Gavin Bolin

CS 470

Expectimax Lab

1. For depth limits of 1-5, how long does it take your expectimax code to find a move at the beginning of the game?

For this part each section

1	2	3	4	5
~0.03s	~0.25s	~0.35s	~1.5s	~10.8s

2. Play against your agent at several depth limits. At what depth limit does your program beat you (if any)?

Compared to the alpha beta algorithm it is much better at countering my moves and stops me from making any advancements. Its interesting that it works better than the previous but makes sense as it takes account of all possible moves from that branch rather than just one. After about depth level 3 it starts to beat me rather than make moves that don't make sense. Sometimes it still slips up at depth 3 but after that its harder to get tot make a mistake.

3. How does the play style of this agent differ from your alpha-beta agent?

This agent compares to the alpha beta agent because it is a lot better at reacting and playing form behind rather than trying to make its own connections. When printing out the evaluation values for each iteration the evaluation value is many times negative if the ai is second meaning it is thinking it is behind and is able to still beat me when it went second.

4. Play your agent with different depth limits against a random agent, for 10 games. Does depth impact its ability to win?

When running the agent against the random agent it was still difficult for the expectimax agent to still win every time but for the most part it would win almost every time. However depth level seemed to be irrelevant and would win around the same amount every time.

5. Play your alpha-beta agent against your expectimax agent for 10 games. Do this for several combinations of depth-limits. How does depth of searches impact the effectiveness of the agent?

The expectimax seemed like it won a lot more than the alpha beta algorithm when running the game initially but after reviewing the results it seemed to win around the same amount with the same depth level. Even with further depth levels it seems to win around the same amount as long as the depth level is the same. This makes sense as the algorithms are very similar and having the expectimax generalize the outputs for a certain iteration would help it to know which initial move would be the best but alpha beta does the same expect it doesn't account for all iterations for the last but finds the best single score and makes its move based on that.

6. How long did this assignment take you? Were any portions especially frustrating? Anything you would change next time around?

This assignment took me probably a collective 4 hours to complete. It was much easier than the previous assignment as the evaluation function was the same though it needed some tweaks as I would set the return of a 4 in a row in an iteration to be infinity rather than multiplying the amount by an arbitrarily large number alone. I struggled to understand which player to have run each time for the expectimax and was causing some trouble bug testing. Bug testing was the hardest as running in terminal without a debugger is a littler harder but using many print statements was very helpful.