

Manual de Utilizador

Inteligencia Artificial 2020/21

BLOKUS



Docente: Joaquim Filipe

Realizado por :

Bernardo Mota nº201900947

Frederico Alcaria nº201701440

Acrónimos

- **BFS**: Algoritmo de busca em largura
- **DFS**: Algoritmo de busca em profundidade
- **A***: Algoritmo de procura informada
- **IDE***: integrated deveopment environment (ambiente de desenvolvimento integrado)

Introdução

Este manual visa a ser um guia compreensivo para a correta utilização do programa desenvolvido, utilizando a linguagem de programação funcional LISP. O objetivo deste programa é indicar quais os passos necessários para chegar ao objetivo dos problemas dados. Esta versão do Blokus usa um tabuleiro de 14 por 14, mas os problemas alteram as condições de jogo dos tabuleiros e definem os seus objetivos.

Instalação e utilização

Para poder executar o programa é necessário o IDE [LispWorks](#) ou outro que consiga interpretar a linguagem LISP.

Abrir e Compilar os ficheiros

No LispWorks vai ser preciso compilar o ficheiro `project.lisp`. Ir `File>Compile and Load` e escolher o ficheiro `project.lisp`. Os restantes ficheiros vão ser compilados ao iniciar o programa, irá ser pedido o path de onde se encontram os ficheiros do projeto necessários (`procura.lisp` , `puzzle.lisp` , `problemas.dat`).

Executar o Programa

Para executar o programa é necessário abrir um listener e chamar a função `iniciar` escrevendo (`iniciar`)

Navegar no Programa

Para navegar no programa é necessário escrever na consola o nº respetivo à opção que deseja escolher.

Input/Output

Tipos de input:

- Ficheiros: o programa vai ler o ficheiro dos problemas e mostrar todos os problemas disponíveis.
- Consola: A interação com o programa é através do listener. O programa lê o input e corre a opção associada ao nº introduzido.

O ficheiro problemas.dat está estruturado da seguinte maneira:

- (objetivo mínimo (problema))
- Objetivo mínimo: é o primeiro elemento da lista e define qual o objetivo mínimo necessário de alcançar para resolver o problema.
- Problema: Matriz 14 por 14 que contém o tabuleiro para o problema.

Ex:

```
(8
((0 0 0 0 2 2 2 2 2 2 2 2 2 2)
(0 0 0 0 2 2 2 2 2 2 2 2 2 2)
(0 0 0 0 2 2 2 2 2 2 2 2 2 2)
(0 0 0 0 2 2 2 2 2 2 2 2 2 2)
(2 2 2 2 2 2 2 2 2 2 2 2 2 2)
(2 2 2 2 2 2 2 2 2 2 2 2 2 2)
(2 2 2 2 2 2 2 2 2 2 2 2 2 2)
(2 2 2 2 2 2 2 2 2 2 2 2 2 2)
(2 2 2 2 2 2 2 2 2 2 2 2 2 2)
(2 2 2 2 2 2 2 2 2 2 2 2 2 2)
(2 2 2 2 2 2 2 2 2 2 2 2 2 2)
(2 2 2 2 2 2 2 2 2 2 2 2 2 2)
(2 2 2 2 2 2 2 2 2 2 2 2 2 2)
(2 2 2 2 2 2 2 2 2 2 2 2 2 2)))
```

Tipos de output:

- Ficheiros: o programa vai gerar um ficheiro com o nome resultados.dat que guarda toda a informação relacionada à execução do programa.
- Consola: A interação com o utilizador é feita através do listener. O programa mostra os menus com as várias opções possíveis, e quando necessário mostram um exemplo de input.

O ficheiro resultados.dat é gerado no fim da execução de um algoritmo e regista a sequência de estado até à solução e as estatísticas de execução:

Ex:

- --/-/-/-/-/ R E S U L T A D O /-/-/-/-/-/-/-/- -

Algoritmo utilizado: A*

Profundidade: 0

```
-- -- -- -- -- : : : : : :
-- -- -- -- -- : : : : : :
-- -- -- -- -- : : : : : :
-- -- -- -- -- : : : : : :
-- -- -- -- -- : : : : : :
-- -- -- -- -- : : : : : :
-- -- -- -- -- : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
```

Peças disponiveis: (10 10 15)

Profundidade: 1

```
# # _ _ _ _ _ : : : : : :
# # _ _ _ _ _ : : : : : :
_ _ _ _ _ _ _ : : : : : :
_ _ _ _ _ _ _ : : : : : :
_ _ _ _ _ _ _ : : : : : :
_ _ _ _ _ _ _ : : : : : :
_ _ _ _ _ _ _ : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
```

Peças disponiveis: (10 9 15)

Profundidade: 2

```
# # _ _ _ _ _ : : : : : :
# # _ _ _ _ _ : : : : : :
_ _ # # _ _ _ : : : : : :
_ # # _ _ _ _ : : : : : :
_ _ _ _ _ _ _ : : : : : :
_ _ _ _ _ _ _ : : : : : :
_ _ _ _ _ _ _ : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
```

Peças disponiveis: (10 9 14)

Profundidade: 3

```
# # _ _ _ _ _ : : : : : :
# # _ _ _ _ _ : : : : : :
_ _ # # _ _ _ : : : : : :
_ # # _ # # _ : : : : : :
_ _ _ _ # # _ : : : : : :
_ _ _ _ _ _ _ : : : : : :
_ _ _ _ _ _ _ : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
```

Peças disponiveis: (10 8 14)

Profundidade: 4

```
# # _ _ _ # _ : : : : : :
# # _ _ _ # # : : : : : :
_ _ # # _ _ # : : : : : :
_ # # _ # # _ : : : : : :
_ _ _ _ # # _ : : : : : :
_ _ _ _ _ _ _ : : : : : :
_ _ _ _ _ _ _ : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
```

Peças disponiveis: (10 8 13)

Profundidade: 5

```
# # _ _ _ # _ : : : : : :
# # _ _ _ # # : : : : : :
_ _ # # _ _ # : : : : : :
_ # # _ # # _ : : : : : :
# _ _ _ # # _ : : : : : :
# # _ _ _ _ _ : : : : : :
_ # _ _ _ _ _ : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
: : : : : : : : : : : :
```

Peças disponiveis: (10 8 12)

Profundidade: 6

```
# # _ _ _ # _ : : : : : : : : : : : : :
# # _ _ _ # # : : : : : : : : : : : : :
_ _ # # _ _ # : : : : : : : : : : : : :
_ # # _ # # _ : : : : : : : : : : : : :
# _ _ _ # # _ : : : : : : : : : : : : :
# # _ _ _ _ # : : : : : : : : : : : : :
_ # _ _ _ _ _ : : : : : : : : : : : : :
: : : : : : : : : : : : : : : : : : :
: : : : : : : : : : : : : : : : : : :
: : : : : : : : : : : : : : : : : : :
: : : : : : : : : : : : : : : : : : :
: : : : : : : : : : : : : : : : : : :
: : : : : : : : : : : : : : : : : : :
: : : : : : : : : : : : : : : : : : :
Peças disponiveis: (9 8 12)
```

Profundidade: 7

```
# # _ _ _ # _ : : : : : : : : : : : : :
# # _ _ _ # # : : : : : : : : : : : : :
_ _ # # _ _ # : : : : : : : : : : : : :
_ # # _ # # _ : : : : : : : : : : : : :
# _ _ _ # # _ : : : : : : : : : : : : :
# # _ _ _ _ # : : : : : : : : : : : : :
_ # _ _ _ # _ : : : : : : : : : : : : :
: : : : : : : : : : : : : : : : : : :
: : : : : : : : : : : : : : : : : : :
: : : : : : : : : : : : : : : : : : :
: : : : : : : : : : : : : : : : : : :
: : : : : : : : : : : : : : : : : : :
: : : : : : : : : : : : : : : : : : :
: : : : : : : : : : : : : : : : : : :
: : : : : : : : : : : : : : : : : : :
Peças disponiveis: (8 8 12)
```

Profundidade: 8

```
# # _ _ _ # _ : : : : : : : : : : : : :
# # _ _ _ # # : : : : : : : : : : : : :
_ _ # # _ _ # : : : : ~ : : : : : : : :
_ # # _ # # _ : : : : ~ : : : : ~ : : :
# _ _ _ # # _ : : : : ~ : : : : ~ : : :
# # _ # _ _ # : : : : ~ : : : : ~ : : :
_ # _ _ _ # _ : : : : ~ : : : : ~ : : :
: : : : ~ : : : ~ : : : ~ : : : ~ : : :
: : : : ~ : : : ~ : : : ~ : : : ~ : : :
: : : : ~ : : : ~ : : : ~ : : : ~ : : :
: : : : ~ : : : ~ : : : ~ : : : ~ : : :
: : : : ~ : : : ~ : : : ~ : : : ~ : : :
: : : : ~ : : : ~ : : : ~ : : : ~ : : :
: : : : ~ : : : ~ : : : ~ : : : ~ : : :
: : : : ~ : : : ~ : : : ~ : : : ~ : : :
Peças disponiveis: (7 8 12)
```

- --/-/-/-/-/E S T A T I S T I C A S/-/-/-/-/-- -

Factor de ramificação média: 3.4375

Número de nós gerados: 55

Número de nós expandidos: 16

```
Penetrância: 0.14545454
Tempo de execução em segundos: 0.013
- --/--/--/--/--/--/--/--/--/--/--/--/--/--/--/-- -- -
```

Exemplo de aplicação

Ao iniciar o programa irá ser pedido o file path até ao local onde se encontra o projeto.

```
Escreva o path da localizacao do projeto entre aspas  
Exemplo: 'C:/Users/username/Desktop/'
```

Ao inserir o path surgirá o menu com a seguinte interface, para escolher uma opção é só introduzir no listener o número correspondente à ação que quer realizar.

```
JOGO DO BLOKUS  
  
1 - Jogar  
0 - Sair
```

Se escolher 1 irá passar para o menu dos tabuleiros, se escolher 0 o programa fecha.

O menu dos tabuleiros apresenta todos os tabuleiros possíveis, que estão localizados no ficheiro problemas.dat.

```
JOGO DO BLOKUS  
  
Escolha o tabuleiro:  
  
1 : Tabuleiro 1  
2 : Tabuleiro 2  
3 : Tabuleiro 3  
4 : Tabuleiro 4  
5 : Tabuleiro 5  
6 : Tabuleiro 6  
0 - Voltar
```

Ao escolher o tabuleiro, o menu dos algoritmos vai aparecer com 3 escolhas possíveis:

- BFS - Breadth-First Search
- DFS - Depth-First Search
- A*

AVISO: O programa não prevê se o problema pode ser resolvido em tempo útil.

- Cuidado no caso do algoritmo BFS que ao começar a expandir nós de profundidade 7, visto que o problema tem uma ramificação muito elevada, dificilmente alcançará o nível 8 em tempo útil. Por essa razão não conseguirá resolver tabuleiros para além do tabuleiro 2.
- E especial atenção no algoritmo DFS, um nível de profundidade máxima demasiado baixo, pode tornar o problema impossível, por isso o tempo de execução torna-se idêntico ao BFS pois toda a árvore terá de ser expandida e mesmo assim nenhum nó objetivo poderá ser encontrado.

```
JOGO DO BLOKUS

Escolha o algoritmo:

1 - Breadth-First Search
2 - Depth-First Search
3 - A*
0 - Voltar
```

BFS (Breadth-First Search)

Se escolher o BFS o algoritmo vai correr no tabuleiro escolhido. Quando o algoritmo terminar irá aparecer todos os nós até à solução final e as estatísticas relacionadas.

DFS (Depth-First Search)

Se escolher o DFS irá aparecer um menu para inserir o limite máximo de profundidade. Quando o algoritmo terminar irá aparecer todos os nós até à solução final e as estatísticas relacionadas.

```
JOGO DO BLOKUS

Qual o limite de profundidade

> 0 - nível de profundidade
0 - Voltar
```

A*

Se escolher o A* irá aparecer um menu para escolher qual a heurística a utilizar. Existem duas disponíveis, uma base e uma original. Quando o algoritmo terminar irá aparecer todos os nós até à solução final e as estatísticas relacionadas.

```

      JOGO DO BLOKUS

      Escolha a heurística

      1 - Heurística base
      2 - Heurística original
      0 - Voltar
  
```

Conclusão do Programa

Quando o algoritmo terminar irá aparecer a sequência de estados até ao fim do problema e as estatísticas relacionadas à execução do algoritmo. Nas estatísticas irá ser possível ver:

- Factor de ramificação média
- Número de nós gerados
- Número de nós expandidos
- Penetrância
- Tempo de execução em segundos

Ex:

```

- --/--/--/--/E S T A T I S T I C A S/--/--/-- -
Factor de ramificação média: 21.36111
Número de nós gerados: 769
Número de nós expandidos: 36
Penetrância: 0.041612484
Tempo de execução em segundos: 0.416
- --/--/--/--/--/--/--/--/--/--/--/-- -
  
```

Toda a informação que é apresentada no final da execução também irá ser guardada no ficheiro resultados.dat. O utilizador depois terá a opção de continuar a usar ou fechar o programa.B