



Project Title:

Sentinel Chess: An 8x8 Chess Variant with the Sentinel Pawn

Submitted by:

Maaz Kashif [22K-4518]

Abdul Rafay [22K-4192]

Umer Ahmed [22K-4213]

Course: AI-Lab

Instructor: Mr. Abdullah Yaqoob

Submission Date: 12-March-2025

1. Project Overview

Project Topic:

This project introduces a new chess variant played on an 8x8 board. It features the **Sentinel Pawn**, a unique piece with both offensive and defensive capabilities.

Objective:

- To develop an AI capable of playing this new chess variant.
- To explore the strategic impact of the Sentinel Pawn on gameplay.
- To implement and test the game using Minimax and Alpha-Beta Pruning.

2. Game Description

Original Game Background:

Based on traditional chess, with modifications to additional piece mechanics.

Innovations Introduced:

- **Sentinel Pawn** that:
 - Moves like a regular pawn.
 - Can attack in a 1-square radius in all directions.
 - Protects adjacent pieces from being captured.
 - Can be captured by any piece except a regular pawn.
 - Promotes to a **Guard Queen** upon reaching the last rank.
- Adjusted game balance to accommodate the new mechanics.

3. AI Approach and Methodology

AI Techniques to be Used:

- **Minimax Algorithm** (Modified for Sentinel Pawn evaluation.)
- **Alpha-Beta Pruning** for optimized decision-making.
- **Optional Reinforcement Learning** for self-play improvement.

Heuristic Design:

- Piece values adjusted to reflect the Sentinel Pawn's strategic importance.
- Defensive and offensive positioning evaluated dynamically.
- Additional evaluation for board control due to the Sentinel Pawn's shielding ability.

Complexity Analysis:

- Custom heuristics required for Sentinel Pawn interactions.

4. Game Rules and Mechanics

Modified Rules:

- The Sentinel Pawn follows unique movement and attack rules.
- It shields adjacent pieces from capture unless itself is taken.
- It promotes to a **Guard Queen** instead of a regular queen.

Winning Conditions:

- Standard chess checkmate rules apply.
- Stalemate and draw conditions remain unchanged.

Turn Sequence:

- Alternating turns between White and Black, as in standard chess.

5. Implementation Plan

Programming Language:

- Python

Libraries and Tools:

- **Pygame** for GUI
- **NumPy** for game state evaluation
- **Chess Engine Libraries**

6. References

- Standard chess rules and strategy resources.
- **Pygame Documentation** – <https://www.pygame.org/docs/>
- **BFS Pathfinding in Grid-based Games** – [Online AI Research Papers]
- **Python Collection Model - Deque** - <https://docs.python.org/3/library/collections.html>