



EXO

APLICAÇÃO EM PROLOG PARA UM JOGO DE TABULEIRO

INÊS ALVES – UP201605335@FE.UP.PT

NUNO CARDOSO – UP201706162@FE.UP.PT

Índice

Introdução	2
Regras	2
Pontuação	3
Representação interna do estado do jogo	4
Tabuleiro	4
Peças	4
Visualização do tabuleiro em modo de texto	5

Introdução

O “Exo” é um jogo de tabuleiro criado pelo designer francês Léandre Proust cujo tema principal é o espaço. O objetivo do jogo passa por criar galáxias e sistemas planetários que giram em torno de uma estrela. Os planetas são diferenciados pelas suas características, relativamente a tamanho, cor e gênero.

Regras

O jogador que começa o jogo é aquele que mais recentemente olhou para as estrelas do céu - a primeira jogada consiste em colocar uma peça com uma estrela à sua frente. Originalmente, este jogo não tem um tabuleiro definido, contudo decidimos criar um de dimensões 14x14 para cada jogador, de modo a simplificar e a assegurar uma melhor jogabilidade por parte do utilizador

Após ambos os jogadores terem posicionado a sua estrela, têm à sua escolha várias peças, num total de 27. Cada peça representa um planeta e estas estão dispostas de forma aleatória, aglomeradas e podendo estar sobrepostas. Cada jogador, à vez, apenas poderá escolher peças que não estejam cobertas, ou seja, que tenham outras peças por cima de si. Como este sistema é de difícil implementação, não contribuindo em nada para a jogabilidade, decidimos utilizar três “stacks” para representar montes de peças, cada uma inicialmente com nove peças aleatórias. Assim, o jogador apenas poderá escolher uma das três peças do topo de cada monte, considerando-se as outras como peças cobertas.



A peça escolhida é, então, colocada num dos quadrados adjacentes à estrela (na vertical, horizontal ou diagonal). Depois desta primeira jogada, as próximas peças terão de ser colocadas ou, igualmente, adjacentes à estrela, ou adjacentes aos restantes planetas do tabuleiro.

Figura 1: Exemplo do jogo na vida real

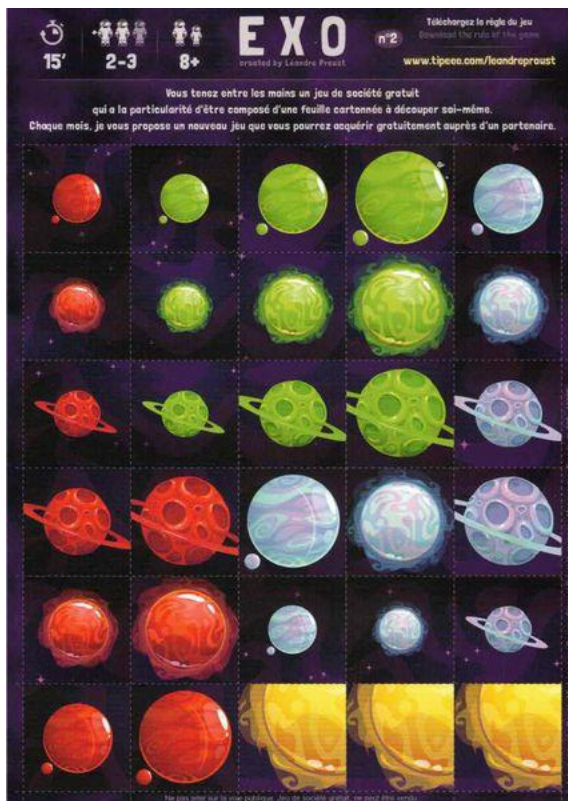


Figura 2: As trinta peças do jogo

Cada planeta tem três características: tamanho (pode ser pequeno, médio ou grande), cor (vermelho, azul, verde) e tipo (terrestre, gasoso, com anéis). Após ser colocada, uma peça não pode ser movida. O objetivo do jogo é, essencialmente, formar grupos de planetas com o maior número de características em comum.

O jogo termina quando ambos os jogadores tiverem 13 planetas na sua galáxia (sobrará sempre uma peça, visto que eram 27 no início do jogo).

Pontuação

Cada jogador ganha um ponto a cada grupo de três planetas que partilhem uma característica. Estes grupos podem ser uma linha de três planetas na vertical, horizontal ou diagonal. Por exemplo, se tivermos uma fila de três planetas com a mesma cor e tipo, a pontuação é dois - um ponto por cada característica partilhada.

No caso de empate, ganha o jogador que tiver mais planetas adjacentes à sua estrela.

Representação interna do estado do jogo

Tabuleiro

Como o máximo de peças que podemos ter em jogo numa direção é 14 (1 estrela e 13 planetas), decidimos criar, como dito anteriormente, um tabuleiro 14x14. Para isto criamos uma lista de listas, em que cada lista representa uma linha do tabuleiro.

```
init_board(Board1, Board2):-
    Board1 = [ [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
                [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
                [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
                [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
                [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
                [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
                [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
                [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
                [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
                [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
                [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
                [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
                [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
                [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0] ],
    Board2 = [ [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
                [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
                [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
                [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
                [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
                [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
                [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
                [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
                [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
                [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
                [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
                [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
                [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
                [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0] ].
```

Figura 3: Representação interna dos tabuleiros.

Peças

Para representar as diferentes peças disponíveis para escolha utilizamos 3 listas de 9 peças cada uma, como exemplificado na figura. Numa fase posterior, pretendemos que estas stacks tenham as peças em posições aleatórias.

Cada planeta tem três características diferentes, representadas em cada peça por 3 caracteres:

- Um para identificar a cor do planeta: R (vermelho) / G (verde) / B (azul);
- Um para o tamanho: 1 (pequeno) / 2 (médio) / 3 (grande);
- Um para o tipo: X (terrestre) / Y (gasoso) / Z (com anéis).

Para simplificar, na visualização do tabuleiro decidimos representar estes grupos de caracteres por uma única letra do alfabeto. Desta forma, criamos uma lista com a legenda dos planetas, como mostra a figura:

```
| A - R1X | B - R2X | C - R3X | D - R1Y | E - R2Y | F - R3Y | G - R1Z | H - R2Z | I - R3Z |
| J - G1X | K - G2X | L - G3X | M - G1Y | N - G2Y | O - G3Y | P - G1Z | Q - G2Z | R - G3Z |
| S - B1X | T - B2X | U - B3X | V - B1Y | W - B2Y | X - B3Y | Y - B1Z | Z - B2Z | % - B3Z |
```

Figura 4: Legenda

Visualização do tabuleiro em modo de texto

Com a função `display_game(.....)` mostramos no ecrã os dois tabuleiros do jogo (chamada a `display_boards()`), legendados com “PLAYER1” e “PLAYER2”, a pontuação atual (chamada a `display_score()`), as peças disponíveis em cada monte para escolha (chamada a `display_stacks()`) e, por fim, uma legenda das peças para auxiliar os utilizadores na sua escolha (chamada a `display_key()`).

Para iniciar o jogo no terminal, deve-se escrever “play(1)”, para ser o jogador 1 a começar, ou “play(2)” para ser o jogador 2. Em qualquer outro caso, a resposta será “no”.

[illegible]

Figura 5: Exemplo de início do jogo

[illegible]

Figura 6: Exemplo de uma jogada intermédia

```

      PLAYER 1
| 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| 0 0 0 0 0 0 0 N 0 0 0 0 0 0 0 0 |
| 0 0 0 0 0 0 0 M 0 0 0 0 0 0 0 0 |
| 0 C F P X T * U Y V W 0 0 0 0 |
| 0 0 0 0 0 0 0 S 0 0 0 0 0 0 0 0 |
| 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |

      PLAYER 2
| 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| 0 0 0 0 K 0 0 0 0 0 0 0 0 0 0 0 |
| 0 0 0 J * G Q 0 0 0 0 0 0 0 0 0 |
| 0 0 0 0 D 0 0 0 0 0 0 0 0 0 0 0 |
| 0 0 0 0 A 0 0 0 0 0 0 0 0 0 0 0 |
| 0 0 0 0 B 0 0 0 0 0 0 0 0 0 0 0 |
| 0 0 0 0 Z 0 0 0 0 0 0 0 0 0 0 0 |
| 0 0 0 0 H 0 0 0 0 0 0 0 0 0 0 0 |
| 0 0 0 0 E 0 0 0 0 0 0 0 0 0 0 0 |
| 0 0 0 0 % 0 0 0 0 0 0 0 0 0 0 0 |
| 0 0 0 0 L 0 0 0 0 0 0 0 0 0 0 0 |
| 0 0 0 0 R 0 0 0 0 0 0 0 0 0 0 0 |
| 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |

      SCORE: 3 4

|      |      | I
| A - R1X | B - R2X | C - R3X | D - R1Y | E - R2Y | F - R3Y | G - R1Z | H - R2Z | I - R3Z |
| J - G1X | K - G2X | L - G3X | M - G1Y | N - G2Y | O - G3Y | P - G1Z | Q - G2Z | R - G3Z |
| S - B1X | T - B2X | U - B3X | V - B1Y | W - B2Y | X - B3Y | Y - B1Z | Z - B2Z | % - B3Z |
Player1: I am playing bro

```

Figura 7: Exemplo de fim do jogo, onde só sobra uma peça (peça I)