

Projeto Nº1: Época Normal



Inteligência Artificial 19/20

Prof. Joaquim Filipe

Eng. Filipe Mariano

Jogo do Cavalo (Knight Game)

Manual de Utilizador

Realizado por:

João Gomes - 150221001

André Gastão - 130221037

21 de Dezembro de 2019

Índice

1. Introdução
2. Instalação
3. Configuração
4. Interface da Aplicação
5. Output

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 do jogo (Knight Game) originalmente criado a partir do problema matemático estudado em Inteligencia Artificial Knight's tour. Neste documento serão descritos todos os passos para que o utilizador consiga instalar e interagir com a aplicação do jogo.

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 auxílio do IDE LispWorks. A estrutura do projeto é composta por 4 ficheiros:

- projeto.lisp - Interação com o utilizador, escrita e leitura de ficheiros.
- puzzle.lisp - Implementação da resolução do problema incluindo seletores, operadores heurísticas e outras funções auxiliares.
- procura.lisp - Implementação dos algoritmos de procura BFS, DFS e A*.
- problemas.dat - Funções com os problemas de A) a f).

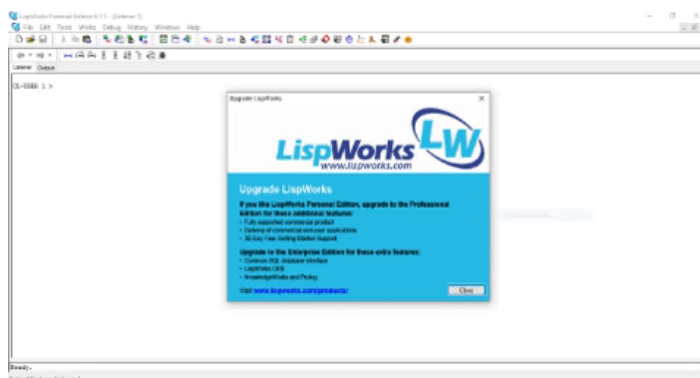
Configuração

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

```
(defun diretorio-atual ()
  "Define o caminho até ficheiros do projeto do root"
  (let ((path "C:\\Path Of Folder\\"))
    path
  )
)

(defun ficheiro-solucao ()
  (let ((ficheiro-path "C:\\Path Of Folder\\solucao.dat"))
    ficheiro-path
  )
)
```

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



Interface da Aplicação

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 visualização

Home Menu

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-Solve a Game                  $
$                2-Game Rules                    $
$                3-Show Boards                   $
$                4-Quit                          $
$                $                                $
$                $                                $
$                $                                $

```

Option ->

Solve a Game

A opção solve a game ou resolver o jogo permite mostrar o menu dos problemas de A a F, em que o utilizador deve escolher a opção 1 e carregar na tecla enter.

O novo menu mostra 6 opções dos problemas, na qual o utilizador pode escolher uma das 6 opções e sendo a setima (7) opção permite a saída do programa principal.

```

$                CHOOSE THE BOARD                $
$                $                                $
$                1-Problem A                    $
$                2-Problem B                    $
$                3-Problem C                    $
$                4-Problem D                    $
$                5-Problem E                    $
$                6-Problem F                    $
$                7-Home Menu                    $
$                $                                $
$                $                                $

```

Option ->

Solve Board

Choose The Problem

As opções problema A) à F), o utilizador pode escolher um dos problemas para que possa obter uma solução e atingir o objetivo.

```

$                CHOOSE THE BOARD                $
$                $                                $
$                1-Problem A                    $
$                2-Problem B                    $
$                3-Problem C                    $

```

```

$          4-Problem D          $
$          5-Problem E          $
$          6-Problem F          $
$          7-Home Menu          $
$                               $
$                               $

```

Option ->

Neste Menu o utilizador pode escolher correr logo o Algoritmo ou colocar manualmente o cavalo em Jogo e Proceder á Procura da Solução.

```

$          • GAME MODES •          $
$                               $
$                               $
$          1-GO TO ALGORITHM        $
$          2-CONFIGURE BOARD        $
$                               $
$          0-Home Menu              $
$                               $
$                               $

```

Option ->

Choose The Search Algorithm

Este menu permite escolher um dos algoritmos BFS,DFS e A* implementados para obter o resultado de um problema proposto

```

$          CHOOSE ALGORITHM          $
$          (search algorithm)         $
$                               $
$          1-Algorithm BFS            $
$          2-Algorithm DFS            $
$          3-Algorithm A*             $
$          4-Algorithm SMA*           $
$          0-Home Menu                $
$                               $
$                               $

```

Option ->

• BFS (Breadth First Search)

BFS ou opção 1, ao escolher este algoritmo o sistema irá resolver o problema escolhido.

• DFS (Depth First Search)

DFS ou opção 2: se utilizador optar por escolher o algoritmo DFS, terá de inserir em primeiro o valor da profundidade para obter a solução do problema escolhido.

• A*

A* ou opção 3 do menu algoritmos, o utilizador pode escolher a opção 3, onde o problema é resolvido através de heurística, baseando nas fórmulas fornecidas pelo enunciado. O menu heurística permite que o utilizador escolha uma das duas opções a seguir.

```

$          H                          $
$          (HEURISTIC)                $
$                               $
$          1-Base ( $h(x)=o(x)/m(x)$ )      $
$                               $

```

```

$          2-Base          $
$          0-Home Menu    $
$                               $
$                               $

```

Heuristic ->

Os resultados obtidos ao escolher o algoritmo A*, com a base da opção 1

O algoritmo A* com a base da opção 1 mostra os resultados na secção A* output, em que as estatísticas correspondem aos seguintes valores: G (Profundidade): 2, H (Heurística): 1.568, tamanho da solução: 6, Nós gerados: 8, Nós expandidos: 6, Penetração: 0.25, Pontos: 49 Objetivos: 70, Fator de ramificação: 1.085

- Outros resultados

Game Rules

o menu game rules mostra a descrição do jogo e suas regras para que o utilizador possa compreender sobre o jogo antes de o iniciar.

```

_____  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.
_____

```

Show Boards

Nesta opção o utilizador pode visualizar a lista dos problemas, em que cada um dos problemas pode ser resolvido.

- List of Boards

```

_____  • CHOOSE THE BOARD •  _____
$                               $
$                               $
$          1-Problem A          $
$          2-Problem B          $
$          3-Problem C          $
$          4-Problem D          $
$          5-Problem E          $
$          6-Problem F          $
$          7-Home Menu          $
$                               $
$                               $

```

Option ->

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 4 do menu principal do jogo do cavalo como mostra abaixo.

Output

CONSOLE OUTPUT

O output de console é o resultado obtido através do tabuleiro criado e dos algoritmos BFS, DFS, A*

FILE OUTPUT

O output do ficheiro é o resultado obtido através dos algoritmos implementados e são visualizados no output console e no ficheiro de solução.dat, exemplo dos resultados obtidos no ficheiro solução segundo o diretoria:

```
"C:\\Users\\andre.gastao\\Documents\\IA1920\\Projecto\\Projecto_2019_2020_IA_PreFinal\\Projecto_2019_2020_IA\\Projec
```



```
_____STATS_____
_____ 16:02:00 of Saturday, 12/21/2019 (GMT+0) _____
```

Final State:

```
((NIL NIL T NIL NIL NIL NIL NIL NIL NIL)
(NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL)
(NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL)
(NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL)
(NIL NIL NIL 22 NIL NIL NIL NIL NIL NIL)
(NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL)
(NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL)
(NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL)
(NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL)
(NIL NIL NIL NIL NIL NIL NIL NIL NIL NIL))
```

```
Algorithm: BFS
G (Depth): 2
H (Heuristic): 0
Solution Length: 6
Generated Nodes: 12
Expanded Nodes: 7
Penetration: 0.16666667
Points: 49
Objective: 70
Average branching factor: 1.198952
OBJECTIVE NOT REACHED
```