## Project Title:

**2D Chess with Adaptive AI Opponent**

**Submitted By:**

• Arsam Uzair (22K-5006)

• Syed Shaheer Hussaini (22K-5111)

• Muhammad Haris Sheikh (22K-5064)

Course: AI

Instructor: Ms. Ayesha Almas Ansari

Submission Date: 29-04-2025

## 1. Project Overview

**Project Topic:**

A 2D chess variant that introduces a dynamic, layered board and an AI opponent capable of strategizing in three dimensions. The game combines traditional chess rules with 2D movement mechanics (e.g., vertical piece movement across board layers) and integrates an AI using a modified Minimax algorithm optimized for 2D spatial analysis.

**Objective:**

To develop a strategic AI capable of playing 2D chess by adapting the Minimax algorithm with Alpha-Beta pruning to account for multi-layered board complexity. Secondary goals include implementing multiplayer functionality and enhancing gameplay through 2D visualization.

## 2. Game Description

**Original Game Background:**

Traditional chess involves 2D movement on an 8x8 grid, with players aiming to checkmate the opponent’s king. Pieces have fixed movement patterns (e.g., rooks move orthogonally, bishops diagonally).

**Innovations Introduced:**

• New Win Conditions: Checkmate can occur on any layer, and a "stacked checkmate" (simultaneous threats across layers) is introduced.

• Multiplayer Support: Real-time multiplayer mode with a chat feature.

**Impact on Gameplay:**

• Increased strategic depth due to 2D movement options.

• Higher computational complexity for AI decision-making.

• Visual and tactical novelty compared to traditional chess.

## 3. AI Approach and Methodology

**AI Techniques:**

• Minimax Algorithm: Modified to evaluate 2D board states.

• Alpha-Beta Pruning: Reduces the branching factor caused by 2D movement.

• Heuristic Function: Combines material value, positional control (central cubes across layers), and king safety.

**Heuristic Design:**

• Material Value: Traditional piece values (queen = 9, rook = 5, etc.) with bonuses for vertical control.

• Positional Advantage: Rewards control of central cubes and threats spanning multiple layers.

• King Safety: Penalizes exposure to cross-layer checks.

**Complexity Analysis:**

• Time Complexity: O(b^d), where branching factor b increases by ~30% due to 2D moves (e.g., 35 possible moves per turn vs. 25 in standard chess), and d represents the depth of the search tree.

• Challenges: Efficient state evaluation, pruning optimization, and real-time performance.

## 4. Game Rules and Mechanics

**Modified Rules:**

• Pawns promote upon reaching any layer’s opposite end.

• Bishops move diagonally across layers (e.g., from (1,1,1) to (3,3,3)).

• Knights follow L-shaped moves in 2D space (e.g., 2 squares horizontally + 1 vertically).

**Winning Conditions:**

• Traditional checkmate or a "stacked checkmate" (king threatened on all layers).

**Milestones and Timeline:**

• Weeks 1–2: Finalize 2D board mechanics and UI.

• Weeks 3–4: Implement 2D move validation and game state management.

• Weeks 5–6: Develop Minimax algorithm with 2D heuristics.

• Week 7: Integrate AI with frontend and optimize pruning.

## 5. References

• Russell, S., & Norvig, P. (2020). Artificial Intelligence: A Modern Approach (Minimax theory).

• Chess Programming Wiki (heuristic design).

• “2D Chess Variants” – Chess.com