

# AI PROJECT REPORT

**Project Title:** Adaptive AI Game for Nine Men's Morris

**Submitted By:** Aisha Jalil (22k-4649), Aniqaz Azhar (22k-4228), Laiba Mohsin (22k-4246)

**Course:** AI

**Instructor:** Alishba Subhani

**Submission Date:** 11th May, 2025

## 1. Executive Summary

### Project Overview:

This project aimed to implement an adaptive AI opponent for the classical strategy game Nine Men's Morris using the Minimax algorithm with Alpha-Beta Pruning. The game adheres to its original three-phase rules (placement, movement, and flying) and challenges the AI to make strategic decisions based on the game's dynamic states. No core game rules were altered, but innovative heuristics were designed to evaluate each state effectively across all three phases of gameplay.

## 2. Introduction

### Background:

Nine Men's Morris is a two-player abstract strategy board game dating back to the Roman Empire. It involves forming "mills" (three aligned pieces) to capture the opponent's pieces. Due to its deep strategic complexity, including distinct game phases and the "flying" mechanic when reduced to three pieces, it presents an ideal case for implementing and testing classical AI search techniques.

### Objectives of the Project:

- Develop an AI opponent using the Minimax algorithm with Alpha-Beta Pruning.
- Design heuristic functions suited for all three phases of the game.
- Integrate the AI into a fully functioning Python-based GUI using Pygame.
- Evaluate the AI's effectiveness in competitive play.

## 3. Game Description

### Original Game Rules:

- Each player begins with nine pieces.
- The game consists of three phases:
  1. **Placing Phase:** Players alternate placing pieces. Mills allow for capturing opponent's pieces.

2. **Moving Phase:** Players move pieces to adjacent points.
  3. **Flying Phase:** If a player has only three pieces, they may move to any open point.
- A player wins by reducing the opponent to two pieces or leaving them with no legal moves.

#### **Innovations and Modifications:**

- No changes were made to the fundamental rules.
- Innovations lie in AI design:
  - Adaptive Heuristic Functions based on game phase.
  - Evaluation metrics include potential mills, moveable pieces of opponent, and potential for opponent forming mills.
  - Emphasis on strategic depth rather than rule changes.

## **4. AI Approach and Methodology**

#### **AI Techniques Used:**

- **Minimax Algorithm:** Simulates future game states to choose the best strategic move.
- **Alpha-Beta Pruning:** Optimizes the Minimax tree by discarding suboptimal branches to improve efficiency.

#### **Algorithm and Heuristic Design:**

Heuristic evaluation included:

- Number of mills formed.
- Number of legal moves of opponent pieces.
- Count of possibilities of mill formations by opponent at current board state
- Difference between number of pieces on board

#### **AI Performance Evaluation:**

- AI's decisions were timed and tested in simulated games.
- Decision time was consistently within 8 to 12 seconds per move in a game.
- Win rates for AI were more
- The AI accurately followed the heuristics and rules, and tried to block mill formation by opponent while also creating its own mills

## 5. Game Mechanics and Rules

### Modified Game Rules:

- No changes were made to the original game rules.
- The AI adheres strictly to traditional mechanics.

### Turn-based Mechanics:

- Players alternate turns in each phase.
- During the flying phase, players with three pieces can move to any unoccupied point.
- After each mill formation, a piece is removed from the opponent, if eligible.

### Winning Conditions:

- Opponent is reduced to two pieces.
- Opponent has no legal moves remaining.

## 6. Implementation and Development

### Development Process:

- Game rules and board were implemented first.
- Minimax algorithm with alpha beta pruning designed and integrated.
- Heuristics tuned based on testing across different phases.
- GUI created using Pygame for visual interaction.

### Programming Languages and Tools:

- Programming Language: Python
- Libraries: Pygame (GUI), copy, time
- Tools: GitHub (version control), Visual Studio Code

### Challenges Encountered:

- Designing flexible heuristics that adapt across three different game phases. By understanding board patterns, and balancing key game features like mill formation, mobility, and threat prevention, we were able to come up with suitable heuristics

- Balancing decision time and AI depth to avoid lag in gameplay, which was addressed by choosing a feasible depth which was 3.

## 7. Team Contributions

- **Aisha Jalil (22k-4649):** Heuristic function, Minimax implementation, Alpha-Beta pruning optimization.
- **Aniqa Azhar (22k-4228):** Game rules implementation, board state management, phase transition handling.
- **Laiba Mohsin (22k-4246):** GUI design using Pygame, integration with gameflow, and final debugging.

## 8. Results and Discussion

### AI Performance:

- Sample average decision time with alpha beta pruning at depth 4 for one game

```
AI Performance Report
=====
Average Decision Time: 1.96 seconds
```

- Sample average decision time with alpha beta pruning at depth 3 for one game

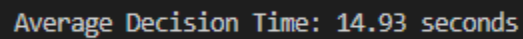
```
=====
Human vs AI
Game Winner: AI

AI Performance Report
=====
Average Decision Time: 16.04 seconds
```

- Sample average decision time with minimax at depth 3 for one game

```
=====
Average Decision Time: 4.80 seconds
```

Sample average decision time with minimax at depth 4 for one game:



Average Decision Time: 14.93 seconds

- Effectiveness increased with deeper lookahead, though computational cost rose significantly without pruning.
- The AI demonstrated strong tactical awareness, especially in mill formation and blocking.

## 9. References

- [Wikipedia - Nine Men's Morris](#)
- Alpha Beta Pruning: <https://www.geeksforgeeks.org/alpha-beta-pruning-in-adversarial-search-algorithms/>
- Pygame Documentation: <https://www.pygame.org/docs/>