Build a Game Playing Agent - Heuristics Analysis

The implementations of the following search algorithms were tested in similar conditions –

- Minimax search
- Alphabeta search
- Iterative deepening with both MM and AB

The only difference being the randomness of the start of the play between the Student and the Improved ID players.

The results from the multiple runs averaged out across the four heuristics functions are as follows –

Heuristic Name	ID_Improved Score (%)	Student Score (%)	Heuristics Details
Heuristic A	72.14	70.71	Worst Performance of the lot. If the Student has fewer move options than the opponent, the function returns: - (# moves the opponent has)/(# moves the computer has)**2. If the Student has more move options than the opponent, the function returns (# moves the opponent has)/(# moves the computer has)**2. The logic for this is similar. If a move causes the computer to have greatly proportionally fewer moves than the opponent, it is a bad move.
Heuristic B	72.14	73.57	Aggressive play in the first half of the game. Active player will try to choose the most aggressive move. Heuristic calculates number of players move vs 3.5 of value of an opponent's moves. In the second half of the game heuristic will calculate number of players move vs number of an opponent's moves.
Heuristic C	72.14	76.43	Best Heuristic from the test results, most aggressive initially, then drop aggressiveness at 1/3 moves and further more at 1/4 moves remaining.
Heuristic D	72.86	75	Least aggressive initially, then increase aggressiveness at 1/3 and further more at 1/4 moves remaining.

From the observation running these tests over time, being aggressive initially helped with focusing on the good lines in the tree and subsequent releasing the aggression helped with a broader democratic/equitable return of evaluation results.

The compact nature of the implementation may have helped with processing deeper with the limited resources on my Apple macbook pro with the following config. This may have helped with beating the ID Improved with additional lightweight evaluation functions implementations.