

PROJECT REPORT

HEXAGONAL TIC-TAC-TOE

**With AI**

Submitted By:

Muhammad Unais (22K-5035)

Muhammad Muzammil (22K-5044)

Course:AI

INSTRUCTOR: Ms. Alishba Subhani

**1. Introduction**

This report outlines the design and implementation of a customized version of the classic Tic Tac Toe game. The project involves developing a Hexagonal Tic Tac Toe game on a 4x4 grid where the user competes against an AI opponent. The game logic follows traditional Tic Tac Toe rules with a winning condition of forming a line of 3 consecutive marks. The AI opponent is built using the Minimax algorithm for strategic decision-making.

**2. Objective**

* To implement a variant of the Tic Tac Toe game with a hexagonal layout.
* To integrate an AI opponent that makes optimal moves using the Minimax algorithm.
* To simulate intelligent gameplay in a 4x4 grid with modified win conditions.

**3. Game Design and Rules**

* The game board is a 4x4 grid.
* The layout is conceptually hexagonal, affecting win paths (horizontal, vertical, and diagonal alignments).
* Players take turns placing their marks (e.g., X for human and O for AI).
* The first player to form a line of 3 consecutive marks in any valid direction wins.

**4. Technologies Used**

* Programming Language: Python
* GUI Library: (Pygame)
* Algorithm: Minimax (with Alpha-Beta Pruning)

**5. Minimax Algorithm Overview**

The Minimax algorithm is used by the AI to simulate all possible game states and select the move that minimizes potential loss while maximizing its gain.

* Each game state is evaluated based on a score: +10 for AI win, -10 for human win, 0 for draw.
* The algorithm recurses through all possible moves and returns the best move.
* Depth of the search can be limited for performance optimization

**6. Implementation Details**

* The board is implemented as a 2D array.
* A function checks for a winner after each move.
* The Minimax function evaluates all possible moves by the AI.
* The human player is prompted for input through the interface.
* The game ends when a player wins or the board is full.

**7. Challenges Faced**

* Mapping hexagonal logic on a 2D grid required careful consideration of win conditions.
* Balancing performance with depth in Minimax evaluation.
* Ensuring accurate detection of 3-in-a-row in varied directions.

**8. Future Enhancements**

* Add support for dynamic board sizes.
* Implement a difficulty level (by limiting Minimax depth).
* Visual enhancements for better UI/UX.

**9. Conclusion**

The Hexagonal Tic Tac Toe game with AI integration demonstrates effective use of game theory and AI algorithms. The Minimax algorithm enables intelligent gameplay, offering users a challenging experience. This project serves as a solid foundation for further exploration in AI-driven gaming and custom board game development.

**10. References**

* Artificial Intelligence: A Modern Approach by Russell & Norvig
* Python Official Documentation
* Online resources and tutorials on Minimax and Tic Tac Toe logic
* Other Likewise projects on GitHub
* Chat-Gpt: Solving errors and bugs.