Minmax

In our Smoothie Game which players are prompted to use and pick the best ingredients for their smoothies and who ever makes the best one wins, we have decided to implement the Minmax Ai algorithm. As the Minimax algorithm is a decision rule which imitates the decisions of a non-AI player to find the optimal move for itself. This algorithm assumes the player will choose the optimal choice as well.

Minmax is an algorithm usually used in games such as Chess, Tic Tac Toe, and checkers and various two player/ turn based games for example. This is an optimal algorithm to use in our game, as this requires players taking correlating turns.

As the minimax algorithm is a recursive or backtracking algorithm which is often used in decision-making games, our game which mainly requires decision-making, as players have to randomly choose between a set of cards, to acquire the best ingredients for their smoothies. That requires decision-making, Players have to make decisions which will be beneficial or detrimental to their success.

In the Minmax algorithm there are two what they call "actors" in the Minimax algorithm. It's "Maximiser" and it's "Minimiser". This is a very simple aspect to understand as the maximiser will choose the best score possible, in this case the Ai will choose the best ingredient for their smoothie. The minimiser will choose the worst score, so to put into context, the AI will choose the worst ingredient possible for their smoothie. In short maximiser will try to get the best ingredients for their smoothie in order to win, and minimiser will get the worst in order for the player to win easily.

We can use that simple logic and implement that into our difficulty levels for our game. As when the player selects the easy difficulty, the AI will essentially choose bad ingredients, and hard difficulty the AI will select the best ingredients. When the player implements the medium difficulty level, we will have the AI randomly implement a variation of selecting good and bad ingredients, making it technically an even playing field between player and AI.

As our game is just having players choose from random cards. When playing against ai, we don't want the player to feel potentially frustrated with our game, if the AI always chooses the best ingredients, as to why the implementation of Minmax is crucial, we implement difficulty levels to ensure that the AI increases in intelligence appropriately in terms of the ingredients it will choose. It will also give the player an understanding of what to expect from the AI in each difficulty level. Cause if the player chooses the hard difficuly it won't be frustrating if the AI always wins, cause the player will already be prepared for the AI to choose the best ingredients.

Implementing the Minmax approach will assure that the AI plays optimally, assuming the player plays optimally as well.

Referencing

javatpoint.com. 2021. Mini-Max Algorithm in Artificial Intelligence, n.a. [Online]. Available at: https://www.javatpoint.com/mini-max-algorithm-in-ai [Accessed 10 June 2024].

Wibowo, HA. 2018. Create AI for Your Own Board Game From Scratch — Minimax — Part 2, 31 October 2018. [Blog]. Available at: https://towardsdatascience.com/create-ai-for-your-own-board-game-from-scratchminimax-part-2-517e1c1e3362 [Accessed 10 June 2024].

*Minimax Algorithm for Tic Tac Toe (Coding Challenge 154).*2019. YouTube video, added by The Coding Train. [Online]. Available at: https://www.youtube.com/watch?v=trKjYdBASyQ [Accessed 22 March 2024].

*Minimax: How Computers Play Games.*2023. YouTube video, added by Spanning Tree. [Online]. Available at: https://www.youtube.com/watch?v=SLgZhpDsrfc [Accessed 22 March 2024].

GITHUB LINK

https://github.com/Hakeemshaik/GADE7321_POE