occupied spaces on the game board. Here is why we switch from AB_Custom_2 to AB_Custom when 12 spaces are occupied: Iterative deepening with Alpha Beta pruning usually evaluates 7 to 9 search depth for the first move; my first move is either the 3rd or 4th knight on the game board; the 9th search depth can result in 12 knights on the game board. In this way the first move will usually be chosen using the AB_Custom_2 evaluation heuristic.

## Performance results:

Two matches were played (see screenshots attached below) with 200 runs for each tournament between two agents. It shows AB_Custom has the highest win rate at 74.9% which is slightly higher than AB_Improved. The win rates of all four evaluation heuristics AB_Improved, AB_Custom, AB_Custom_2 and AB_Custom_3 fall between 73% and 75%.
If the number of runs for each tournament is 10 or 20, the win rates have a large variance. The win rate variances are much smaller when each tournament has 200 runs, but different match runs may still exhibit a 1% to 2% variance.

```
                         **************************
                              Playing Matches
                         **************************

 Match #    Opponent     AB_Improved    AB_Custom    AB_Custom_2  AB_Custom_3
                          Won | Lost    Won | Lost    Won | Lost    Won | Lost
    1        Random       198 |   2     198 |   2     197 |   3     200 |   0
    2        MM_Open      169 |  31     175 |  25     183 |  17     171 |  29
    3       MM_Center     194 |   6     192 |   8     194 |   6     193 |   7
    4     MM_Improved     162 |  38     169 |  31     164 |  36     171 |  29
    5        AB_Open      107 |  93     107 |  93     88  |  112    105 |  95
    6       AB_Center     110 |  90     118 |  82     111 |  89     114 |  86
    7     AB_Improved     101 |  99     89  |  111    88  |  112    93  |  107
-----------------------------------------------------------------------------
           Win Rate:      74.4%         74.9%         73.2%         74.8%
```

```
                         **************************
                              Playing Matches
                         **************************

 Match #    Opponent     AB_Improved    AB_Custom    AB_Custom_2  AB_Custom_3
                          Won | Lost    Won | Lost    Won | Lost    Won | Lost
    1        Random       197 |   3     198 |   2     197 |   3     195 |   5
    2        MM_Open      174 |  26     179 |  21     174 |  26     184 |  16
    3       MM_Center     197 |   3     190 |  10     195 |   5     191 |   9
    4     MM_Improved     165 |  35     163 |  37     165 |  35     166 |  34
    5        AB_Open      101 |  99     107 |  93     100 |  100    104 |  96
    6       AB_Center     110 |  90     113 |  87     108 |  92     109 |  91
    7     AB_Improved     96  |  104    99  |  101    92  |  108    90  |  110
-----------------------------------------------------------------------------
           Win Rate:      74.3%         74.9%         73.6%         74.2%
```

## Evaluation function recommendation:

AB_Custom function which uses (#my_moves - 2 * #opponents_moves) to evaluate the board score is recommended based on the following reasons:

1. It has the highest win rate according to the match results
2. It aggressively narrows down the opponent's move options by using the weight of 2
3. It's a meta strategy as opposed to AB_Custom_2. Because a knight close to the edge automatically results in fewer move options in comparison to a knight in the center of the game board if other occupied spaces are not considered
4. In competitions, I would recommend using an Opening Book that the game agent can use to quickly look up moves in the beginning of the game. Then use AB_Custom as the evaluation function in iterative deepening with Alpha Beta pruning to search close to the end game for each move. This would be a better overall strategy compared with using AB_Custom_3 with iterative deepening and Alpha Beta pruning for the entire game. This is because iterative deepening with Alpha Beta pruning very likely cannot search until the end game in the beginning of the game