AIND PROJECT 2 Heuristic review

1. Custom (Ultimately chosen one)
   - Return own_moves – opp_moves
2. Custom_2
   - Return own_moves / (opp_moves + 1)
3. Custom_3
   - Return 3 * own_moves + 1 / (opp_moves + 0.3)

I finally chose the heuristic that difference of own moves and opponent moves. It's very simple and easy to calculate, but it is also reasonable enoughly.

```
**************************
    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     8  |  2     9  |  1     9  |  1
   2       MM_Open      7  |  3     7  |  3     8  |  2     6  |  4
   3      MM_Center     7  |  3     9  |  1     7  |  3     7  |  3
   4     MM_Improved    8  |  2     7  |  3     6  |  4     8  |  2
   5       AB_Open      6  |  4     5  |  5     6  |  4     5  |  5
   6      AB_Center     7  |  3     7  |  3     5  |  5     7  |  3
   7     AB_Improved    6  |  4     8  |  2     5  |  5     4  |  6
----------------------------------------------------------------------
          Win Rate:      72.9%       72.9%       65.7%       65.7%
```

One of the matches result

'tournarment.py' simulates 10 isolation games with each heuristic, and all heuristics perform well against to Random opponent, Center opponent., but it seems a bit ahead or equal to AB-Center, AB-Improved. Since my laptop specification is not good, Win rate changed a lot in each simulation.

Although I have not implemented it because of lack of my coding ability, I predict that more complicated heuristic which gives a penalty if the player gets closer to the corner, or considering the case of partitioning can show much better performance.