**Project Title:** ***First to the Centre***
**Submitted By:** M. Hamza Khan [22k-4664]
M. Aneeq [22k-4388]
**Course:** AI
**Instructor:** Miss Alina Arshad and Almas Ayesha Ansari
**Submission Date:** 22nd March 2025

---

## 1. Project Overview

- **Project Topic:**

In this project we will create an AI opponent for 'First to the Centre,' a unique chess variant. While it keeps the core rules of chess, it introduces a new way to win, moving the king to one of the four central squares (d4, d5, e4, or e5) without being in check. To shake things up even more, rooks move like bishops, and bishops move like rooks, making for a fresh strategic challenge. The AI will be built to play against a human, adjusting its strategy to handle these new mechanics.

- **Objective:**

The goal of this project is to:
- Develop an AI opponent that can play 'First to the Centre' against a human.
- Implement Minimax with Alpha-Beta pruning for AI decision-making.
- Design a custom heuristic evaluation function to balance king movement with overall board control.
- Adapt the AI to handle the new movement rules for rooks and bishops.
- Create a user-friendly interface for Human vs. AI gameplay.

## 2. Game Description

- **Original Game Background:**

The original game is classic chess, the one we've all been familiar with since childhood. It follows the standard rules where players aim to checkmate their opponent's king using strategic moves and piece coordination.

- **Innovations Introduced:**

'First to the Centre' is a new chess variant with two main rule modifications:
**1.** A player wins immediately if they move their king to one of the four central squares (d4, d5, e4, or e5) without being in check.
**2.** Peice movement changes: Rooks move diagonally like bishops and bishops move vertically and horizontally like rooks.

## 3. AI Approach and Methodology

- **AI Techniques to be Used:**

- Minimax Algorithm – AI will use Minimax to evaluate the best move sequences.
- Alpha-Beta Pruning – To improve search efficiency and reduce computation time.
- Custom Heuristic Evaluation – Designed for 'First to the Centre' strategy.

- **Heuristic Design:**

- King's distance to the central squares (closer = better).
- King safety (avoiding checks and threats).
- Standard chess heuristics (material value, piece mobility, board control).
- Adjusted piece values due to modified movement rules for rooks and bishops

- **Complexity Analysis:**

Minimax search complexity: O(b^d), where b is the branching factor and d is the search depth. The changed piece movements introduce new strategic considerations, requiring modifications to standard chess evaluation functions.

## 4. Game Rules and Mechanics

- **Modified Rules:**

- Standard chess rules apply, except for the following changes:
  1. A player wins if they move their king to d4, d5, e4, or e5 without being in check.
  2. Rooks move diagonally like bishops.
  3. Bishops move vertically and horizontally like rooks.

- **Winning Conditions:**

- Win by checkmate (same as traditional chess).
- Win by First to the Centre condition (king reaches a central square safely).

- **Turn Sequence:**

- Human vs. AI: The human player moves first (White) by default, but an option to switch sides will be included.
- The AI will make its move immediately after the human player.

## 5. Implementation Plan

- **Programming Language:** *(Python)*

- **Libraries and Tools:**

- Python-Chess (for board representation and move generation).
- NumPy (for numerical computations).
- Pygame (for the user interface).

- **Milestones and Timeline:**

- Week 1-2: Implement game rules and board setup.
- Week 3-4: Develop the Minimax algorithm with basic heuristics.
- Week 5-6: Refine AI strategy
- Week 7: Develop a user interface and integrate AI.
- Week 8: Final testing, debugging, and documentation.

## 6. References


- Chess Programming Wiki (https://www.chessprogramming.org)
- Python-Chess Library Documentation
- AI Strategy Guides for Minimax & Alpha-Beta Pruning