Agent	Adversary	Program args	Wins
Monte Carlo Tree Search	Minimax	Fair, 50 rounds	85.50%
Minimax (alpha-beta, iterative deepening, #my_moves - #opponent_moves)	Minimax	Fair, 50 rounds	78.50% <= Baseline

The Baseline algorithm Minimax with iterative deepening was chosen because it is optimal in the long run and can be easily sped up with alpha-beta pruning as well as simple heuristics. Hence, it performs very good against a pure minimax adversary.
The Monte Carlo Tree Search algorithm sacrifices certainty and bases it's decision on probability. Since it plays a complete game with every iteration there are less surprises in the endgame and chances are it finds killer branches early on. But because the branching factor is high in the beginning, great parts of the tree will be thrown away and the first moves are risky. The further the game progresses though the higher the certainty becomes and should approach minimax quicker than the baseline algorithm. MCTS performance should be able to improve significantly with an opening book.