Algorithm for Tic Tac Toe Game with MCTS AI:

1. Start the Tic Tac Toe game and initialize the board, current player ('X'), AI player ('O'), and an empty cell marker (' ').

2. Create the TicTacToe class representing the game logic, including methods to check for a winner, draw, and make a move.

3. Implement the Monte Carlo Tree Search (MCTS) algorithm within the TicTacToe class:
a. The 'play_random_simulation' function simulates random playouts of the game starting from a given board state until a terminal state (win, loss, or draw) is reached.
b. The 'monte_carlo_simulation' function uses MCTS to perform random simulations for a given move on the board. It calculates the number of wins obtained from these simulations and returns the count.
c. The 'get_best_move' function performs MCTS for all available moves on the board, selecting the move with the highest win rate as the AI's best move.

4. Create the TicTacToeGUI class to build the graphical user interface using tkinter:
a. Initialize the main window and set the title.
b. Create a 3x3 grid of buttons representing the Tic Tac Toe board cells.
c. Make the first move by the AI (O) from the center position (4) using MCTS.

5. Implement the 'on_click' method to handle player moves and AI responses in the TicTacToeGUI class:
a. When a player clicks on an empty cell, update the board and check for a win, draw, or ongoing game.
b. If the game is over, show a message box with the result and end the game.
c. If not, switch the current player for the next move.
d. If it's the AI's turn, use MCTS to find the best move and update the board accordingly.
e. Check for a win, draw, or ongoing game after the AI move, and display the result if the game ends.

6. Run the game loop using `root.mainloop()` to handle player clicks and AI moves, creating an interactive experience.

Highlighting MCTS:
The use of the Monte Carlo Tree Search (MCTS) algorithm significantly improves the AI opponent's decision-making process. MCTS allows the AI to explore possible moves through random simulations and gradually refine its strategy over time. By using MCTS, the AI becomes more challenging to beat, offering a smarter and more engaging gaming experience for players.