

Speaker: <Daniel Guzman> (Minimax)

Reviewer 1 (Joseph Shifman): Cool algorithm with cool applications, I want more in-depth explanation on how the decision tree is formed and how it comes into play

Thank you, the decision tree is formed through how the Minimax algorithm recursively calls itself when there is a move available on the board in order to figure out if that is the best move for the A.i

Reviewer 2 (Marcus Naredo): Sounds interesting, remember something similar from AI. Maybe more information on how the implementation works, although it sounds like it's not fully implemented yet.

Thank you, and yes at the time it wasn't completely implemented yet.

Reviewer 3 (Isael Melchor): very interesting implementation, especially the idea of an AI knowing its percentage of winning or losing a match,

Thank you

Reviewer 4 (Dalton Bailey): Very interesting idea, and explained well. I like the idea of using a decision tree to create a game AI.

Thank you, and yes the decision tree greatly helps visualize how the algorithm works.

Reviewer 5 (Derek Borders):

- What parts of the presentation were most effective?
 - Nice visuals. Nice
- Did you find any aspects of the algorithm particularly confusing or unclear? If so, what might help to clarify those parts?
 - How are scores calculated?

Good question, the score of a state is calculated by counting the most amount of pieces lined up either within a row, column or diagonal.

- After the presentation, do you know why the algorithm is important, what the general idea behind the algorithm is, and how efficient it is asymptotically?
 - I think so. I think I've seen versions of this before too.