

# MOD X

## Relatório Intercalar



Mestrado Integrado em Engenharia Informática e  
Computação

Programação em Lógica

### **Grupo ModX3:**

António Manuel Vieira Ramadas - 201303568

Rui Miguel Teixeira Vilares - 201207046

Faculdade de Engenharia da Universidade do Porto  
Rua Roberto Frias, sn, 4200-465 Porto, Portugal

8 de Outubro de 2015

# 1 O Jogo MOD X

## 1.1 Contextualização

O MOD X é um jogo de tabuleiro, nascido em 2015, recomendado para jogadores com mais de 15 anos de idade. É um jogo de estratégia, divertido e fácil de aprender. Cada partida reúne 2 a 4 jogadores e tem a duração prevista de 20 a 30 minutos.

## 1.2 Componentes do jogo

- 1 Tabuleiro 8x8;
- 56 peças de jogo em formato X (14 em cada cor);
- 72 marcadores de pontuação (18 em cada cor);
- 5 peças *Joker* em formato X (brancas);



Figura 1: Caixa do MOD X

## 1.3 Objetivo do jogo

O objetivo deste jogo é criar padrões com peças coloridas (vermelho, preto, amarelo e laranja). Pretende-se assim, formar o maior número de padrões possível, de forma a conseguir a melhor pontuação. O vencedor é o jogador com mais pontos. Evidentemente, é também suposto bloquear os adversários de modo que não consigam construir esses padrões<sup>1</sup>.

## 1.4 Regras

- Define-se a ordem dos jogadores, cada um escolhe a cor das suas peças e define-se o limite de pontos<sup>2</sup>;
- Cada jogador inicia o jogo com 14 peças e 18 marcadores;

<sup>1</sup><https://boardgamegeek.com/boardgame/131387/mod-x>

<sup>2</sup><https://www.cryptozoic.com/games/mod-x>

- Os *Jokers* são dispostos inicialmente de forma aleatória no tabuleiro;
- No seu turno, cada jogador coloca uma peça no tabuleiro, em qualquer posição livre;
- Os padrões utilizados para ganhar pontos são o "X", o "+" e o "cinco em linha";

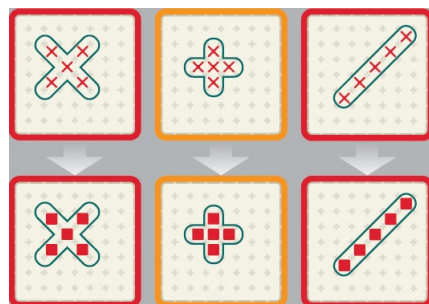
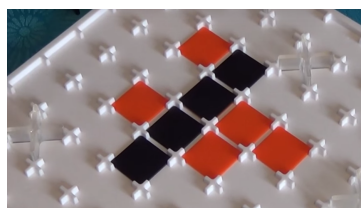


Figura 2: Padrões usados

- O jogador pode usar os *Jokers* para formar padrões, como se das suas próprias peças se tratassem;
- Quando um padrão é formado, retiram-se as peças de jogo e introduzem-se marcadores nessas posições. As peças de jogo podem agora voltar a ser utilizadas e as casas com marcadores também;
- Cada marcador colocado corresponde a um ponto.
- Caso tenha sido usado um *Joker* para formar um padrão, nessa posição não é introduzido um marcador. O *Joker* é agora colocado numa posição ao critério do jogador. Atenção, o *Joker* não pode ser usado para formar um novo padrão de imediato;



(a) Peças



(b) Marcadores

Figura 3: Substituição das peças por marcadores

- O primeiro jogador a atingir o número de pontos, determinado inicialmente, é o vencedor.
- O jogo pode também acabar quando um jogador já não dispõe de peças ou marcadores. Nesse caso, ganha o jogador com mais pontos até ao momento.

## 2 Representação do Estado do Jogo

### 2.1 Representação do estado inicial do tabuleiro

$[[[-1, -1], [-1, -1], [-1, -1], [-1, -1], [-1, -1], [-1, -1], [-1, -1], [-1, -1],$   
 $[-1, -1], [0, -1], [1, -1], [2, -1], [2, -1], [1, -1], [0, -1], [-1, -1],$   
 $[-1, 11], [0, 11], [1, 11], [2, 11], [2, 11], [1, 11], [0, 11], [-1, 11],$   
 $[-1, 22], [0, 22], [1, 22], [2, 22], [2, 22], [1, 22], [0, 22], [-1, 22],$   
 $[-1, -1], [0, -1], [1, -1], [2, -1], [2, -1], [1, -1], [0, -1], [-1, -1],$   
 $[-1, 11], [0, 11], [1, 11], [2, 11], [2, 11], [1, 11], [0, 11], [-1, 11],$   
 $[-1, 22], [0, 22], [1, 22], [2, 22], [2, 22], [1, 22], [0, 22], [-1, 22],$   
 $[-1, -1], [-1, -1], [-1, -1], [-1, -1], [-1, -1], [-1, -1], [-1, -1], [-1, -1]]$

INSERIR IMAGEM

### 2.2 Representação de um estado intermédio do tabuleiro

INSERIR CODIGO

INSERIR IMAGEM

### 2.3 Representação de um estado final do tabuleiro

INSERIR CODIGO

INSERIR IMAGEM

## 3 Visualização do Tabuleiro

## 4 Movimentos

**Falha se o *Joker* é posto numa posição errada**

placeJoker(Board, Player, TileNumber).

**Devolve as jogadas possíveis em ListOfMoves**

validMoves(Board, Player, ListOfMoves).

**Falha se não poder ser feito o movimento**

move(Board, Player, TileNumber, NewBoard).

**Avalia a jogada e devolve o seu valor em Value**

value(Board, Player, TileNumber, Value).

**Caso o jogo tenha acabado devolve o vencedor em Winner**

gameOver(Board, Winner).

**Jogada do computador em que consoante diferentes níveis de dificuldade (Level) devolve jogadas diferentes**

choose\_Move(Board, Level, Move).

**Põe as bases e tira as cruces do jogador Player**

putTiles(Board, Player, NewBoard).