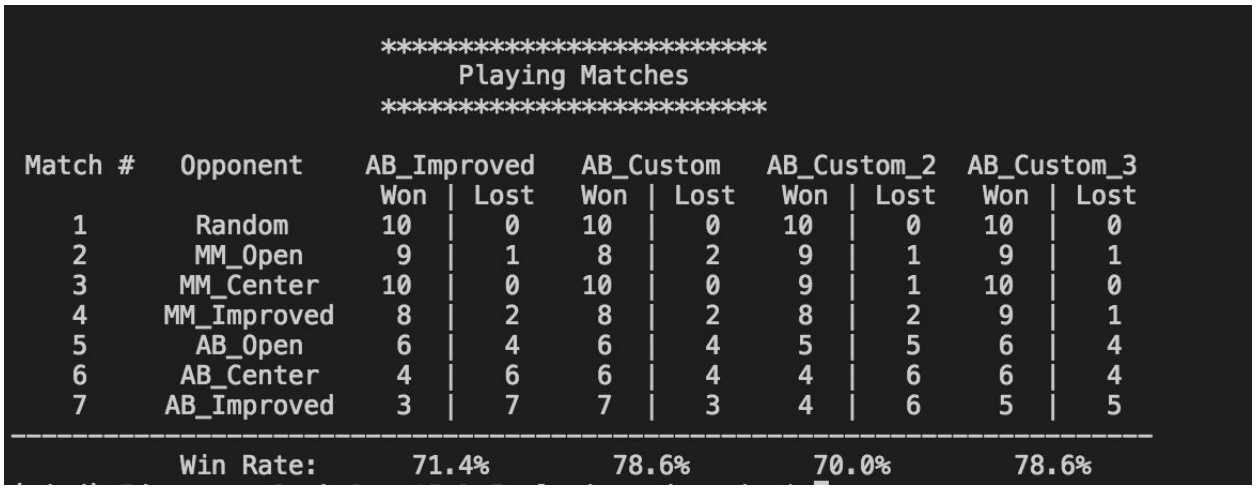
# Performance of the Heuristic Functions



```
                    **************************
                         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    10  |   0    10  |   0    10  |   0
   2       MM_Open       9  |   1     8  |   2     9  |   1     9  |   1
   3      MM_Center     10  |   0    10  |   0     9  |   1    10  |   0
   4     MM_Improved     8  |   2     8  |   2     8  |   2     9  |   1
   5       AB_Open       6  |   4     6  |   4     5  |   5     6  |   4
   6      AB_Center      4  |   6     6  |   4     4  |   6     6  |   4
   7     AB_Improved     3  |   7     7  |   3     4  |   6     5  |   5
--------------------------------------------------------------------------
          Win Rate:      71.4%        78.6%        70.0%        78.6%
```

custom_score: This heuristic function subtracts the number of available opponent moves from the player. I choose this function because the more moves I had the higher the chances of winning. Overall, it's pretty simple but effective. It had a 78.6% winning rate which was not bad for just a simple function. When running the tournament script, the function performed well vs Random, MM_Open, MM_Center, and MM_Improved opponents but not so well with AB_Open, AB_Center, and AB_Improved. I believe that the function lost most of the games vs the AB_Center due to my game agent no selecting the center move or one that reacts properly to center moves.

custom_score_2: This heuristic function returns 1 if the number of available moves is greater than the opponents or 0 if not. I choose this function as it will return 1 if the player has higher chances of winning. Even though this evaluation function is simple, it achieves a winning rate of 70.0%. It did well on most opponents except the AB_Improved. I believe this is due to my function returning 1 as long as the player has more number of moves.

Custom_score_3: This heuristic function returns the number of moves divided by the number of opponent moves available. I used 1 as a fallback if the opponent has no available moves. The reason I choose this function is because the returned number would be high if the player has high number of available moves and the opponent has less. This heuristic function achieves a winning rate of 78.6% which is similar to my first function. I think it had a similar score because I'm dividing the same numbers. Like my previous functions, it performed badly vs the AB_Open, AB_Center,  and AB_Improved. This is probably due to my agent no selecting moves optimized for center and reflection moves.