

Programação em Lógica

*Aplicação em Prolog do Jogo de Tabuleiro Exo*

*MIEIC – Turma 5 – Grupo Exo\_2*

*(20 de outubro de 2019)*

Ana Filipa Campos Senra         **up201704077@fe.up.pt**

Cláudia Inês da Costa Martins**up201704136@fe.up.pt**

Índice

Descrição do Jogo

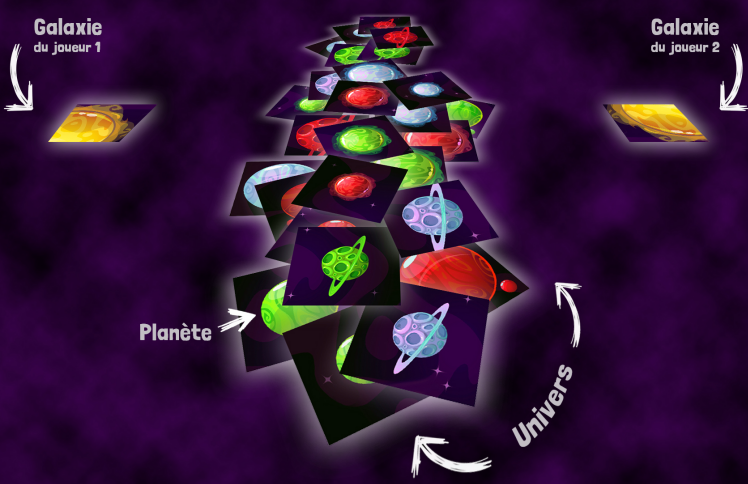


1. Objetivo do Jogo

O objetivo do Jogo Exo é criar combinações de planetas alinhando-os.

1. Configuração

No início do jogo, cada jogador cria a sua galáxia ao colocar uma estrela à sua frente. Formámos o universo ao misturar os 27 planetas entre as estrelas (ver a figura 1).



Galáxia do Jogador 2

Galáxia do Jogador 1

Universo

Planeta

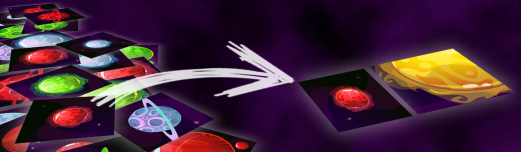
Figura –

O jogador inicial é que o observou as estrelas mais recentemente.

1. Como jogar

Os jogadores revezam-se no sentido dos ponteiros do relógio. Durante a sua vez, o jogador deve escolher um planeta disponível (um que não esteja coberto por outro planeta) do universo e colocá-lo na sua galáxia.

O primeiro planeta na galáxia do jogador deve ser colocado num dos 8 quadrados adjacentes à estrela. Posteriormente, os planetas devem ser colocados ou adjacentes à estrela ou a um planeta da galáxia. Um planeta não pode ser movido depois de colocado na galáxia.



Figura

Os planetas possuem diferentes características: tamanho (pequeno, médio, grande), cor (azul, vermelho, verde) e tipo (terreste, gasoso, anelado). O jogador ganha pontos quando cria linhas na sua galáxia com 3 planetas com características comuns.

1. Finalização do Jogo e Condições de Vitória

O jogo acaba quando cada jogador possui 13 planetas num jogo com dois jogadores ou 9 planetas num jogo com 3 jogadores. Os jogadores ganham 1 ponto por cada linha (horizontal, vertical ou diagonal) de três planetas com uma característica em comum. Se uma linha de planetas partilha mais de que uma característica, um ponto é alocado por cada característica partilhada.



3 planetas pequenos = 1 ponto

3 planetas verdes = 1 ponto

3 planetas anelares = 1 ponto

Figura

O jogador com mais pontos é declarado vencedor. Em casa de empate, o jogador com mais planetas adjacentes à sua estrela ganha.

Modelação do Jogo em Prolog

Representação interna do estado do jogo

Uma vez que neste jogo cada jogador tem o seu tabuleiro de jogo, são exemplificadas a seguir 6 representações do estado de jogo: duas iniciais, duas intermédias e duas finais, uma para cada jogador.

**Situação inicial**

Jogador 1:

initialBoard ([

[sun]]).

Jogador 2:

initialBoard ([

[sun]]).

**Situação intermedia**

Jogador 1:

intermediateBoard ([

[empty, planet (medium, white, terrestrial), empty],

[empty, planet (medium, green, terrestrial), planet (small, green, gaseous)],

[empty, planet (small, red, terrestrial), planet (large, green, gaseous)],

[planet (small, red, ringed), sun, empty]]).

Jogador 2:

intermediateBoard ([

[planet (large, green, terrestrial), empty, empty],

[planet (small, green, terrestrial), empty, planet (medium, white, gaseous)],

[sun, planet (medium, green, gaseous), empty],

[planet (small, red, gaseous), empty, planet (small, green, ringed)]]).

**Situação final**

Jogador 1:

finalBoard ([

[empty, empty, planet (small, white, ringed), planet (medium, white, ringed)],

[empty, empty, planet (medium, white, terrestrial), planet (medium, white, gaseous)],

[empty, planet (medium, green, ringed), planet (medium, green, terrestrial), planet (small, green, gaseous)],

[empty, empty, planet (small, red, terrestrial), planet (large, green, gaseous)],

[empty, planet (small, red, ringed), sun, planet (large, red, ringed)],

[planet (medium, red, gaseous), planet (small, white, terrestrial), empty, empty]]).

Jogador 2:

finalBoard ([

[planet (large, white, ringed), empty, empty],

[planet (large, green, ringed), empty, empty],

[planet (large, green, terrestrial), empty, planet (medium, white, terrestrial)],

[planet (small, green, terrestrial), empty, planet (medium, white, gaseous)],

[sun, planet (medium, green, gaseous), planet (medium, red, ringed)],

[planet (small, red, gaseous), planet (small, white, gaseous), planet (small, green, ringed)],

[planet (large, red, gaseous), empty, planet (large, white, ringed)]]).

**Jogador 1: Jogador 2:**

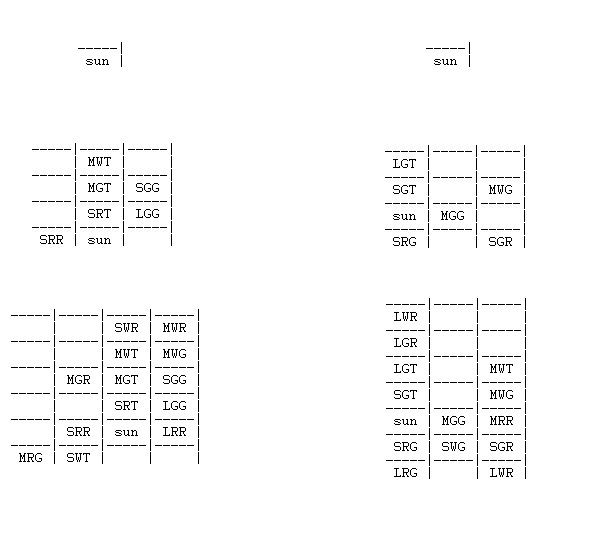


Figura 4. Situação Inicial.

**Jogador 1: Jogador 2:**

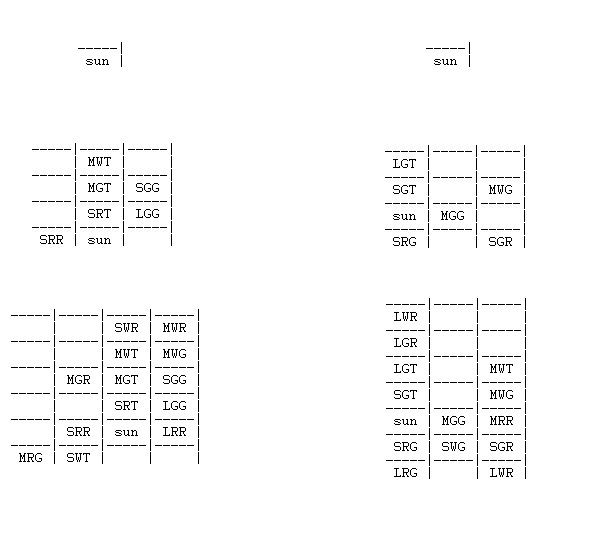
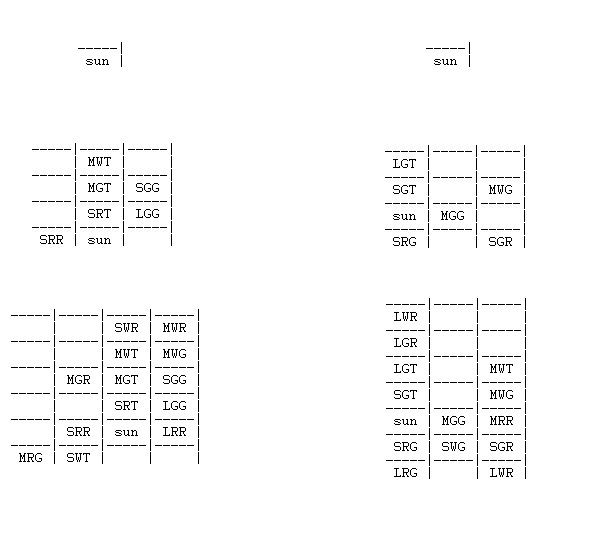


Figura 5. Situação Intermédia.



**Jogador 1: Jogador 2:**

Figura 6. Situação Final.

Visualização do tabuleiro em modo de texto

/\* Display game function \*/

 display\_game (Board, Player): -

    board (Board, Player).

/\* Writing Divisions \*/

writeDivisions (0): - write('\n').

writeDivisions(N): -

    N > 0,

    N1 is N - 1,

    write ('-----|'),

    writeDivisions(N1).

/\* Prints the Matrix \*/

printMatrix ([]).

printMatrix ([Head | Tail]): -

    length (Head, LenList),

    write('\n'),

    writeDivisions (LenList),

    write (' '),

    printLine (Head),

    printMatrix (Tail).

/\* Prints each Line \*/

printLine ([]).

printLine ([Head | Tail]): -

    symbol (Head, S),

    write (S),

    write (' | '),

    printLine (Tail).

/\*Representing a play \*/

symbol (empty, '   ').

symbol (sun, 'sun').

symbol (planet (Size, Colour, Type), S): -

    symbolSize (Size, S1),

    symbolColour (Colour, S2),

    symbolType (Type, S3),

    atom\_concat (S1, S2, SF1),

    atom\_concat (SF1, S3, S).

symbolSize(small,'S').

symbolSize(medium,'M').

symbolSize(large,'L').

symbolColour(red,'R').

symbolColour(green,'G').

symbolColour(white,'W').

symbolType(terrestrial,'T').

symbolType(gaseous,'G').

symbolType(ringed,'R').