Heuristic Analysis

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

Match #   Opponent    AB_Improved   AB_Custom   AB_Custom_2  AB_Custom_3
                      Won | Lost   Won | Lost   Won | Lost   Won | Lost
   1       Random      7  |  3     10  |  0     8  |  2     9  |  1
   2       MM_Open     5  |  5      8  |  2     5  |  5     8  |  2
   3      MM_Center    8  |  2      8  |  2     6  |  4     8  |  2
   4     MM_Improved   5  |  5      8  |  2     6  |  4     7  |  3
   5       AB_Open     5  |  5      6  |  4     3  |  7     2  |  8
   6      AB_Center    6  |  4      7  |  3     5  |  5     5  |  5
   7     AB_Improved   5  |  5      6  |  4     6  |  4     5  |  5
----------------------------------------------------------------------
          Win Rate:      58.6%        75.7%        55.7%        62.9%
```

Custom_1 seemed to perform better against all the different oponents.

I will start with the analysis of Custom_3 as 1 and 2 use it in addition to their own score calculation code.

Custom_3 determines the score of a move based on its distance from the center. 1 for the center and 0 for the furthest positions. My reasoning behind this is that even though I have not done the math, my intuition tells me that if the player stays as near as possible to the center, its future moves should have more options as they would be further away from the boundaries. I selected a range between 0 and 1 so I would be able to use it in the Custom 1 and 2 functions without it affecting their results too much.

Custom_1 bases its score in the amount of open moves from the player minus the number of open moves from the opponent and then it adds the score from Custom_3 for optimization. My reasoning behind this is simply that as a general rule increasing the number of future open moves for the player should always give better chances of winning and reducing the amount of open moves for the opponent should increase the chance of the opponent loosing.

Custom_2 bases its score in the amount of open moves from the player and then it adds the score from Custom_3 for optimization. My reasoning behind this is simply that as a general rule increasing the number of future open moves for the player should always give better chances of winning.