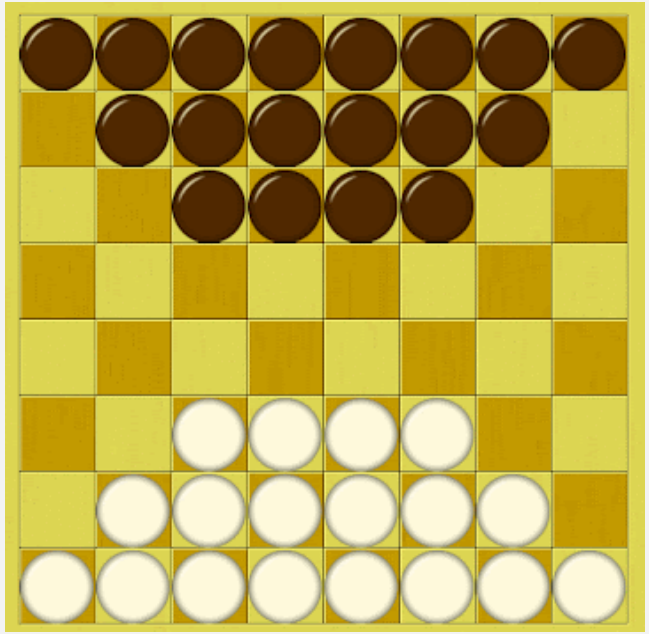
DAMEO: AI

SEARCH-BASED AGENT

EIACD ASSIGNMENT

U.PORTO
FACULDADE DE CIÊNCIAS
UNIVERSIDADE DO PORTO

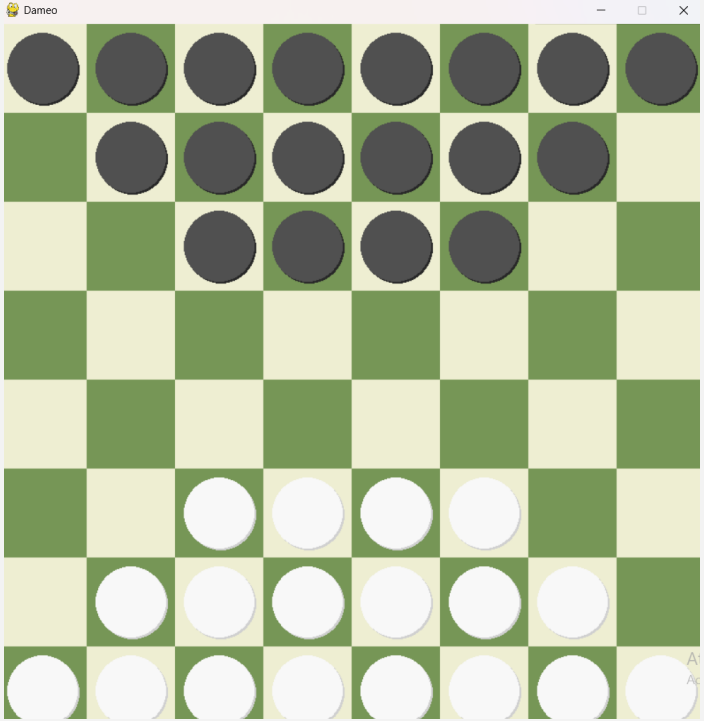
U.PORTO
FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO



Henrique Pilão (up202405162)|Paulo Pereira (up202406285) | EIACD - PL3 | FCUP & FEUP

Introduction to Dameo

- **Overview:** 2-player strategy board game (8x8), implemented in Python/Pygame.
- **Key Rules:** Distinct Pawn/King movement, Promotion, Mandatory Maximum Capture.
- **Goal:** Capture all opponent pieces or block all their moves.



Problem Formulation (Adversarial Search)

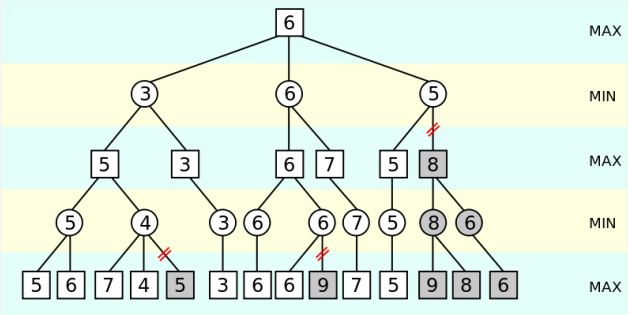
- Approach: Minimax algorithm for decision-making.
- State: 8x8 Board matrix + Current Player.
- Actions: Valid moves/captures (represented as paths).
- Evaluation: State value determined by a heuristic function.

Implementation Overview

- **Core Tech:** Python & Pygame library.
- **Structure:** Modular design (main, menu, game_loop, utils (core logic), ia_dameo (AI), dameo_pygame (drawing)).

Implemented AI Algorithms

Main AI: Minimax with Alpha-Beta Pruning (Medium/Hard difficulties).



Simple AI: "Easy" difficulty uses basic evaluation (depth 1) / random choice, prioritizes promotions.

Implementation Details: Heuristics & Operators

Heuristic Function:

(evaluate_board): Considers piece count/value, advancement, center control, mobility.

Game Logic (utils):

Handles mandatory/max/multi-captures, King vs. Pawn rules.

Tie-breaking:

Random choice among moves with equal Minimax scores.

Experimental Results

(Results are Win % / Draw % / Loss % from the perspective of the Row AI)

| vs Easy (Black) | vs Medium (Black) | vs Hard (Black) | | :--- | :--- | :--- |

| Easy (White) | 45% / 10% / 45% | 5% / 5% / 90% | 1% / 2% / 97% |

| Medium (White)| 92% / 5% / 3% | 48% / 15% / 37% | 15% / 10% / 75% |

| Hard (White) | 98% / 2% / 0% | 78% / 12% / 10% | 46% / 20% / 34% |

Interface

Type:
Graphical User Interface (GUI) using Pygame.

Features:

Menus (Mode, Difficulty, Rules), Board/Piece Rendering, Visual Feedback (Selection, Possible Moves), Mouse Interaction.

Conclusions & Learnings

Summary: Successfully implemented Dameo game with functional AI.

Challenges: Complex capture logic, heuristic tuning, GUI integration.

Learnings: Adversarial search (Minimax), heuristic design, Pygame development.

Bibliografia:

GitHub Repository - Dameo Game: GitHub - stifler9/Dameo: Dameo game (similar to checkers)

Russell, S., & Norvig, P. (2021). Artificial Intelligence: A Modern Approach (4th ed.). Pearson.

Google Gemini. (2025). Assistive AI for programming and game design./https://gemini.google.

OpenAI ChatGPT. (2025). Assistive AI for programming and game development. https://chat.openai.com.