

Quartetto

Relatório Intercalar
Programação em Lógica

Mestrado Integrado em Engenharia Informática
e Computação

(18 de outubro de 2018)

Francisco Teixeira Ferreira _ up201605660@fe.up.pt
Nuno Tiago Tavares Lopes _ up201605337@fe.up.pt

Índice

<i>O Jogo : Quartetto</i>	<i>3</i>
<i>Representação do estado do jogo.....</i>	<i>6</i>
<i>Visualização do tabuleiro.....</i>	<i>8</i>

O Jogo : Quartetto

Quartetto é jogado no comum tabuleiro de 8x8.

Tabuleiro:

Cada jogador possui quatro peças, sendo que começam alinhadas nas quatro posições centrais das linhas com índices 1 e 8 (considerando que os índices do tabuleiro tradicional vão de 1 a 8).

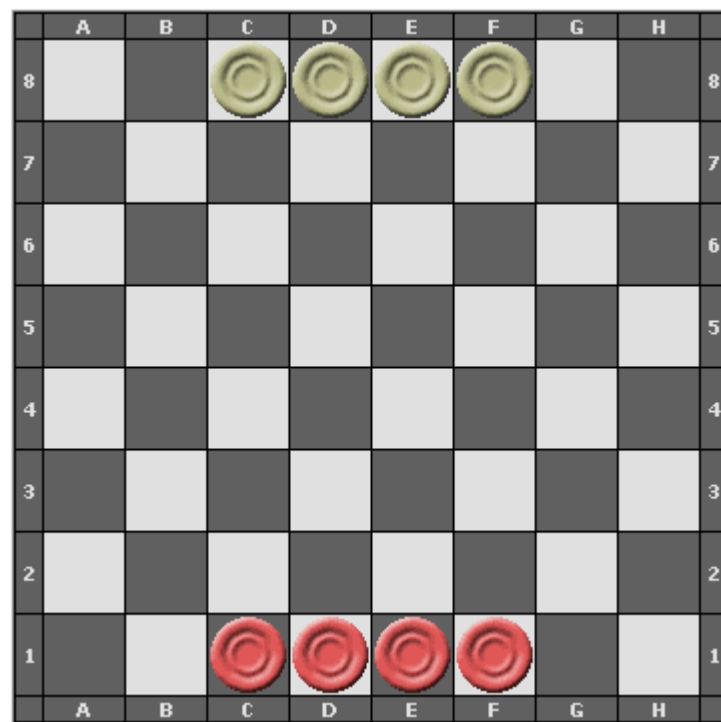


Fig. 1 - Estado inicial do tabuleiro

Objetivo:

O objetivo do jogo passa por colocar todas as peças de uma determinada cor nas seguintes condições:

1. Os pontos centrais das células devem ser vértices de um quadrado que tenha sido alvo de uma rotação.
2. O quadrado tem como dimensões mínimas 5x5.

Exemplos de Jogos Finalizados com Sucesso:

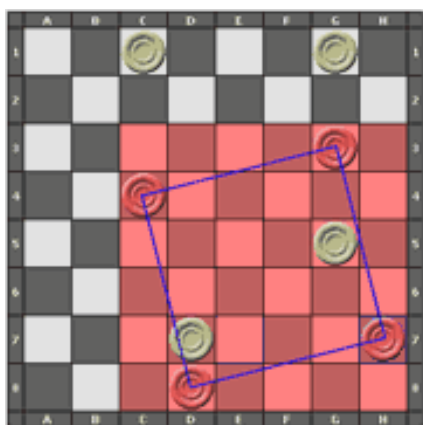


Fig. 2 - Vitória das peças pretas (6x6)

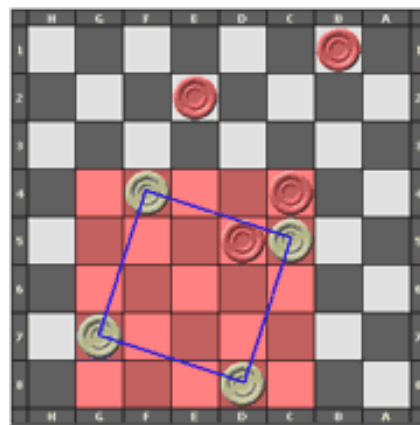


Fig. 3 - Vitória das peças brancas (5x5)

Exemplos de Jogadas que não finalizam um jogo:

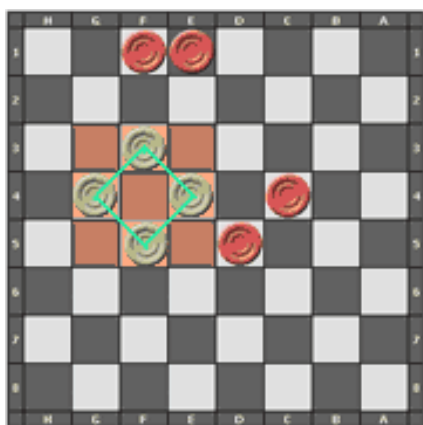


Fig. 5 - Peças brancas não ganham porque o quadrado é apenas de 3x3

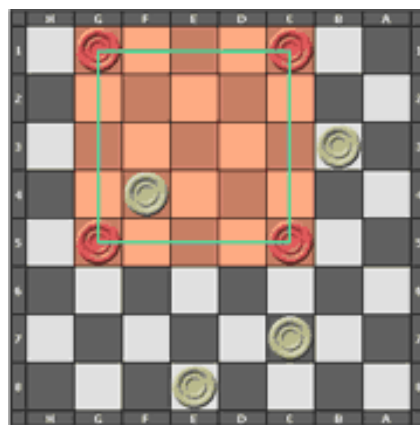


Fig. 4 - Peças pretas não ganham porque o quadrado formado não se encontra inclinado

Gameplay:

As movimentações do jogo são iniciadas pelo jogador que tenha as peças pretas, alternando depois sempre que houver a movimentação de uma peça.

As peças podem ser movidas o número de casas que for desejado pelo jogador, tanto horizontal como verticalmente, com as únicas condições de que não podem ficar numa casa que já esteja ocupada nem passar por cima dessa mesma casa (movimentação semelhante à torre no jogo de Xadrez).

Referências:

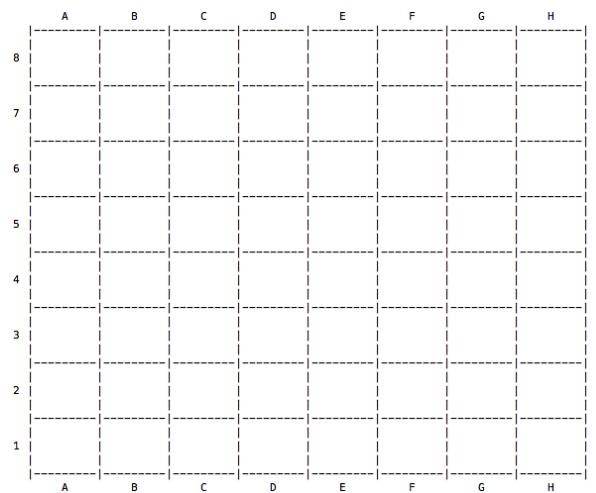
<http://www.iggamecenter.com/info/en/quartetto.html>

<https://boardgamegeek.com/boardgame/35164/quartetto>

Representação do estado do jogo

Situação vazia:

```
emptyBoard([
  [empty, empty, empty, empty, empty, empty, empty, empty],
  [empty, empty, empty, empty, empty, empty, empty, empty],
  [empty, empty, empty, empty, empty, empty, empty, empty],
  [empty, empty, empty, empty, empty, empty, empty, empty],
  [empty, empty, empty, empty, empty, empty, empty, empty],
  [empty, empty, empty, empty, empty, empty, empty, empty],
  [empty, empty, empty, empty, empty, empty, empty, empty],
  [empty, empty, empty, empty, empty, empty, empty, empty]
]).
```



	A	B	C	D	E	F	G	H
8								
7								
6								
5								
4								
3								
2								
1								

Fig. 6 - Tabuleiro vazio visto na consola

Situação inicial:

```
initialBoard([
  [empty, empty, white, white, white, white, empty, empty],
  [empty, empty, empty, empty, empty, empty, empty, empty],
  [empty, empty, empty, empty, empty, empty, empty, empty],
  [empty, empty, empty, empty, empty, empty, empty, empty],
  [empty, empty, empty, empty, empty, empty, empty, empty],
  [empty, empty, empty, empty, empty, empty, empty, empty],
  [empty, empty, empty, empty, empty, empty, empty, empty],
  [empty, empty, black, black, black, black, empty, empty]
]).
```

	A	B	C	D	E	F	G	H	
8			O				O		8
7									7
6							X		6
5			X						5
4							O		4
3									3
2				O				X	2
1				X					1
	A	B	C	D	E	F	G	H	

Fig. 7 - Situação inicial vista na consola

Situação final:

```
finalBoard([
  [empty, empty, white, empty, empty, empty, white, empty],
  [empty, empty, empty, empty, empty, empty, empty, empty],
  [empty, empty, empty, empty, empty, empty, black, empty],
  [empty, empty, black, empty, empty, empty, empty, empty],
  [empty, empty, empty, empty, empty, empty, white, empty],
  [empty, empty, empty, empty, empty, empty, empty, empty],
  [empty, empty, empty, white, empty, empty, empty, black],
  [empty, empty, empty, black, empty, empty, empty, empty]
]).
```

	A	B	C	D	E	F	G	H	
8			O	O	O	O			8
7									7
6									6
5									5
4									4
3									3
2									2
1			X	X	X	X			1
	A	B	C	D	E	F	G	H	

Fig. 8 - Situação final vista na consola

Visualização do tabuleiro

Segue-se o código utilizado para mostrar o tabuleiro na consola:

```
symbol(empty, S) :- S=' '.
symbol(black, S) :- S='X'.
symbol(white, S) :- S='O'.

printBoard(B) :-
    write('      A      B      C      D      E      F      G      H      '), nl,
    printBoard(B, 8),
    nl, write('      A      B      C      D      E      F      G      H      ').

printBoard([H|T], N) :-
    N > 0,
    write(' |-----|-----|-----|-----|-----|-----|-----| '), nl,
    write(' |           |           |           |           |           |           |           | '), nl,
    write(N), write(' | '),
    printLine(H),
    write(' '), write(N), nl,
    write(' |           |           |           |           |           |           |           | '), nl,
    N1 is N-1,
    printBoard(T, N1).

printBoard([], _) :-
    write(' |-----|-----|-----|-----|-----|-----|-----| ').

printLine([H|T]) :-
    write('      '), symbol(H,S),
    write(S), write('      | '),
    printLine(T).

printLine([]).

display_game(B, P) :-
    printBoard(B).
```

As figuras 6, 7 e 8, apresentadas anteriormente, representam os outputs na consola produzidos pelo código acima apresentado.