

Quartetto

Relatório Final

Programação em Lógica

Mestrado Integrado em Engenharia Informática

e Computação

(18 de novembro de 2018)

Francisco Teixeira Ferreira \_ up201605660@fe.up.pt

Nuno Tiago Tavares Lopes \_ up201605337@fe.up.pt

Índice

[Introdução 3](#_Toc530218219)

[O Jogo : Quartetto 4](#_Toc530218220)

[Representação do estado do jogo 7](#_Toc530218221)

[Visualização do tabuleiro 9](#_Toc530218222)

# 1 Introdução

**Introduzir aqui a estrutura do relatório e a sua descrição de cada tópico como a literallysofia**

# 2 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).

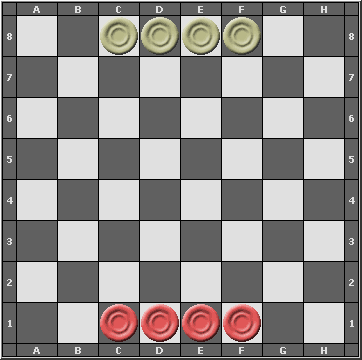


Fig. 1 - Estado inicial do tabuleiro

**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:**

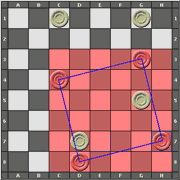


Fig. 2 - Vitória das peças pretas (6x6)

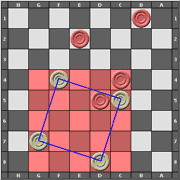
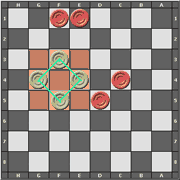
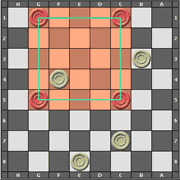


Fig. 3 - Vitória das peças brancas (5x5)



**Exemplos de Jogadas que não finalizam um jogo:**

Fig. 5 - Peças pretas não ganham porque o quadrado formado não se encontra inclinado

Fig. 4 - Peças brancas não ganham porque o quadrado é apenas de 3x3

**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>

# 

# 3 Lógica do Jogo

# 3.1 Representação do estado do jogo

Situação incial:

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]

]).

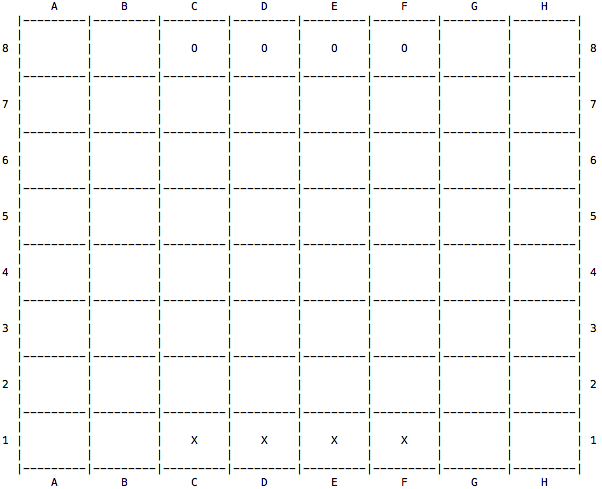


Fig. 6 - Situação inicial vista na consola

Exemplo de uma 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]

]).

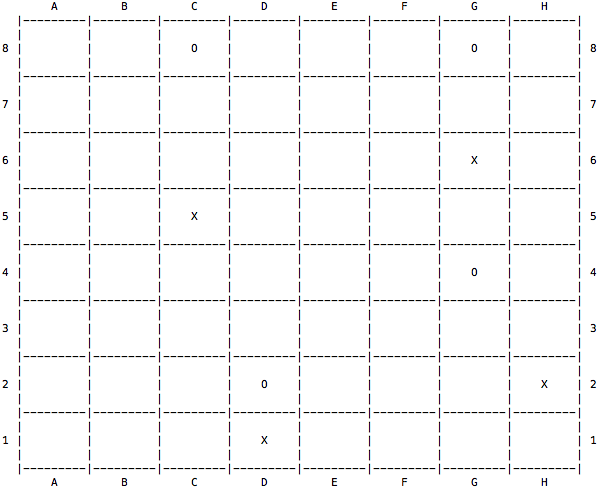


Fig. 7 - Situação final vista na consola

# 

# 3.2 Visualização do tabuleiro

Os predicados responsáveis pela visualização do tabuleiro na linha de comandos encontram-se no ficheiro *board.pl*.

Para imprimir o tabuleiro basta chamar o predicado ***display\_game(+Board, +Player)***, onde ***Board*** é um tabuleiro de um estado de jogo e o ***Player*** é o jogador atual. Consoante o jogador, o predicado chama outros dois predicados, ***printBoard(+Board)*** e ***printBoardReverse(+Board)***, para inverter o tabuleiro, simulando assim um jogo de tabuleiro na vida real. Por sua vez, cada um destes predicados faz uso de outros predicados com funções cada vez mais específicas para imprimir o tabuleiro de uma forma simples, formatada e concisa sempre que necessário.

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

# 3.2 Lista de Jogadas Válidas

# 4 Interface com o Utilizador

Os predicados responsáveis pela interface com o utilizador estão maioritariamente no ficheiro *menus.pl*.

A interface foi realizada com o intuito de ser simples e fácil para o utilizador comum, para tal, todos os menus estão devidamente identificados e para navegar entre eles basta selecionar o respetivo identificador, colocar um *Ponto* (.) e pressionar *Enter*.

