Different heuristics were tested in this game. My goal is to develop a heuristic such that Student outperforms ID_Improved. In all cases, this was achieved. I ultimately selected to use the first one in my submitted agent, as the Student outperforms by over 9%.

1. own_moves - opp_moves
    a. ID_Improved 72.14%
    b. Student 81.43%
2. 1/(opp_moves+1)+own_moves)
    a. ID_Improved 70.00%
    b. Student 76.43%
3. own_moves^2 - opp_moves^2)
    a. ID_Improved 79.29%
    b. Student 80.71%

Output:
return float(own_moves^2 - opp_moves^2)

```
************************
 Evaluating: ID_Improved
************************

Playing Matches:
----------
  Match 1: ID_Improved vs   Random      Result: 16 to 4
  Match 2: ID_Improved vs   MM_Null     Result: 18 to 2
  Match 3: ID_Improved vs   MM_Open     Result: 15 to 5
  Match 4: ID_Improved vs MM_Improved           Result: 11 to 9
  Match 5: ID_Improved vs   AB_Null             Result: 15 to 5
  Match 6: ID_Improved vs   AB_Open     Result: 18 to 2
  Match 7: ID_Improved vs AB_Improved           Result: 18 to 2


Results:
----------
ID_Improved         79.29%

************************
   Evaluating: Student
************************

Playing Matches:
----------
  Match 1:   Student   vs   Random              Result: 18 to 2
  Match 2:   Student   vs   MM_Null             Result: 18 to 2
  Match 3:   Student   vs   MM_Open             Result: 12 to 8
  Match 4:   Student   vs MM_Improved   Result: 15 to 5
  Match 5:   Student   vs   AB_Null             Result: 16 to 4
  Match 6:   Student   vs   AB_Open             Result: 14 to 6
  Match 7:   Student   vs AB_Improved    Result: 20 to 0


Results:
----------
Student             80.71%
[Finished in 319.5s]
```

return float(own_moves - opp_moves)

```
*************************
 Evaluating: ID_Improved
*************************

Playing Matches:
----------
  Match 1: ID_Improved vs   Random      Result: 17 to 3
  Match 2: ID_Improved vs   MM_Null     Result: 14 to 6
  Match 3: ID_Improved vs   MM_Open     Result: 13 to 7
  Match 4: ID_Improved vs MM_Improved           Result: 9 to 11
  Match 5: ID_Improved vs   AB_Null             Result: 16 to 4
  Match 6: ID_Improved vs   AB_Open     Result: 16 to 4
  Match 7: ID_Improved vs AB_Improved           Result: 16 to 4


Results:
----------
ID_Improved         72.14%

*************************
   Evaluating: Student
*************************

Playing Matches:
----------
  Match 1:   Student   vs   Random              Result: 15 to 5
  Match 2:   Student   vs   MM_Null             Result: 18 to 2
  Match 3:   Student   vs   MM_Open             Result: 16 to 4
  Match 4:   Student   vs MM_Improved   Result: 14 to 6
  Match 5:   Student   vs   AB_Null             Result: 17 to 3
  Match 6:   Student   vs   AB_Open             Result: 14 to 6
  Match 7:   Student   vs AB_Improved   Result: 20 to 0


Results:
----------
Student             81.43%
[Finished in 330.0s]
```

return float(1/(opp_moves+1)+own_moves)

```
Playing Matches:
----------
  Match 1: ID_Improved vs   Random      Result: 16 to 4
  Match 2: ID_Improved vs   MM_Null     Result: 17 to 3
  Match 3: ID_Improved vs   MM_Open     Result: 9 to 11
  Match 4: ID_Improved vs MM_Improved           Result: 11 to 9
  Match 5: ID_Improved vs   AB_Null             Result: 13 to 7
  Match 6: ID_Improved vs   AB_Open     Result: 15 to 5
  Match 7: ID_Improved vs AB_Improved           Result: 17 to 3


Results:
----------
```

```
ID_Improved         70.00%

*************************
  Evaluating: Student
*************************

Playing Matches:
----------
  Match 1:   Student   vs   Random                Result: 17 to 3
  Match 2:   Student   vs   MM_Null               Result: 15 to 5
  Match 3:   Student   vs   MM_Open               Result: 14 to 6
  Match 4:   Student   vs MM_Improved   Result: 11 to 9
  Match 5:   Student   vs   AB_Null               Result: 15 to 5
  Match 6:   Student   vs   AB_Open               Result: 17 to 3
  Match 7:   Student   vs AB_Improved    Result: 18 to 2


Results:
----------
Student             76.43%
[Finished in 328.6s]
```