**Project Title:** **"Ludo Vortex: Multi-Dimensional Mayhem with Adaptive AI"**  
**Submitted By**

**Nabiha Fatima 21K-3634**

**ALMEEZA TARIQ 21K-3586**

**HUZAIFA GHORI 21K-3602**  
**Course:** AI  
**Instructor:** Miss Mehak  
**Submission Date:** 8/3/2025

1. **Project Overview**

Project Topic:  
An advanced version of Reversi (Othello) with an AI opponent enhanced through Minimax with Alpha-Beta Pruning and custom heuristics. The system emphasizes adaptive intelligence that analyzes the player’s strategies and adjusts move selection dynamically.

Objective:  
Develop a responsive and strategic AI capable of challenging human players using:

* Optimized Minimax decision-making (with Alpha-Beta Pruning).
* Heuristic evaluation prioritizing corners, edges, and mobility.
* An interactive Pygame-based interface for visual gameplay.

1. **Game Description**

Original Game Background:  
Reversi is a two-player turn-based strategy game on an 8×8 grid. Players take turns placing black or white discs to flip their opponent’s discs, aiming to control the majority of the board by the end of the game.

Innovations Introduced:

1. AI Optimization:
   * Use of Alpha-Beta Pruning within Minimax for faster computation.
   * Prioritization of strategic positions (e.g., corners and edges).
2. Adaptive Heuristics:
   * AI adjusts based on player move history and board conditions.
3. Difficulty Modes:
   * Configurable AI levels (Easy, Medium, Hard) to scale complexity.
4. Visual Gameplay:
   * Interactive Pygame GUI for clear game progression and feedback.
5. **AI Approach and Methodology**

AI Techniques:

* Minimax Algorithm with depth control for decision trees.
* Alpha-Beta Pruning to skip non-promising branches.
* Heuristic Evaluation Function:
  + Corner control (weight = 0.5).
  + Edge stability (weight = 0.2).
  + Disc parity and mobility (weight = 0.3).

Complexity Analysis:

* Branching Factor: Up to 10-20 legal moves per turn.
* Minimax Complexity: O(b^d), reduced to O(b^(d/2)) via pruning.
* Heuristic tuning is key to high-performance decision-making.

1. **Game Rules and Mechanics**

Modified Rules:

* Traditional rules of Reversi retained.
* AI adapts to the opponent’s strategy in real-time.
* Players can choose difficulty level before the game starts.

Winning Conditions:

* Game ends when neither player has legal moves.
* Player with the most discs on the board wins.

Turn Sequence:

1. Player makes a valid move.
2. Opponent's discs between placed disc and player’s existing discs are flipped.
3. Turn passes to opponent unless no valid moves are available.
4. **Implementation Plan**

Programming Language: Python

Libraries/Tools:

* Pygame for GUI interface.
* NumPy for board state handling.
* Scikit-learn (optional for pattern learning or reinforcement learning).

**Timeline:**

* Weeks 1–2: Finalize game rules and UI design.
* Weeks 3–4: Implement Minimax and heuristic evaluation.
* Weeks 5–6: Build full gameplay and interaction system.
* Week 7: Integrate and test AI responsiveness.
* Week 8: Polish gameplay, finalize report and prepare demo.

1. **References**
2. Russell, S., & Norvig, P. (2020). Artificial Intelligence: A Modern Approach.
3. Reversi AI Research Papers and Tutorials.
4. GitHub Repositories and Blogs on Heuristic AI in Board Games.