Mark Demore II

CSCE523

Assignment #2

1. A)

B)

C)

D)

1. A)

B)

C)

1. Compile Instructions: Open the IntelliJ project and click run. (I believe the only files I altered were).

Description:

1. Implementation: I implemented my alpha-beta minimax using a recursive helper function. The helper is fed each child of the current board state and evaluates subsequent states using depth, a Boolean determining the maximizing player, and the alpha and beta values to prune different subtrees.
2. Heuristic: My heuristic makes use of the Euler number and Quad counts that are calculated elsewhere in the program. Based on Winand’s evaluation function, the heuristic favors a smaller Euler number and Quads of 3 or 4 pieces, with a smaller preference for Quads of 2 pieces. The Euler heuristic takes thee Euler number and scales it using tanh, it is then subtracted from one. The Quad heuristic increases by 2 with every quad of 3 or 4 pieces, and by 1 with every quad of 2 pieces, it is then scaled with tanh. The final heuristic value returned for each board is the tanh of the sum of the Euler and Quad heuristics. It returns a -1 if the board is a loss, and a 1 if the board is a win, in place of the heuristic.
3. Experiences: The heuristic is not terribly good at winning, but not totally ignorant. I attempted scaling the Euler and Quad heuristics in relation to each other by doubling them before adding for the final heuristic calculation, but there was no observable performance increase. I also attempted subtracting the opponents corresponding heuristic as part of the calculation, but this incurred a substantial time cost at depths greater than 3.