

Manual de Utilizador

Inteligência Artificial - Escola Superior de Tecnologia de Setúbal
2020/2021

André Nascimento nº160221075

Eduardo Ferreira nº110221031

Acrónimos e Convenções usadas

Nega-memo - Memoização que vai chamar o algoritmo negamax e vai guardar o resultado na hash table.

Introdução

Este manual servirá de guia para a execução e resolução do problema do Quatro, proposto ambita da unidade curricular de Inteligencia Artificial (IA).

Este é constituído por um tabuleiro de 4x4 casas, juntamente com um conjunto de peças, podendo ter até 16 peças dependendo do problema, estas podem ter algumas características na forma de formato (cheia ou oca) e cor (brancas ou pretas).

O problema proposto tem como objetivo, a aplicação de algoritmos relacionados com a teoria de jogos, lecionada nas aulas teoricas da unidade curricular de Inteligencia Artificial(IA).

Instalação e utilização

1. O utilizador começa por abrir o LispWorks, de seguida vai abrir no *editor* o ficheiro *interact.lisp* e depois clica no botão *compile buffer* que vai compilar todas as funções necessárias para correr o projeto.
2. voltar para o *listener* e executa o comando *_* (iniciar).
3. Após ter sido iniciado, o programa interage com o utilizador, perguntando-lhe se é humano ou computador, o utilizador terá de escolher entre uma dessas opções. Caso o utilizador escolher "humano", este poderá jogar contra o computador, no caso de escolher "Computador" então o computador irá jogar contra si mesmo.
4. De seguida será depois também perguntado se o utilizador deseja jogar em primeiro lugar ou em segundo lugar.
5. Após ter sido escolhida a ordem de jogadores, ser-lhe-á pedido o tempo de processamento que o algoritmo deverá ter.
6. Após ter sido escolhido o tempo limite, também será pedido qual a profundidade que deverá avaliar o algoritmo.
7. Após ter a profundidade ter sido escolhida, o jogo começa.

Input/Output

Quando o programa é executado, o utilizador primeiro escolhe se vai jogar contra o computador ou se o computador vai jogar contra outro computador. As jogadas, as peças e o tabuleiro vão ser mostrados ao longo do jogo, assim como irão ser todas guardadas no documento *log.dat*.

Após o jogo terminar, o resultado será guardado no ficheiro *log.dat*.

Exemplo de aplicação

```
CL-USER 1 > (iniciar)
Human or Computer?
Human
1st or 2nd (write 1 or 2)?
1
Time the machine has to play?
5
Desired depth?
3
```

Figura 1: Exemplo da execução do programa.

	0		0		0		0	
	0		0		0		0	
	0		0		0		0	
	0		0		0		0	

- (BRANCA QUADRADA ALTA CHEIA)
- (BRANCA QUADRADA ALTA OCA)
- (BRANCA QUADRADA BAIXA OCA)
- (BRANCA QUADRADA BAIXA CHEIA)
- (PRETA QUADRADA ALTA CHEIA)
- (PRETA QUADRADA ALTA OCA)
- (PRETA QUADRADA BAIXA OCA)
- (PRETA QUADRADA BAIXA CHEIA)
- (BRANCA REDONDA ALTA CHEIA)
- (BRANCA REDONDA ALTA OCA)
- (BRANCA REDONDA BAIXA CHEIA)
- (BRANCA REDONDA BAIXA OCA)
- (PRETA REDONDA ALTA CHEIA)
- (PRETA REDONDA ALTA OCA)
- (PRETA REDONDA BAIXA CHEIA)
- (PRETA REDONDA BAIXA OCA)

NIL

Qual a Peca que deseja jogar?

(BRANCA QUADRADA ALTA CHEIA)

Qual a Coordenada X?

0

Qual a Coordenada Y?

0

Figura 2: Exemplo de como inserir as peças.

	(BRANCA QUADRADA ALTA CHEIA)		0		(BRANCA QUADRADA ALTA OCA)		0	
0		0		0		0		
0		0		0		0		
0		0		0		0		

(BRANCA QUADRADA BAIXA OCA)

(BRANCA QUADRADA BAIXA CHEIA)

(PRETA QUADRADA ALTA CHEIA)

(PRETA QUADRADA ALTA OCA)

(PRETA QUADRADA BAIXA OCA)

(PRETA QUADRADA BAIXA CHEIA)

(BRANCA REDONDA ALTA CHEIA)

(BRANCA REDONDA ALTA OCA)

(BRANCA REDONDA BAIXA CHEIA)

(BRANCA REDONDA BAIXA OCA)

(PRETA REDONDA ALTA CHEIA)

(PRETA REDONDA ALTA OCA)

(PRETA REDONDA BAIXA CHEIA)

(PRETA REDONDA BAIXA OCA)

NIL

Qual a Peca que deseja jogar?

(PRETA REDONDA BAIXA OCA)

Qual a Coordenada X?

0

Qual a Coordenada Y?

1

Figura 3: Peças inseridas após o computador ter terminado a resposta.