CS-470: Artificial Intelligence: Project 2

Taylor Martin
University of Idaho
Moscow, Idaho, USA
mart8517@vandals.uidaho.edu

1 ABSTRACT

This project utilized AI searching and evaluation techniques from the field of Artificial Intelligence, specifically from the content presented in CS-470. The primary objective focused on creating a connect four game and creating an AI agent for a human to play the connect four game against. To accomplish this objective, the MinMax searching and evaluation algorithm was implemented with alpha-beta pruning. The adopted methodology exhibited highly favorable outcomes, thereby instilling a high level of confidence in the generated results.

2 MINMAX ALGORITHM

2.1 MinMax Description

The implementation of the MinMax algorithm is a crucial component of the project and resides in the agent class within the agent module. The primary method of this class, called "best move," accepts the current player's color and the difficulty setting as parameters. It initializes a dictionary of legal moves, with one entry for each available column (excluding full columns). Using a "simulate move" method, it calculates the resulting board state if each move were to be made and passes it to the recursive search function. The algorithm stores the evaluated values for each move in the dictionary and ultimately returns the move with the highest alpha value as the optimal move for the AI agent.

2.2 Depth Of Search

The search method within the MinMax algorithm takes parameters such as the search depth (interpreted as the AI's difficulty level, I had it set at 10000), the current board state, and the current player. As the search progresses, the method maintains a list of the next set of legal moves, where each move is represented by the resulting board state if that move were executed. The search method utilizes the depth limit as a measure of difficulty. A higher depth limit allows for a deeper search, resulting in better moves returned by the algorithm.

2.3 Extra Features

To improve the efficiency of the algorithm, the search method employs alpha-beta pruning. If the search reaches a terminal state (e.g., the maximum depth is reached, no more moves are possible due to a full board, or an end game condition is met), the method returns. Otherwise, for each board state in the search method's list, recursive search is called while comparing and updating the maximum alpha and minimum beta values. This pruning technique eliminates suboptimal branches in the search tree, reducing unnecessary computations.

2.4 Evaluation Function

The evaluation function used in the MinMax algorithm assigns a score to each move based on the number of token streaks it generates. Four-token streaks are valued at ten thousand, three-token streaks at one thousand, and two-token streaks at ten. The evaluation function calculates the sum of these values to determine the overall score for a move.

2.5 How The Program Handles Win & Loss Encounters

The program effectively handles win and loss scenarios, demonstrating a strong strategic approach. The AI agent consistently attempts to block the opponent's potential wins and actively pursues its own winning paths. It employs tactics to trap opponents into disadvantageous positions, displaying a persistent nature without conceding easily. While I have achieved victories against the AI on multiple occasions, I observed that by fine-tuning the heuristic and increasing the difficulty level or search depth, the AI gradually improved its gameplay, ultimately surpassing my abilities. This highlights the AI's adaptability and ability to elevate its performance as the difficulty settings are enhanced.

2.6 Gameplay Screenshots

Figure 1: Starting Connect 4. Playing Against Computer

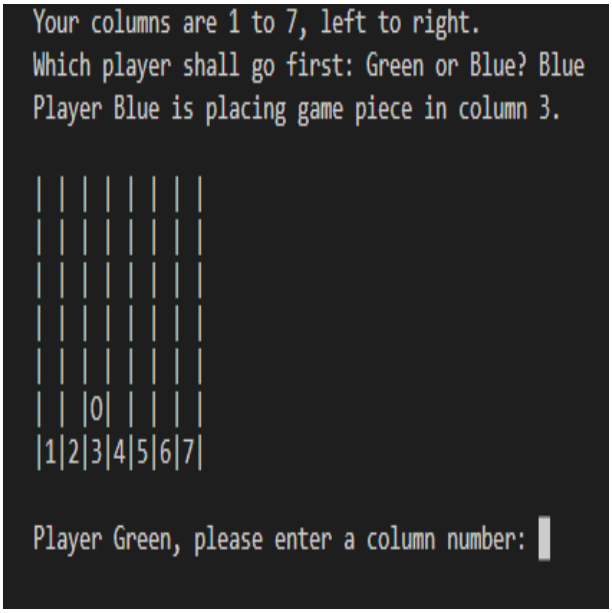

Figure 2: Common Computer Opening Move

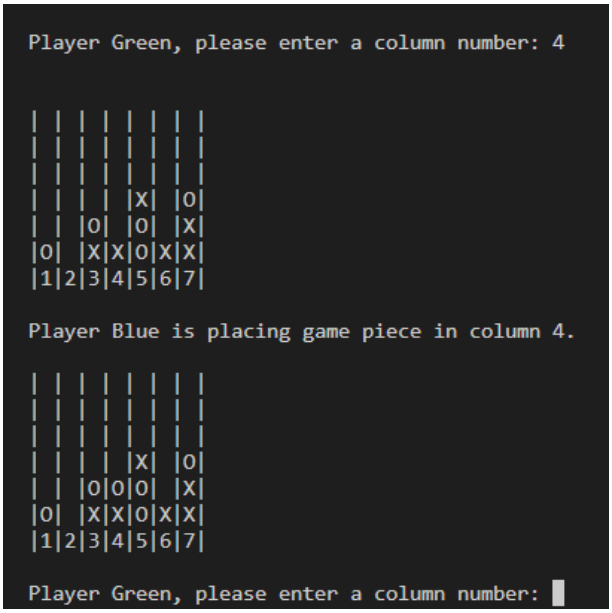

Figure 4: AI Diagonal Streak Block

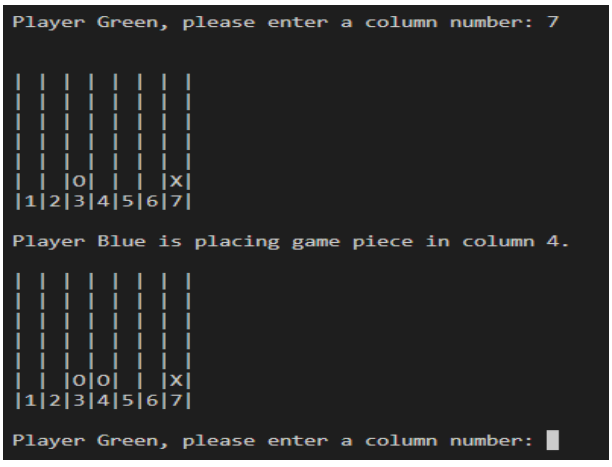

Figure 3: Human Player Move

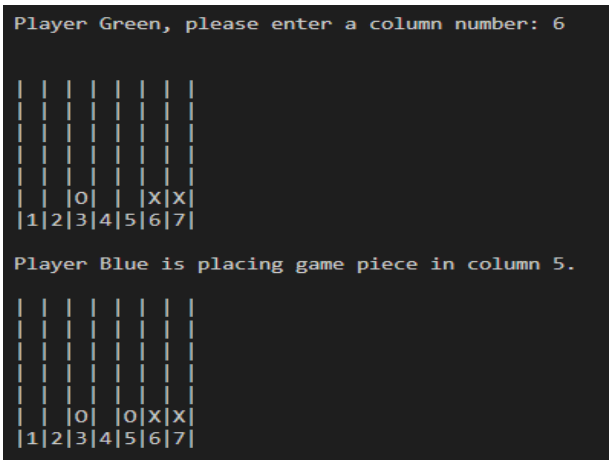

Figure 5: AI Horizontal Streak Block

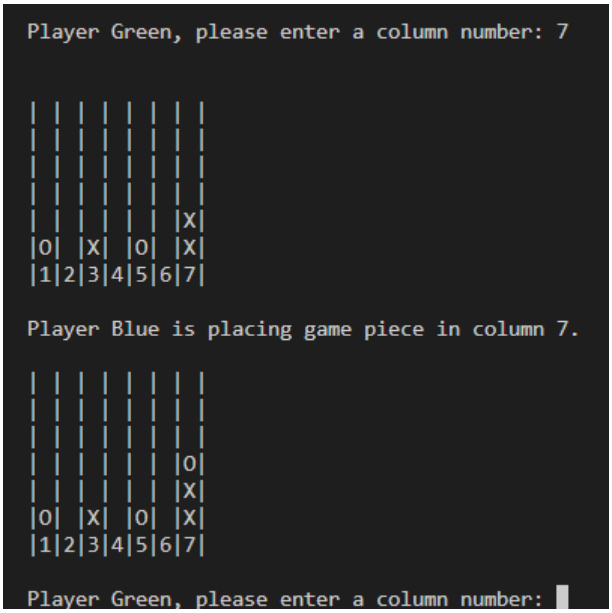

Figure 6: AI Vertical Streak Block

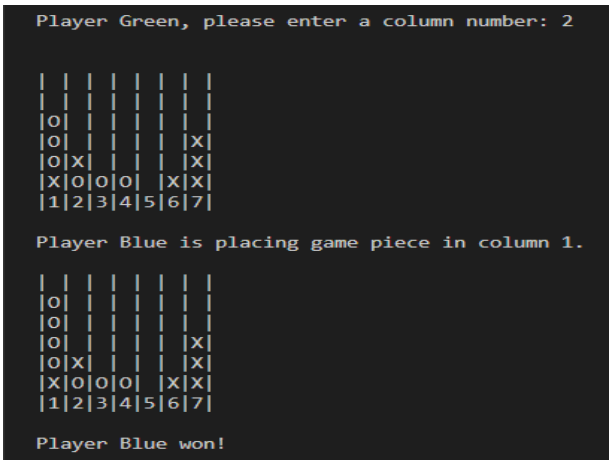

Figure 7: Computer Win Example 1

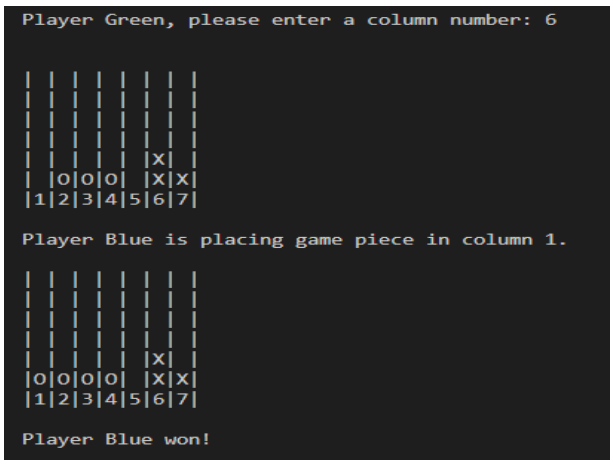

Figure 8: Computer Win Example 2

3 CONCLUSION

This project successfully applied AI searching and evaluation techniques, specifically the MinMax algorithm with alpha-beta pruning, to develop an AI agent for playing the game of Connect Four. The implementation of the MinMax algorithm played a critical role in achieving the project's objectives. By incorporating the concepts learned from the CS-470 course, the AI agent demonstrated remarkable performance. Overall, this project demonstrates the effectiveness of AI techniques, specifically the MinMax algorithm, in developing intelligent game-playing agents. The successful implementation of the algorithm and its associated components instills confidence in the generated results. The project not only showcases the adaptability and performance of the AI agent but also highlights the potential for further enhancements and exploration in the field of AI game-playing agents.