

---

# MARTIAN CHESS

---

Programação em Lógica

2016/2017

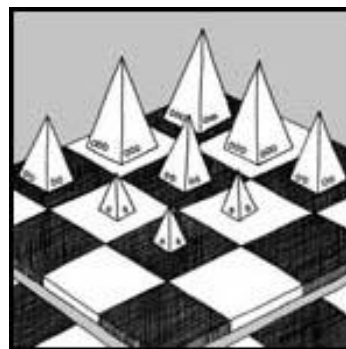
Inês de Carvalho Pereira Ferreira      up201305866

Tomás Vieira Caldas      up201404990

## 1. Descrição

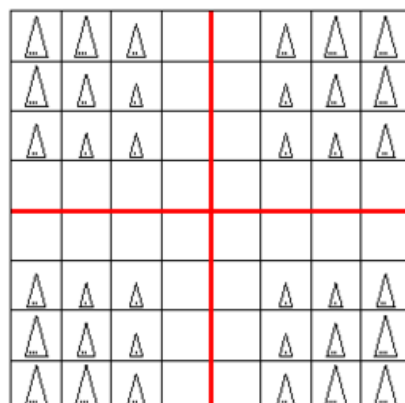
### 1.1 História

Martian Chess é um jogo de estratégia, inventado por Andrew Looney. É jogado num tabuleiro de xadrez, com peças Icehouse. Estas peças foram desenhadas para um jogo chamado Icehouse, também ele inventado por Andrew Looney e John Cooper, inspiradas em histórias curtas de ficção científica que Andrew estava a escrever.



### 1.2 Regras

O jogo pode ser jogado por 2 ou 4 jogadores, sendo que no desenvolvimento do projeto desta unidade curricular, vamos apenas abordar o modo de jogo para 2 jogadores, sendo o tabuleiro representado no lado direito da imagem.



Cada jogador começa com 9 peças, 3 grandes (rainhas), 3 médias (drones) e 3 pequenas (peões). A cor das peças é irrelevante para este jogo, sendo que as peças que cada jogador controla são as que se encontram no quadrante que lhe pertence. O jogador captura as peças inimigas que se encontra no lado do oponente, mas este acaba por ficar com a peça que capturou a sua.

Quando um jogador não controla nenhuma rainha, pode mover um peão para a mesma casa que um drone (ou vice-versa), retirar ambar do tabuleiro e colocar nessa casa uma rainha. Se um jogador não controlar nenhum drone, pode fazer o mesmo com dois peões.

O jogo acaba quando não resta nenhuma peça no seu lado do tabuleiro. As pontuações são calculadas pelas peças que capturaram, sendo que cada rainha vale 3 pontos, cada drone vale 2 pontos e cada peão vale 1 ponto.

Fonte: <http://www.looneylabs.com/rules/martian-chess>

## 2. Abordagem do jogo

### 2.1 Representação do estado do jogo

A representação do tabuleiro inicial será feita com uma lista de listas:

```
board ([      [queen,queen,drone,vazio],
               [queen,drone,pawn,vazio],
               [drone,pawn,pawn,vazio],
               [vazio,vazio,vazio,vazio],
               [vazio,vazio,vazio,vazio],
               [vazio,pawn,pawn,drone],
               [vazio,pawn,drone,queen],
               [vazio,drone,queen,queen]
        ]).
```

Posteriormente, temos alguns predicados que transformam cada átomo na representação que queremos que apareça no output.

- `translate(vazio,'')`.
- `translate(queen,'Q')`.
- `translate(drone,'D')`.
- `translate(pawn,'P')`.

## 2.2 Visualização do tabuleiro

```
| ?- play_game(_X) .
Q | Q | D |   |
Q | D | P |   |
D | P | P |   |
  |   |   |   |
  |   |   |   |
  | P | P | D |
  | P | D | Q |
  | D | Q | Q |
```

## 2.3 Movimentos

Cada tipo de peça tem um movimento característico. As rainhas movimentam-se quantas casas o jogador quiser, em todas as direções (horizontal, vertical e diagonal), não podendo passar por cima de outras peças. Os drones podem movimentar-se na vertical e horizontal (nunca diagonal), uma ou duas casas de cada vez. Os peões podem movimentar-se na diagonal, apenas uma casa de cada vez.

As jogadas serão feitas com *move* (*xi*, *yi*, *xf*, *yf*), sendo *x* inicial, *y* inicial, *x* final e *y* final, respetivamente. Dentro desta função terá a validação da jogada, isto é, verifica qual a peça que se encontra nas coordenadas iniciais e se o movimento para as coordenadas finais é válida para aquela peça.