



Mestrado Integrado em Engenharia Informática e Computação

Programação em Lógica

Syrtis

GRUPO 4: SYRTIS

ANTÓNIO PEDRO ARAÚJO FRAGA (UP201303095)

CAROLINA MOREIRA

O jogo Syrtis

História

O **Syrtis** é um jogo de dois jogadores inspirado nas areias movediças do Mar Mediterrâneo, foi concebido e desenhado por David Vander Laan em 2014, um jogo que se enquadra na categoria de Estratégia Abstracta.

Constituição do tabuleiro

Este jogo tem um tabuleiro com 36 ou 16 “casas”, casas essas que podem ser de quatro tipos, circulares ou quadradas, com possibilidade de uma coloração clara ou escura. As peças utilizadas são constituídas por quatro torres, duas redondas, claras, e duas quadradas, escuras.

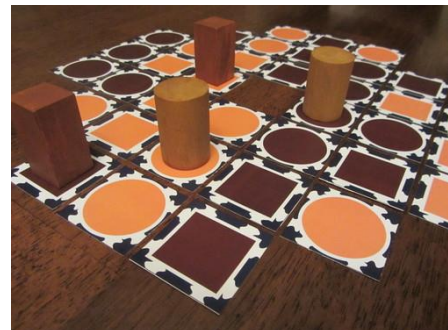


Figura 1 (um exemplo de um tabuleiro de jogo)

Distribuição de casas

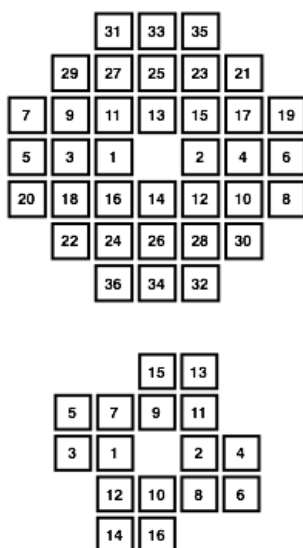


Figura 2 (disposição de casas)

Inicialmente é escolhida aleatoriamente uma casa para ser colocada na posição 1. Depois disso, é escolhida outra casa de cor e forma oposta para ser colocada na posição 2. O processo de escolha aleatória para as posições ímpares e a escolha de uma casa com cor e forma oposta para as posições pares continua até o tabuleiro estar completo, criando-se assim um tabuleiro anti-simétrico.

Um dos jogadores deve escolher a posição inicial das torres, cada uma delas deve estar situada numa casa desocupada com a mesma forma e cor. O jogador que não escolheu a disposição das peças, deve escolher com que peças jogar, claras ou escuras, as peças claras jogam primeiro.

Para jogos mais rápidos são usadas 4 casas de cada tipo, e deve-se seguir a disposição da figura da esquerda.

O que são ilhas?

Uma ilha é uma casa ou um conjunto de casas que têm a mesma forma ou a mesma coloração (casas que se apresentem na diagonal não fazem parte da ilha). Há assim quatro tipos de ilhas, ilhas de casas redondas, de casas quadradas, de casas claras e de casas escuras.

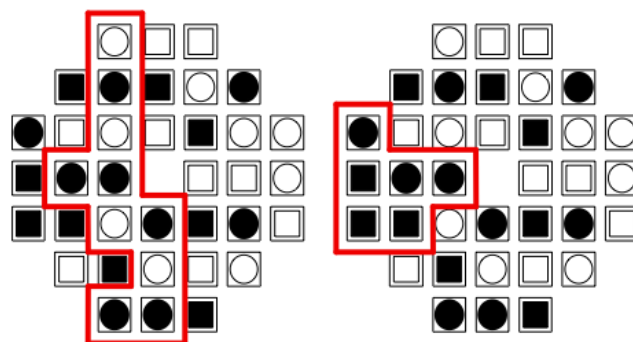


Figura 3

Movimentos

Um jogador pode fazer exatamente um movimento: **mover uma torre**, **afundar uma casa**, **mover uma casa** ou **passar**. No fim de cada jogada, todas as casas devem estar interligadas.

Mover uma torre: Se estás a jogar com peças claras, podes mover uma torre clara para qualquer casa desocupada de uma ilha de cor clara ou uma ilha de casas circulares em que a peça se encontre. Se estás a jogar com peças escuras, podes mover a torre para qualquer casa desocupada de uma ilha de casas escuras ou uma ilha de casas quadradas em que a peça se encontre.

Afundar uma casa: (1) Um jogador pode remover uma casa que esteja adjacente a uma casa onde esteja situada uma das peças do jogador. (2) Um jogador pode afundar uma casa desocupada. (3) Um jogador pode afundar uma casa que tenha pelo menos um lado aberto, ou seja, uma casa que não tenha uma casa vizinha em qualquer um dos lados. (*Relembrar que as casas devem estar interligadas ao final de cada ronda*)

Mover uma casa: Um jogador pode mover uma casa ocupada por uma das suas torres através de qualquer número de espaços vazios interligados, em qualquer número de direções. Quando um jogador move uma casa não pode aumentar a altura ou largura do tabuleiro. Por altura e largura do tabuleiro entenda-se caixa delimitada na figura à direita.

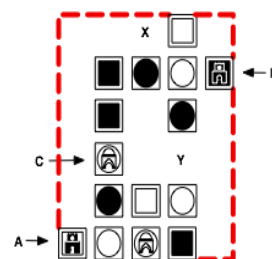


Figura 4

Passar: Um jogador pode passar quando bem entender, mas é obrigado a passar se não pode tomar mais nenhuma ação.

Fim do jogo

Há três possibilidades de chegar ao fim do jogo:

Completar uma ilha: Um jogador é vencedor se todas as casas com cor ou forma igual à das suas torres estiverem ligadas entre si. Se isto acontecer quando o tabuleiro é inicialmente construído, o jogador que coloca as torres no tabuleiro deve trocar as casas entre os pares de posições que estas são inicialmente colocadas ex: (11 e 12) (7 e 8) (25 e 26).

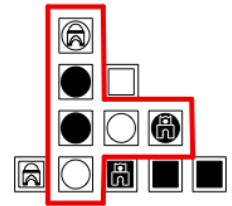


Figura 5 (uma situação de vitória para as peças claras)

Afundar peças: Um jogador é vencedor se afundar quatro casas sem que o jogador oponente afunde alguma. Perde assim uma corrida contra areias movediças.

Iniciativa: Esta regra impede empates. O jogo termina em qualquer uma das seguintes situações:

- Ambos os jogadores completam uma ilha ao mesmo tempo.
- Os jogadores passam em turnos consecutivos.
- Um jogador passa 4 vezes consecutivas.

Nesta situação, o jogador vencedor é aquele que tem mais peças afundadas desde que o outro jogador tenha afundado alguma peça. Numa situação improvável em que nenhuma peça tenha sido afundada, ganha o jogador que jogou primeiro.

Representação do estado de jogo

Para uma melhor jogabilidade, é necessário ter conhecimento da cor e a forma das casas em que as peças (torres) se encontram, e como tal decidimos seguir o conselho do professor e usar o SWI-Prolog, tornando-se possível dar coloração aos caracteres.

No jogo original, as peças têm uma coloração clara e escura, mas para uma melhor visualização na consola, trocamos a cor clara por azul e a cor escura por vermelho.

Representação de um estado inicial do tabuleiro

L0 = ['vazio', 'vazio', 'o-azul', 'quadrado-vermelho', 'o-azul', 'vazio', 'vazio'],

L1 = ['vazio', 'quadrado-vermelho', 'o-azul', 'quadrado-vermelho', 'o-vermelho', 'quadrado-vermelho', 'vazio'],

L2 = ['o-vermelho', 'o-azul', 'quadrado-azul', 'quadrado-vermelho', 'o-vermelho', 'quadrado-azul', 'quadrado-azul'],

L3 = ['o-vermelho', 'quadrado-azul', 'quadrado-azul', 'vazio', 'o-vermelho', 'o-vermelho', 'quadrado-azul'],

L4 = ['o-vermelho', 'o-vermelho', 'quadrado-azul', 'o-azul', 'o-vermelho', 'quadrado-vermelho', 'quadrado-azul'],

L5 = ['vazio', 'o-azul', 'quadrado-azul', 'o-azul', 'quadrado-vermelho', 'o-azul', 'vazio'],

L6 = ['vazio', 'vazio', 'quadrado-vermelho', 'o-azul', 'quadrado-vermelho', 'vazio', 'vazio'],

Tabuleiro = [L0, L1, L2, L3, L4, L5, L6].

	0	1	2	3	4	5	6
0			()	[]	()		
1		[]	()	[]	()	[]	
2	()	()	[]	[]	()	[]	[]
3	()	[]	[#]		(*)	()	[]
4	()	()	[]	()	()	[]	[]
5		()	[]	(*)	[]	()	
6			[#]	()	[]		

Figura 6 (representação de um possível estado inicial)

Representação de um estado intermedio do tabuleiro

L0 = ['vazio', 'vazio', 'vazio', 'vazio', 'quadrado-azul', 'vazio', 'vazio'],

L1 = ['vazio', 'vazio', 'quadrado-vermelho', 'o-vermelho', 'o-azul', 'quadrado-vermelho', 'vazio'],

L2 = ['vazio', 'vazio', 'quadrado-vermelho', 'vazio', 'o-vermelho', 'vazio', 'vazio'],

L3 = ['vazio', 'vazio', 'quadrado-azul', 'vazio', 'vazio', 'vazio', 'vazio'],

L4 = ['vazio', 'vazio', 'o-vermelho', 'quadrado-azul', 'o-azul', 'vazio', 'vazio'],

L5 = ['vazio', 'quadrado-azul', 'o-azul', 'o-azul', 'quadrado-vermelho', 'vazio', 'vazio'],

L6 = ['vazio', 'vazio', 'vazio', 'vazio', 'vazio', 'vazio', 'vazio'],

Tabuleiro = [L0, L1, L2, L3, L4, L5, L6].

	0	1	2	3	4	5	6
0					[]		
1			[]	()	()	[#]	
2			[]		()		
3			[*]				
4			()	[]	()		
5		[#]	()	(*)	[]		
6							

Figura 7 (Um possível estado intermédio)

Representação de um estado final do tabuleiro

L0 = ['vazio', 'vazio', 'vazio', 'vazio', 'quadrado-azul', 'vazio', 'vazio'],

L1 = ['vazio', 'vazio', 'quadrado-vermelho', 'o-vermelho', 'o-azul', 'quadrado-vermelho', 'vazio'],

L2 = ['vazio', 'vazio', 'quadrado-vermelho', 'vazio', 'o-vermelho', 'vazio', 'vazio'],

L3 = ['vazio', 'vazio', 'quadrado-azul', 'vazio', 'vazio', 'vazio', 'vazio'],

L4 = ['vazio', 'vazio', 'o-vermelho', 'quadrado-azul', 'o-azul', 'vazio', 'vazio'],

L5 = ['vazio', 'quadrado-azul', 'o-azul', 'o-azul', 'quadrado-vermelho', 'vazio', 'vazio'],

L6 = ['vazio', 'vazio', 'vazio', 'vazio', 'vazio', 'vazio', 'vazio'],

Tabuleiro = [L0, L1, L2, L3, L4, L5, L6].

	0	1	2	3	4	5	6
0							
1			[*]				
2			()	[]			
3			()	()	(#)		
4		[*]	()	[#]	[]	[]	
5							
6							

Figura 8 (Um possível estado final)

Visualização do tabuleiro

`imprimeTabuleiro([Cabeca | Cauda], X, Tamanho):-`

`X < Tamanho,`

`imprimeNCol(X, 0, Tamanho),`

`imprimeLinha(Cabeca, X, 0, Tamanho),`

`imprimeSeparador(-1, Tamanho), %--desde o indice -1, para cobrir os numeros das
linhas--%`

`X1 is X + 1,`

`imprimeTabuleiro(Cauda, X1, Tamanho).`

Este será um possível predicado para imprimir o tabuleiro de jogo, e um dos outputs esperados seria o conteúdo apresentado na Figura 5.

Movimentos

Os movimentos possíveis neste jogo já foram enumerados e descritos acima na descrição do jogo, como tal estes seriam os possíveis predicados:

passa.

removeCasa(Linha, Coluna, Tabuleiro).

moveCasa(Linha, Coluna, LinhaDest, ColunaDest, Tabuleiro).

moveTorre(Linha, Coluna, LinhaDest, ColunaDest, Tabuleiro).