In this project, I have implemented 3 heuristics. The reasons of choosing these three heuristics are shown below:

1. Number of my legal moves not in corners - Number of my legal opponent's' not in corners
   - First obtained the legal moves of me and my opponent
   - Then calculated how many of them are in corners of the game board
   - Returns the difference between number of my legal moves not in corners and number of my legal opponent's' not in corners
2. Number of my moves - 2*number of my opponent's' moves
   - Returns the difference between the number of available moves of computer player and twice the number of available moves to opponent. The multiplier of 2 is to add a penalty for having more moves of opponents.
   - Align with the goal of this game, which is to maximize the number of moves of computer player while minimize the number of moves of opponents.
3. - (the distance from my current location to center - the distance from opponent's current location to center)
   - First calculate the distance from my current location to center and the distance from opponent's current location to center respectively
   - Then obtained the difference between the two

The output of running tournament.py is as follows:
```
*************************
 Playing Matches
*************************
```

Match #	Opponent	AB_Improved Won | Lost	AB_Custom Won | Lost	AB_Custom_2 Won | Lost	AB_Custom_3 Won | Lost
1	Random	8 | 2	8 | 2	8 | 2	7 | 3
2	MM_Open	4 | 6	5 | 5	5 | 5	7 | 3
3	MM_Center	9 | 1	6 | 4	8 | 2	7 | 3
4	MM_Improved	7 | 3	7 | 3	7 | 3	6 | 4
5	AB_Open	5 | 5	6 | 4	4 | 6	4 | 6
6	AB_Center	5 | 5	7 | 3	5 | 5	7 | 3
7	AB_Improved	6 | 4	5 | 5	6 | 4	4 | 6

| | Win Rate: | 62.9% | 62.9% | 61.4% | 60.0% |

Finally, I chose AB_Custom because its performance is the best among all evaluation functions. Also, there are three other reasons that help me made my final decision:

- Compared with other two heuristics, it considers not only number of legal moves but also how many of them are in corners. In other words, it goes deeper than the other two heuristics in terms checking what are the available moves not in the corners.
- It is an easy and simple function so it has computational benefit compared with complex heuristics.
- for a player to win, this play does not want to separate his/her moves, rather, he/she would cluster them. Thus, this heuristic performs better cause that it checks all the legal moves and categorized them into corners and non-corners.