ISEL
INSTITUTO SUPERIOR DE
ENGENHARIA DE LISBOA

Departamento de Engenharia de Eletrónica e Telecomunicações e de Computadores
Licenciatura em Engenharia Informática e de Computadores
Licenciatura em Matemática Aplicada à Tecnologia e à Empresa
Licenciatura em Engenharia Informática, Redes e Telecomunicações

First Practical Project

Artificial Intelligence course

2024/2025 Summer Semester

Version 1.00

Teacher: Nuno Leite

ISEL, 9 March 2025

ISEL
INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA

Artificial Intelligence course
First Practical Project
2024/2025 Summer Semester

Learning objectives

At the end of the **first practical project**, students should be able to:

- □ Understand and master the syntax of Prolog programs;
- □ Understand and master Prolog basic programming constructs such as atoms, numbers, compound terms, lists, operators and arithmetic;
- □ Understand and control backtracking;
- □ Know how to use typical built-in predicates;
- □ Understand how to implement a Two-person, perfect-information game;
- □ Understand the minimax principle;
- □ Understand the alpha-beta algorithm: an efficient implementation of minimax.

The Checkers game

In this project, your group will design and program the famous *Checkers* game. Checkers is a two player abstract strategy board game which involve forward movements of uniform game pieces and mandatory captures by jumping over opponent pieces. The rules of the game can be consulted in: `https://en.wikipedia.org/wiki/Checkers`, `https://www.britannica.com/topic/checkers`.

Your assignment is to develop a Prolog program that allow two human players to play, using the standard input to enter game commands. The Checkers board and other needed info is displayed in the standard output. Hence, the input/output is text-based.

An example of the interaction is the following:

```
Welcome to the Checkers Prolog game!

  A B C D E
5 . ○ . ○ . 5
4 ○ . ○ . ○ 4
3 . . . . . 3
2 . ● . ● . 2
1 ● . ● . ● 1
  A B C D E

Player 1 (row/column): 2b 3c
...
```

In addition to allow playing by humans, the program should have an operation mode where a human player plays against the computer. The computer player (AI player) is programmed using the alpha-beta algorithm, an efficient implementation of the minimax principle.

Notes:

- The board is modeled using Prolog lists.
- For reading and writing in Prolog, use the built-in predicates, `read(N)` and `write(N)`. The following query writes the black and white circles (Unicode symbols):
  `?- write('\u25cf'), write('\u25cb').`
- You should try the game first with small boards, e.g., 2 × 2 (just to test some movements), …, 3 × 3, and then bigger boards. The Prolog code could be generic in order to facilitate the different layout generation.
- When testing the minimax algorithm, begin with small boards, e.g., 3 × 3. For bigger boards, minimax will be too slow, and you have to run the alpha-beta algorithm with and without limited depth. You can use a limited depth value of 3, for example.

Due date: 12 April 2025 until 23:59.

The delivery of the work must present the report and Prolog code developed, delivered in the Moodle system. The report must be concise, explain the main predicates/structure, and justify all decisions taken. It must indicate the student group composition and the curricular unit info.