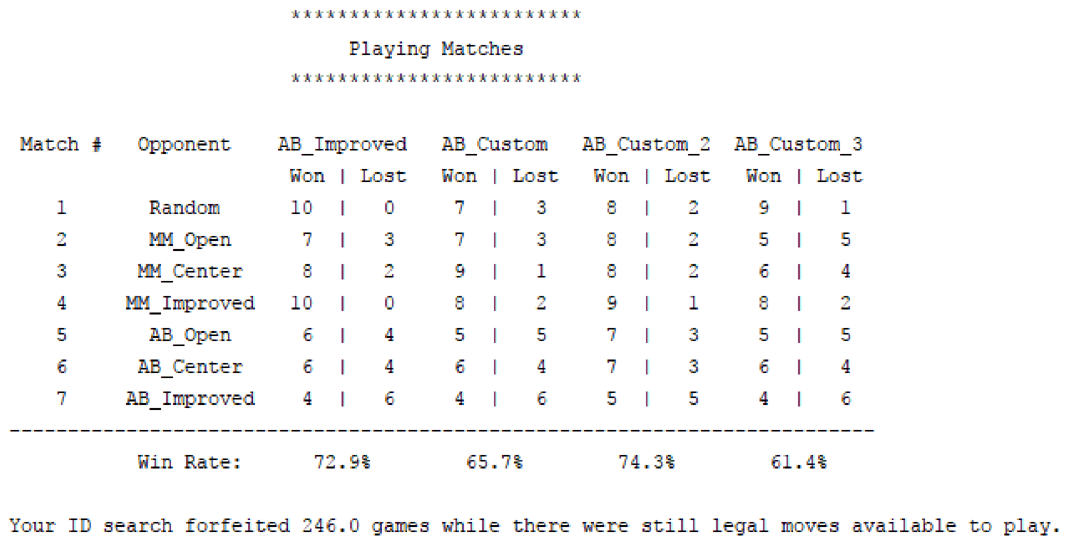
I tried 4 different heuristics –

- Custom Score #1: Return negative of number of opponent moves
    - Here, we are evaluating the best move by finding the move that leads to least number of opponent moves
- Custom Score #2: Return Number of Player moves – 2*No of Opponent moves
    - Here, we are evaluating the best move by finding the move that leads to better moves for the current player than the opponent player
- Custom Score #3: Return the difference between current and opponent Player's distance from the board center
    - Here, we are evaluating the best move by finding the move that brings the current player closer to the center than the opponent.
- Custom Score #4: Combination of score logic from #2 & #3
    - Here, we are evaluating the best move by using the combined logic from points 2 & 3 above.

<u>Comparison between Custom Score logic 1, 2 & 3:</u>

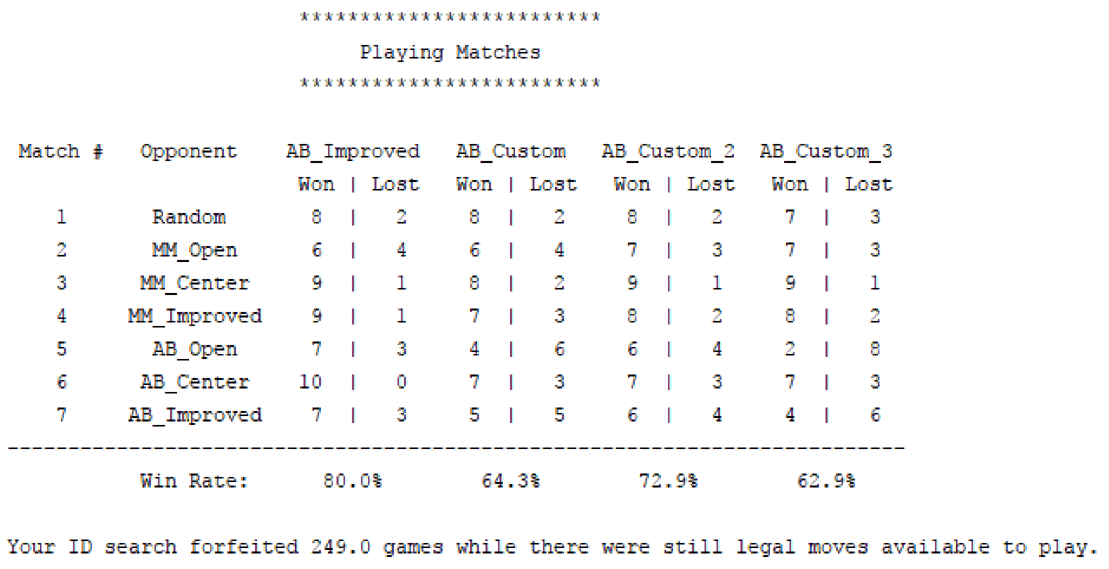


```
                         *************************
                              Playing Matches
                         *************************

 Match #   Opponent    AB_Improved   AB_Custom   AB_Custom_2  AB_Custom_3
                        Won | Lost   Won | Lost   Won | Lost   Won | Lost
    1       Random      10  |   0     7  |   3     8  |   2     9  |   1
    2       MM_Open      7  |   3     7  |   3     8  |   2     5  |   5
    3      MM_Center     8  |   2     9  |   1     8  |   2     6  |   4
    4     MM_Improved   10  |   0     8  |   2     9  |   1     8  |   2
    5       AB_Open      6  |   4     5  |   5     7  |   3     5  |   5
    6      AB_Center     6  |   4     6  |   4     7  |   3     6  |   4
    7     AB_Improved    4  |   6     4  |   6     5  |   5     4  |   6
--------------------------------------------------------------------------
           Win Rate:      72.9%        65.7%        74.3%        61.4%

Your ID search forfeited 246.0 games while there were still legal moves available to play.
```

<u>Comparison between Custom score logic 4, 2 & 3:</u>



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

 Match #   Opponent    AB_Improved   AB_Custom   AB_Custom_2  AB_Custom_3
                        Won | Lost   Won | Lost   Won | Lost   Won | Lost
    1       Random       8  |   2     8  |   2     8  |   2     7  |   3
    2       MM_Open      6  |   4     6  |   4     7  |   3     7  |   3
    3      MM_Center     9  |   1     8  |   2     9  |   1     9  |   1
    4     MM_Improved    9  |   1     7  |   3     8  |   2     8  |   2
    5       AB_Open      7  |   3     4  |   6     6  |   4     2  |   8
    6      AB_Center    10  |   0     7  |   3     7  |   3     7  |   3
    7     AB_Improved    7  |   3     5  |   5     6  |   4     4  |   6
--------------------------------------------------------------------------
           Win Rate:      80.0%        64.3%        72.9%        62.9%

Your ID search forfeited 249.0 games while there were still legal moves available to play.
```

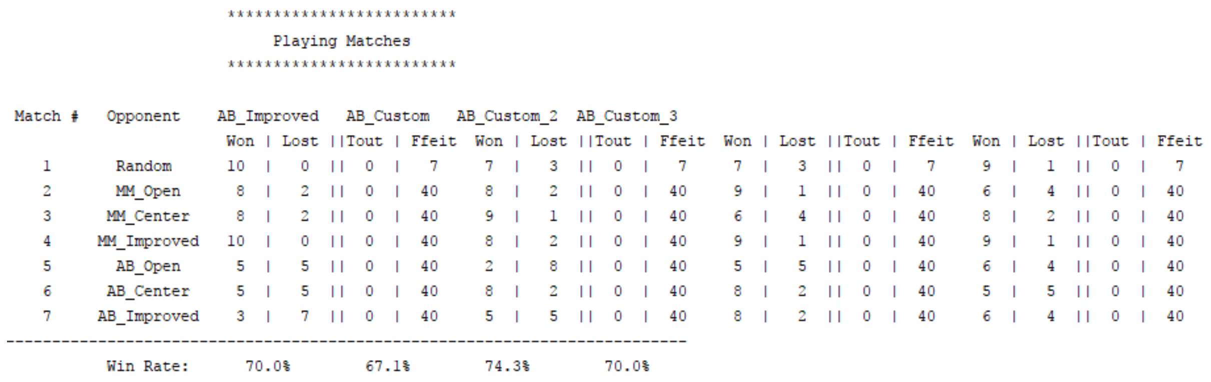
<u>Observations & Analysis</u>:

From all the scoring functions that I tried, custom scoring function #2 ("Number of Player moves – 2*No of Opponent moves") gives the best performance since it compares the available options of both the current player and the opponent. Custom Score logic #4 combines the best of #2 & #3 but is less effective (than #2) because its computation logic takes much longer resulting in exploration of fewer levels (and nodes by iterative deepening).

<u>Recommendations:</u>

Based on my analysis, I would like to recommend my evaluation function #2 which is "my_moves – 2*my_opponent_moves". The main reasons are –

- Better performance (win rate) compared to the other evaluation functions that I tested
- This weighs the opponents_moves higher than the current player's moves for more aggressive game play and chases the opponent
- The timeouts/forfeits isn't bad with this approach and it compares well with other approaches. I updated tournament.py to display the timeouts & forfeits for my evaluation functions 1, 2 & 3. Here are the observations (didn't adjust headers & footer rows)



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

 Match #   Opponent    AB_Improved   AB_Custom   AB_Custom_2  AB_Custom_3
                        Won | Lost ||Tout | Ffeit  Won | Lost ||Tout | Ffeit  Won | Lost ||Tout | Ffeit  Won | Lost ||Tout | Ffeit
    1       Random      10  |   0  ||  0  |   7     7  |   3  ||  0  |   7     7  |   3  ||  0  |   7     9  |   1  ||  0  |   7
    2       MM_Open      8  |   2  ||  0  |  40     8  |   2  ||  0  |  40     9  |   1  ||  0  |  40     6  |   4  ||  0  |  40
    3      MM_Center     8  |   2  ||  0  |  40     9  |   1  ||  0  |  40     6  |   4  ||  0  |  40     8  |   2  ||  0  |  40
    4     MM_Improved   10  |   0  ||  0  |  40     8  |   2  ||  0  |  40     9  |   1  ||  0  |  40     9  |   1  ||  0  |  40
    5       AB_Open      5  |   5  ||  0  |  40     2  |   8  ||  0  |  40     5  |   5  ||  0  |  40     6  |   4  ||  0  |  40
    6      AB_Center     5  |   5  ||  0  |  40     8  |   2  ||  0  |  40     8  |   2  ||  0  |  40     5  |   5  ||  0  |  40
    7     AB_Improved    3  |   7  ||  0  |  40     5  |   5  ||  0  |  40     8  |   2  ||  0  |  40     6  |   4  ||  0  |  40
--------------------------------------------------------------------------
           Win Rate:      70.0%        67.1%        74.3%        70.0%
```