

# Manual de Utilizador

---

Inteligência Artificial - Projeto 2

Projeto realizado por: Francisco Zacarias nº190221105

## Table of Contents

---

- [Manual de Utilizador](#)
- [Table of Contents](#)
- [0. Glossário](#)
- [1. Introdução](#)
- [2. Instalação e Utilização](#)
  - [2.1 Instalação](#)
  - [2.2 Utilização](#)
- [3. Escrita de ficheiros](#)
- [4. Exemplo de Aplicação](#)
  - [4.1 Humano contra Computador](#)
    - [4.1.1 Jogada do humano](#)
- [5. Limitações a nível do utilizador](#)

## 0. Glossário

---

*REPL - Read-Eval-Print Loop*

## 1. Introdução

---

Este manual irá guiar o utilizador através dos processos de instalação do ambiente do projeto, a instalação dos ficheiros, navegação geral da aplicação e o conteúdo dos ficheiros relevantes ao utilizador. Este manual apenas assume que o utilizador está familiarizado com as regras do jogo do quatro.

## 2. Instalação e Utilização

---

Este capítulo serve para o utilizador conseguir instalar e utilizar corretamente este jogo.

### 2.1 Instalação

Para instalar o jogo, é necessário extrair a pasta **Lisp** do diretório do projeto, para uma partição do computador onde o LispWorks tenha permissões de leitura e escrita. À partida, qualquer partição que não contenha o sistema operativo será suficiente. De seguida é necessária uma alteração no ficheiro projeto.lisp. Ao abrir o mesmo, existe uma função chamada **diretorio-projeto**, que contém o caminho para a pasta do projeto.

```
(defun diretorio-projeto ()
  "Devolve o diretorio do projeto"
  "D:\\IA\\FranciscoZacarias_190221105_P2\\Lisp\\"
)
```

O diretório destacado na imagem deve ser alterada para ajustar ao ambiente do utilizador. Notar que, em Windows, são utilizadas duas "\", pois isto representa um caracter especial do lisp. Ao utilizar dois, estamos na verdade apenas a dizer ao lisp para ler só 1, por isso é fundamental que o caminho do diretório do projeto seja separado por "\\" e inclusivamente deixar mais duas no fim do caminho.

## 2.2 Utilização

Para a utilização do projeto, é recomendado a utilização do *LispWorks*, pois foi o ambiente no qual o projeto foi testado.

Para compilar o projeto, é apenas preciso compilar o ficheiro *interact.lisp*, e esse ficheiro tratará de compilar os restantes. Para iniciar o jogo, o utilizador deve chamar, no *REPL*, a função *iniciar* i.e. (iniciar).

## 3. Escrita de ficheiros

---

Este programa vai escrever no ficheiro *logs.dat* as estatísticas das jogadas. Tanto para o jogador (tais como peça jogada, tabuleiro e posição) como para o Algoritmo.

No ficheiro há uma divisão entre os jogos para aumentar a legibilidade. Um jogo fica representado desta forma:

```
===== NOVO JOGO =====

Jogada HUMANO || Peça: (BRANCA QUADRADA ALTA OCA) || Posicao: (x= 1,y=0) || Tabuleiro: (((BRAN
Jogada PC2 || Nós analisados:122 || Cortes MAX: 0 || Cortes MIN: 0 || Tempo Gasto: 16ms || Util
Jogada HUMANO || Peça: (PRETA QUADRADA ALTA OCA) || Posicao: (x= 1,y=1) || Tabuleiro: (((BRANC
Jogada PC2 || Nós analisados:82 || Cortes MAX: 0 || Cortes MIN: 0 || Tempo Gasto: 0ms || Utilid
===== FIM DO JOGO =====
```

Para as jogadas do computador são guardadas as seguintes estatísticas:

- Nós analisados: Quantidade de nós que o algoritmo gerou
- Cortes MAX: Quantidade de cortes feitos para o MAX
- Cortes MIN: Quantidade de cortes feitos para o MIN
- Tempo gasto: Tempo que o algoritmo demorou a encontrar a melhor jogada, em milissegundos.
- Utilidade: Utilidade do nó escolhido
- Tabuleiro: Tabuleiro resultante do operador aplicado

## 4. Exemplo de Aplicação

---

Este jogo pode ser jogado de duas formas. Existe a possibilidade de um utilizador jogar contra um computador, ou de fazer o computador jogar contra ele próprio.

## 4.1 Humano contra Computador

Este modo de jogo é um modo de jogo interativo em que o utilizador vai jogar contra o computador. Ao iniciar o programa, deve ser escolhido a opção "Humano vs PC", quem joga primeiro, o tempo limite para o computador fazer uma jogada e a profundidade de procura do algoritmo.

### 4.1.1 Jogada do humano

Quando é a vez do jogador, é apresentado o tabuleiro e a reserva no ecrã.

```

Modo de jogo: Humano vs PC
Primeiro a jogar: Humano
Tempo limite do PC para jogar: 5000ms
Profundidade limite do algoritmo: 2
  0 1 2 3
0. ((BRANCA REDONDA ALTA OCA) 0 0 0)
1. ((PRETA QUADRADA ALTA CHEIA) 0 0 0)
2. ((PRETA REDONDA BAIXA OCA) 0 0 0)
3. ((BRANCA QUADRADA BAIXA CHEIA) 0 0 0)

0. (PRETA REDONDA ALTA OCA)
1. (BRANCA QUADRADA ALTA OCA)
2. (PRETA QUADRADA ALTA OCA)
3. (BRANCA QUADRADA BAIXA OCA)
4. (PRETA QUADRADA BAIXA OCA)
5. (BRANCA REDONDA ALTA CHEIA)
6. (PRETA REDONDA ALTA CHEIA)
7. (BRANCA REDONDA BAIXA CHEIA)
8. (PRETA REDONDA BAIXA CHEIA)
9. (BRANCA QUADRADA ALTA CHEIA)
10. (BRANCA QUADRADA BAIXA CHEIA)
11. (PRETA QUADRADA BAIXA CHEIA)
Escolha uma peca do tabuleiro (COORDENADA X):
Input>

```

Para o utilizador jogar, terá que introduzir as coordenadas x e y, respetivamente da posição em que quer jogar.

As coordenadas do tabuleiro são da seguinte forma:

0,0	0,1	0,2	0,3
1,0	1,1	1,2	1,3
2,0	2,1	2,2	2,3
3,0	3,1	3,2	3,3

Isto significa que, a título de exemplo, se o jogador quiser jogar na primeira linha, coluna 2 iria introduzir 1 para o x, e 0 para o y.

Por fim, para escolher a peça que pretende jogar, deverá escolher o numero respetivo da peça, apresentado na reserva. Digamos que neste caso o utilizador queria jogar a peca (BRANCA QUADRADA ALTA OCA), na segunda posição, iria introduzir no jogo o numero 1.

```
Escolha uma peca do tabuleiro (COORDENADA X):
Input>1
Escolha uma peca do tabuleiro (COORDENADA Y):
Input>0
Escolha uma peca da reserva (numero):
Input>1
Jogada HUMANO || Peca: (BRANCA QUADRADA ALTA OCA) || Posicao: (x= 1,y=0) || Tabuleiro: (((BRANCA REDONDA ALTA OCA) (BRANCA QUADRADA ALTA OCA) 0 0) ((PRETA QUADRADA ALTA CHEIA) 0 0 0) ((PRETA REDONDA BAIXA OCA) 0 0 0) ((BRANCA QUADRADA BAIXA CHEIA) 0 0 0))
Jogada PC2 || NÃ's analisados:387 || Cortes MAX: 0 || Cortes MIN: 120 || Tempo Gasto: 16ms || Utilidade:1000 || Tabuleiro:(((BRANCA REDONDA ALTA OCA) (BRANCA QUADRADA ALTA OCA) (PRETA REDONDA ALTA OCA) 0) ((PRETA QUADRADA ALTA CHEIA) 0 0 0) ((PRETA REDONDA BAIXA OCA) 0 0 0) ((BRANCA QUADRADA BAIXA CHEIA) 0 0 0))
```

Aqui vemos qual foi a jogada do utilizador, que é impressa logo depois de escolher a peça. Depois vê a jogada do computador, com as estatísticas incluídas.

Após a jogada do computador, volta a apresentar o tabuleiro e é a vez do utilizador jogar. O jogo acaba quando há um empate, ou alguém venceu, como é apresentado na imagem seguinte.

```
0 1 2 3
0. ((BRANCA REDONDA ALTA OCA) (BRANCA QUADRADA ALTA OCA) (PRETA REDONDA ALTA OCA) 0)
1. ((PRETA QUADRADA ALTA CHEIA) 0 0 0)
2. ((PRETA REDONDA BAIXA OCA) 0 0 0)
3. ((BRANCA QUADRADA BAIXA CHEIA) 0 0 0)

0. (PRETA QUADRADA ALTA OCA)
1. (BRANCA QUADRADA BAIXA OCA)
2. (PRETA QUADRADA BAIXA OCA)
3. (BRANCA REDONDA ALTA CHEIA)
4. (PRETA REDONDA ALTA CHEIA)
5. (BRANCA REDONDA BAIXA CHEIA)
6. (PRETA REDONDA BAIXA CHEIA)
7. (BRANCA QUADRADA ALTA CHEIA)
8. (BRANCA QUADRADA BAIXA CHEIA)
9. (PRETA QUADRADA BAIXA CHEIA)
Escolha uma peca do tabuleiro (COORDENADA X):
Input>3
Escolha uma peca do tabuleiro (COORDENADA Y):
Input>0
Escolha uma peca da reserva (numero):
Input>0
Jogada HUMANO || Peca: (PRETA QUADRADA ALTA OCA) || Posicao: (x= 3,y=0) || Tabuleiro: (((BRANCA REDONDA ALTA OCA) (BRANCA QUADRADA ALTA OCA) (PRETA REDONDA ALTA OCA) (PRETA QUADRADA ALTA OCA)) ((PRETA QUADRADA ALTA CHEIA) 0 0 0) ((PRETA REDONDA BAIXA OCA) 0 0 0) ((BRANCA QUADRADA BAIXA CHEIA) 0 0 0))
HUMANO venceu!
```

## 5. Limitações a nível do utilizador

Todos os utilizadores da aplicação deverão estar conscientes das seguintes limitações do projeto:

- Não são feitas verificações aos inputs. Isto é, qualquer input do utilizador não é validado e será corrido pelo programa, o que poderá despoletar um erro no projeto. É, no entanto, apresenta a lista de opções válidas, sempre que é pedido input do utilizador.
- Devido a limitações de performance, é possível que o computador não escolha uma jogada dentro do tempo limite se a profundidade for muito elevada. Em certas situações poderá mesmo acabar com a memória disponível do *LispWorks* e será necessário reiniciar o mesmo.