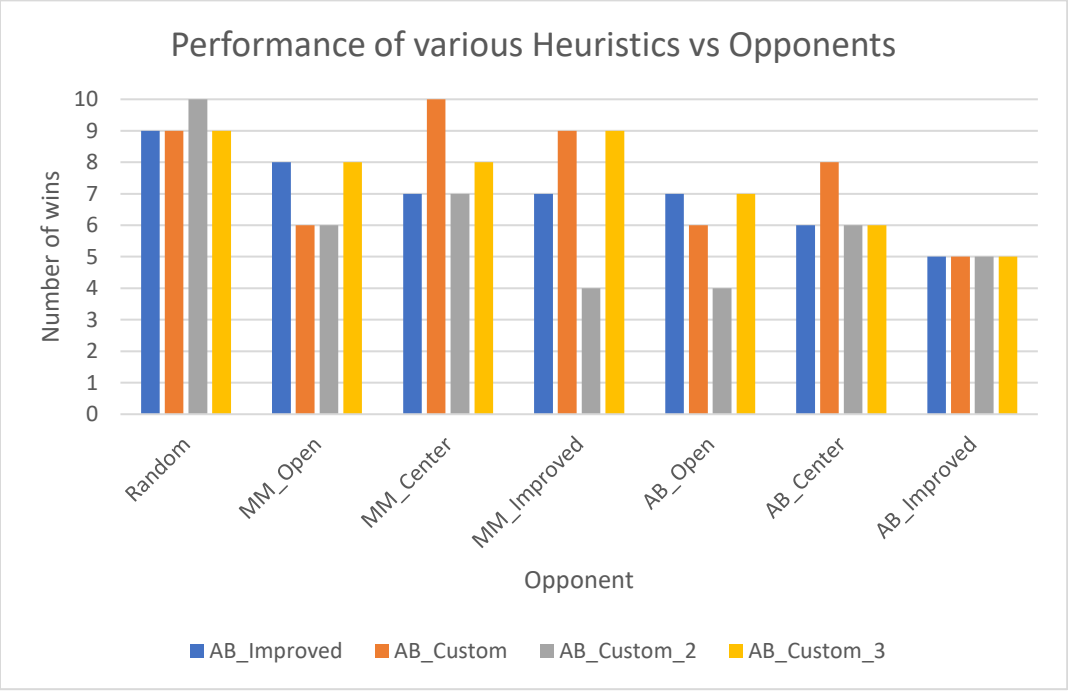
```
**************************
    Playing Matches
**************************
```

Match #	Opponent	AB_Improved		AB_Custom		AB_Custom_2		AB_Custom_3	
		Won	Lost	Won	Lost	Won	Lost	Won	Lost
1	Random	9	1	9	1	10	0	9	1
2	MM_Open	8	2	6	4	6	4	8	2
3	MM_Center	7	3	10	0	7	3	8	2
4	MM_Improved	7	3	9	1	4	6	9	1
5	AB_Open	7	3	6	4	4	6	7	3
6	AB_Center	6	4	8	2	6	4	6	4
7	AB_Improved	5	5	5	5	5	5	5	5
	Win Rate:	70.0%		75.7%		60.0%		74.3%	

The three heuristic functions that I used are as follows:

1) AB_Custom: #legal_moves_player – 2*#legal_moves_opponent
2) AB_Custom 2: (2*#legal_moves_opponent ^2 – 3*#legal_moves_opponent^3)
3) AB_Custom 3: 2*#legal_moves_player – 5*(#legal_moves_opponent^2)



The best performing agent among the above 3 is the agent AB Custom. The reasons are as follows:

1) By weighting the # of legal moves of the opponent more, the win rate increases. This is because AB_Improved had a heuristic which was the difference in the number of moves of the player and the opponent.
2) Adding a power to the # of legal moves of the opponent seems to have a comparable rate as the linear relation. This shows that it doesn't matter much if the powers of the number of moves are changed. What matters is that relationship which tries to boost the number of players moves and tries to reduce the number of opponent's moves is captured.
3) Trying to reduce the opponent's moves alone did not give good enough results. A balance of player's moves and opponent moves should be considered.

   There could be better heuristics that may be tried which try to take into account the positions of the pieces on the board as well.