

Projeto N°2: Época Normal



Inteligência Artificial 19/20

Prof. Joaquim Filipe

Eng. Filipe Mariano

Jogo do Cavalo (Knight Game)

Manual do Utilizador

Realizado por:

João Gomes - 150221001

André Gastão - 130221037

28 de Janeiro de 2020

Indice

1. Introdução
2. Instalação
3. Configuração e Settings
 - valores padrao
 - valores configuraveis
4. Interface da Aplicação
 - Menus
5. Output
 - output de consola
 - output do log-file~
6. Glossário

Exemplo de aplicação

Problema para que foi concebido

1. Introdução

Este documento é escrito recorrendo á linguagem de marcação markdown, serve como relatório do manual de utilizador do projeto Jogo do Cavalo (Knight Game).

No âmbito da unidade curricular de Inteligência Artificial, foi nos proposto o projecto número 2 do jogo (Knight Game) originalmente criado a partir do problema matemático estudado em Inteligencia Artificial Knight's tour.

Este projeto vamos considerar o jogo do cavalo na versão de dois jogadores, possibilitando um enquadramento teórico-prático, com conhecimentos adquiridos no âmbito da Teoria de Jogos, aplicando as regras propostas na secção 1 do enunciado.

Neste documento serão descritos todos os passos para que o utilizador consiga instalar e interagir com a aplicação do jogo.

2. Instalação

A aplicação necessita da instalação do IDE LispWorks.

LispWorks é uma Plataforma integrada que serve como ferramenta de desenvolvimento para Common Lisp. Poderá adquirir o PersonalEdition e fazer o seu download [aqui](#)



O sistema do Jogo do Cavalo foi implementado em linguagem LISP e foi desenvolvido com auxilio do IDE LispWorks. A estrutura do projeto é composta por 4 ficheiros:

- interact.lisp - Interação com o utilizador, escrita e leitura de ficheiros.
- jogo.lisp - Implementação da resolução do problema incluindo seletores, operadores do algoritmo Minimax com corte de Alfa-Beta e outras funções auxiliares.
- algoritmo.lisp - Implementação do algoritmo de procura de profundidade Minimax com cortes de Alfa-Beta*.
- log.dat - Funções com os problemas de .

3. Configuração

Visto que a estrutura do projeto é composta por 4 ficheiros distintos, para cada maquina temos de configurar o path ou o caminho da diretoria desses ficheiros então a alteração a realizar deverá alterar o *Path* no ficheiro interact.Lisp

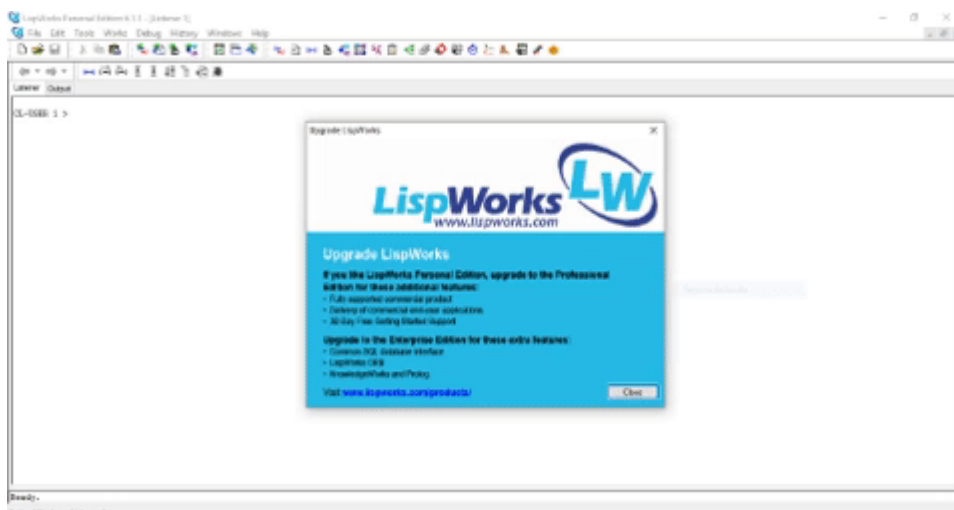
```
"permite iniciar o programa"
(defun start ()
  "Permite iniciar o programa"

  (load (compile-file (concatenate 'string (diretoria-atual) "jogo.lisp")))
  (load (compile-file (concatenate 'string (diretoria-atual) "algoritmo.lisp")))
  (configurar-tempo 5000)
  (configurar-profundidade 3)
  (menu-Load)
)

(defun diretoria-atual ()
  "Define o caminho para os ficheiros do projeto a partir da raiz"
  (let ((path
    "C:\\Users\\andre.camuenhi\\Documents\\IPS_130221037\\IA1920\\Projetos\\Projecto_2019_2020_IA_P2\\Projeto\\"))
    path
  )
)

(defun ficheiro-solucao ()
  (let ((path
    "C:\\Users\\andre.camuenhi\\Documents\\IPS_130221037\\IA1920\\Projetos\\Projecto_2019_2020_IA_P2\\Projeto\\log.dat"))
    path
  )
)
```

Após a configuração do path temos então de abrir o Lisp Works, de seguida abrir o ficheiro interact.Lisp e Compilar o mesmo.



4. Interção com Sistema

Com a instalação e a configuração dos ficheiros Podemos assim correr a aplicação:

- ☒ Instalação
- ☒ Configuração
- ☐ Executar

Devido a consola não ser *Responsive* aconselha-se á maximização da janela do listener para ter uma melhor experiência de vizualização

4.1 Menus

- Home o menu principal mostra as opções do jogo, para qual o utilizador pode escolher uma das três opções para jogar e a opção quatro serve para sair do programa

```

$                JOGO DO CAVALO                $
$                •(Knight Game) •                $
$                                                    $
$                1-Play                          $
$                2-Game Rules                    $
$                3-Settings                      $
$                4-Quit                         $
$                                                    $
$_____ $
Option -> 1

```

4.2 Games Modes

A opção Play permite mostrar o menu do jogo com as opções entre Humano com Humano, Humano com Computador e Computador contra Computador, em que o utilizador deve escolher a opção 1 e carregar na tecla enter.

O novo menu mostra 3 opções a escolha por tipo de jogos, na qual o utilizador pode escolher uma das três opções e sendo o zero (0) opção permite a saída do programa principal.

- Humana vs Human Ao escolher na opção 1, o sistema permite jogar Humano contra Humano e acumular pontos

```

$_____ $
$                GAME MODES                      $
$                                                    $
$                                                    $

```

```

$          1- Human vs Human          $
$          2- Human vs CPU             $
$          3- CPU vs CPU               $
$          0- Home Menu                $
$                                     $
$                                     $
$_____ $

```

Option -> 1

A Configurar

HUMAN VS HUMAN

```

021 012 042 049 057 095 060 016 023 043
086 089 054 031 015 001 040 085 080 033
098 047 030 083 070 020 068 027 045 092
029 003 084 056 071 081 009 002 014 079
059 078 053 044 074 036 006 034 072 097
062 013 019 088 010 011 063 050 064 041
066 061 067 058 032 017 039 000 051 037
028 007 046 005 065 024 082 093 018 038
091 076 077 048 035 096 052 008 090 026
087 004 022 075 069 055 094 073 025 099

```

KNIGHT GAME

MATCH STARTED

```

021 012 042 049 057 0-1 060 016 023 043
086 089 054 031 015 001 040 085 080 033
098 047 030 083 070 020 068 027 045 092
029 003 084 056 071 081 009 002 014 079
NIL 078 053 044 074 036 006 034 072 097
062 013 019 088 010 011 063 050 064 041
066 061 067 058 032 017 039 000 051 037
028 007 046 005 065 024 082 093 018 038

```

091	076	077	048	035	096	052	008	090	026
087	004	022	075	069	055	094	073	025	0-2

ROUND I

Points Player 1 (White): 00095

Points Player 2 (Black): 00099

Expanded Nodes: 0000000000

Analyzed Nodes: 0000000000

Cut Nodes: 0000000000

Player (WHITE)

021	012	042	049	057	0-1	060	016	023	043
086	089	054	031	015	001	040	085	080	033
098	047	030	083	070	020	068	027	045	092
029	003	084	056	071	081	009	002	014	079
NIL	078	053	044	074	036	006	034	072	097
062	013	019	088	010	011	063	050	064	041
066	061	067	058	032	017	039	000	051	037
028	007	046	005	065	024	082	093	018	038
091	076	077	048	035	096	052	008	090	026
087	004	022	075	069	055	094	073	025	0-2

Jogador (WHITE)

- Por favor escolha um operador de 1 a 8

->

021	012	042	049	057	0-1	060	016	023	043
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

```

086 089 054 031 015 001 040 085 080 033
098 047 030 083 070 020 068 027 045 092
029 003 084 056 071 081 009 002 014 079
NIL 078 053 044 074 036 006 034 072 097
062 013 019 088 010 011 063 050 064 041
066 061 067 058 032 017 039 000 051 037
028 007 046 005 065 024 082 093 018 038
091 076 077 048 035 096 052 008 090 026
087 004 022 075 069 055 094 073 025 0-2

```

Jogador (WHITE)

- Por favor escolha um operador de 1 a 8

->1

ROUND II

Points Player 1 (White): 00163

Points Player 2 (Black): 00099

Expanded Nodes: 0000000000

Analyzed Nodes: 0000000000

Cut Nodes: 0000000000

Player (BLACK)

```

021 012 042 049 057 NIL 060 016 023 043
NIL 089 054 031 015 001 040 085 080 033
098 047 030 083 070 020 0-1 027 045 092
029 003 084 056 071 081 009 002 014 079

```

```

NIL  078  053  044  074  036  006  034  072  097
062  013  019  088  010  011  063  050  064  041
066  061  067  058  032  017  039  000  051  037
028  007  046  005  065  024  082  093  018  038
091  076  077  048  035  096  052  008  090  026
087  004  022  075  069  055  094  073  025  0-2

```

Jogador (BLACK)

- Por favor escolha um operador de 1 a 8

->

Human vs CPU

A opção Human vs CPU permite o utilizador escolher o jogo entre homem contra CPU ou seja o jogador um contra a maquina, em que CPU irá aguardar pela a jogada do utilizador, sendo que que a vez de CPU jogar e vence quem acumula a maior pontuação. exemplo:

Option -> 2

A Configurar

HUMAN VS CPU

```

070  039  017  002  009  024  080  093  014  063
016  049  046  028  073  074  050  021  053  015
078  022  096  052  062  020  077  030  056  075
069  026  086  076  065  071  007  083  032  059
033  089  087  019  043  029  005  058  011  084
055  041  066  099  094  064  097  008  045  067
061  092  038  025  036  000  057  027  095  031
010  060  042  085  012  047  003  044  040  088
098  004  048  023  082  001  068  081  018  034
079  051  091  013  006  072  037  054  035  090

```


KNIGHT GAME									
MATCH STARTED									
070	NIL	017	002	009	024	080	0-1	014	063
016	049	046	028	073	074	050	021	053	015
078	022	096	052	062	020	077	030	056	075
069	026	086	076	065	071	007	083	032	059
033	089	087	NIL	043	029	005	058	011	084
055	041	066	099	094	064	097	008	045	067
061	092	038	025	036	000	057	027	095	031
010	060	042	085	012	047	003	044	040	088
098	004	048	023	082	001	068	081	018	034
079	051	0-2	013	006	072	037	054	035	090

ROUND I

Points Player 1 (White): 00093

Points Player 2 (Black): 00091

Expanded Nodes: 0000000000

Analyzed Nodes: 0000000000

Cut Nodes: 0000000000

Player (WHITE)

070	NIL	017	002	009	024	080	0-1	014	063
016	049	046	028	073	074	050	021	053	015
078	022	096	052	062	020	077	030	056	075

069	026	086	076	065	071	007	083	032	059
033	089	087	NIL	043	029	005	058	011	084
055	041	066	099	094	064	097	008	045	067
061	092	038	025	036	000	057	027	095	031
010	060	042	085	012	047	003	044	040	088
098	004	048	023	082	001	068	081	018	034
079	051	0-2	013	006	072	037	054	035	090

Jogador (WHITE)

- Por favor escolha um operador de 1 a 8

->7

ROUND II

Points Player 1 (White): 00167

Points Player 2 (Black): 00189

Expanded Nodes: 0000057167

Analyzed Nodes: 0000026353

Cut Nodes: 0000018728

Player (BLACK)

070	NIL	017	002	009	024	080	NIL	014	063
016	049	046	028	073	0-1	050	021	053	015
078	022	096	052	062	020	077	030	056	075
069	026	086	076	065	071	007	083	032	059
033	NIL	087	NIL	043	029	005	058	011	084
055	041	066	099	094	064	097	008	045	067
061	092	038	025	036	000	057	027	095	031

010	060	042	085	012	NIL	003	044	040	088
0-2	004	048	023	082	001	068	081	018	034
079	051	NIL	013	006	072	037	054	035	090

ROUND III

Points Player 1 (White): 00167

Points Player 2 (Black): 00189

Expanded Nodes: 0000000000

Analyzed Nodes: 0000000000

Cut Nodes: 0000000000

Player (WHITE)

070	NIL	017	002	009	024	080	NIL	014	063
016	049	046	028	073	0-1	050	021	053	015
078	022	096	052	062	020	077	030	056	075
069	026	086	076	065	071	007	083	032	059
033	NIL	087	NIL	043	029	005	058	011	084
055	041	066	099	094	064	097	008	045	067
061	092	038	025	036	000	057	027	095	031
010	060	042	085	012	NIL	003	044	040	088
0-2	004	048	023	082	001	068	081	018	034
079	051	NIL	013	006	072	037	054	035	090

Jogador (WHITE)

- Por favor escolha um operador de 1 a 8
->8

ROUND IV

Points Player 1 (White): 00232

Points Player 2 (Black): 00281

Expanded Nodes: 0000062213

Analyzed Nodes: 0000031193

Cut Nodes: 0000019826

Player (BLACK)

070	NIL	017	002	009	024	080	NIL	014	063
016	049	046	028	073	NIL	050	021	053	015
078	022	096	052	062	020	077	030	NIL	075
069	026	086	076	0-1	071	007	083	032	059
033	NIL	087	NIL	043	NIL	005	058	011	084
055	041	066	099	094	064	097	008	045	067
061	0-2	038	025	036	000	057	027	095	031
010	060	042	085	012	NIL	003	044	040	088
NIL	004	048	023	082	001	068	081	018	034
079	051	NIL	013	006	072	037	054	035	090

ROUND V

Points Player 1 (White): 00232

Points Player 2 (Black): 00281

Expanded Nodes: 0000000000

Analyzed Nodes: 0000000000

Cut Nodes: 0000000000

Player (WHITE)

070	NIL	017	002	009	024	080	NIL	014	063
016	049	046	028	073	NIL	050	021	053	015
078	022	096	052	062	020	077	030	NIL	075
069	026	086	076	0-1	071	007	083	032	059
033	NIL	087	NIL	043	NIL	005	058	011	084
055	041	066	099	094	064	097	008	045	067
061	0-2	038	025	036	000	057	027	095	031
010	060	042	085	012	NIL	003	044	040	088
NIL	004	048	023	082	001	068	081	018	034
079	051	NIL	013	006	072	037	054	035	090

Jogador (WHITE)

- Por favor escolha um operador de 1 a 8

->

Human vs Human

A opção Human vs Human permite o utilizador escolher o jogo entre humano contra humano , em que O seu funcionamento consiste em realizar uma jogada e trocar de jogador sucessivamente.

4.3 Game Rules

Neste Menu o utilizador pode ver as regras, os objetivos e as jogadas possíveis

_____GAME
Rules_____

(Knight Game)

- 1- Esta versão do jogo consiste num tabuleiro com 10 linhas e 10 colunas (10X10)
- 2- Em que cada casa possui uma pontuação com valor entre 00 e 99 (Aleatório), sem repetição nas células do tabuleiro.
- 3- O objectivo do jogo é acumular mais pontos que o adversário, usando um

cavalo de xadrez.

Cada jogador tem um cavalo da sua cor (branco ou preto).

4- Todas as jogadas seguintes são efectuadas através de um movimento de cavalo

(usando as regras tradicionais do Xadrez para o cavalo).

Um cavalo não pode saltar para uma casa vazia (sem número)

e também não pode fazê-lo para uma casa que esteja ameaçada pelo cavalo adversário.

5- O jogo termina quando não for possível movimentar qualquer um dos cavalos no tabuleiro,

sendo o vencedor o jogador que ganhou mais pontos.

Option ->

4.4 Setting

Este menu permite ao utilizador escolher as opções de 1 a 2 para as definições do jogo. Se a escolha for 0 é dado um jogador Random

Option -> 3

§		§
§	SETTINGS	§
§		§
§		§
§	1- Execution Time	§
§	2- Depht	§
§	3- Initial Player	§
§	0- Home Menu	§
§		§
§		§

Option ->

Option ->

4.4 QUIT

O menu que permite ao utilizador sair do programa, sem ter a necessidade de realizar o jogo até o fim ou atingir um dado objetivo.

O utilizador pode efectuar a paragem do jogo, usando a opção 0 do menu principal do jogo do cavalo.

5. Output

5.1 CONSOLE OUTPUT

O output de console é o resultado obtido das jogadas ou seja:

1. Estado;
2. Número de nós gerados;
3. Número de nós expandidos;
4. Número de nós avaliados;
5. Número de cortes alfa e beta;
6. Tempo que o algoritmo dispendeu a devolver a solução
7. Hora da jogada

5.2 FILE OUTPUT

O output do ficheiro é o resultado obtido através do algoritmo implementado e são visualizados no output console e no ficheiro de log.dat, exemplo dos resultados obtidos no ficheiro solução segundo o directoria:

1. Estado;
2. Número de nós gerados;
3. Número de nós expandidos;
4. Número de nós avaliados;
5. Número de cortes alfa e beta;
6. Tempo que o algoritmo dispendeu a devolver a solução

```
"C:\\Users\\andre.camuenhi\\Documents\\IPS_130221037\\IA1920\\Projetos\\Projecto_2019_2020_IA_P2\\Projeto\\log.dat"
```

6. Glossário

neste glossario mostramos os acrónimos com suas convenções utilizadas

- Atomos: palavras para todos tipos de dados que não são sequenciais, em que neste projeto utilizamos simbolos, numeros(inteiros, reais), Boolean (T ou Nil) e strings
- Listas: tipos de dados dinamicos, usamos diversas listas com listas por exemplo: tabuleiro com 10 linhas e 10 colunas, lista de operadores (operadores no seu todo)
- Selectores: são listas do tipo abstratos, permitindo assim a construção e obter elementos, exemplo: cons, car, cdr etc.

- Macros: consistem em definir a estrutura sintática da linguagem para substituir as funções, neste projeto usamos first, second, third etc. nas funções com operadores, estados e pontos.
- Append: é uma função de manipulação de dados que permite acrescentar um valor a uma lista.
- Reverse: é uma função que permite inverter os caracteres numa lista.
- Concatenate: permite juntar duas ou mais palavras de uma lista.
- Operações matemáticas: são funções aritméticas que permitem efetuar as operações matemáticas ou cálculos matemáticos, nomeadamente sinais de mais, menos, multiplicação e divisão, função modular, Raiz quadrada entre outros.
- os tipos Booleanos ou Predicados: utilizamos null, listp, zerop, etc para verificar os dados e serem validados.
- Equal e eq: são operações de igualdade e tem a sua diferença na forma como são apresentadas.
- variáveis: permite guardar e inicializar os valores, por exemplo:
 - Let: avalia as expressões feitas em paralelo.
 - Let*: avalia as sequências das expressões.
 - defparameter: são variáveis que estão fora do ambiente Léxico, pois são visíveis em todo o programa, logo corre-se o risco de provocar efeitos laterais imprevisíveis e por convenção tem de se nomear com astériscos(*) à esquerda.
- Funções: permitem construir e avaliar um determinado programa, em Lisp o caso não é diferente e neste projeto existem funções para construir tabuleiro e avaliar as expressões.

.....
cut here

