

Stac

Relatório Intercalar



Mestrado Integrado em Engenharia Informática e Computação

Programação em Lógica

Grupo Stac_1:

José Miguel Melo - ei12050

Rui Gomes - ei12038

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

12 de Outubro de 2014

1 O Jogo XXX

Stac é jogo de tabuleiro de dimensões 5x5, criado por Daniel Isom e publicado em 2014. Desde a sua existência já ganhou dois prêmios:

- 2 Player Print and Play Best Abstract Game - 2014
- 2 Player Print and Play Best Language-Independent Game - 2nd Place - 2014

Para ser jogado, são necessários dois jogadores e as seguintes peças:

- 2 peões, um para cada jogador
- 8 peças, 4 de cada cor para cada jogador
- 25 peças de uma única cor

Inicialização do jogo:

1. Colocar as 25 peças de cores iguais em todas as posições do tabuleiro
2. Colocar os peões em cantos opostos do tabuleiro (na diagonal)
3. 8 peças são divididas (4 para cada jogador, tendo em conta a cor)

Objectivo do jogo: Conseguir 4 torres de 3 peças. O primeiro jogador a conseguir atingir o objetivo é vencedor. Caso acabem as jogadas possíveis sem os jogadores terem completado as 4 torres, ganha o jogador com mais torres feitas.

Movimentos possíveis: Em cada jogada, o jogador pode mover o seu peão um qualquer número de casas, na horizontal e vertical, podendo passar por cima do adversário, caso este não esteja a carregar uma peça. Em momento algum, os jogadores podem ocupar a posição do adversário.

Os jogadores apenas podem carregar uma peça de cada vez, podendo-a levar para uma posição vazia ou para uma posição com 1 ou 2 peças. Uma peça não pode ser movida duas vezes consecutivas pelo mesmo jogador.

Quando um jogador coloca uma peça numa posição que já possui 2 peças, faz uma torre, colocando uma das 4 peças que lhe são atribuídas no início de jogo no topo. As peças das torres não podem ser removidas. Os jogadores só podem passar por cima de uma torre caso estejam a carregar uma peça, caso contrário são obrigados a ficar em cima da torre.

Imagens do jogo (from <http://www.boardgamegeek.com/boardgame/155943/stac>)



Fig. 1 - Início de jogo

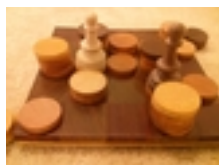


Fig. 2 - Ambos os jogadores com duas pilhas



Fig. 3 - Fim de jogo, peão branco ganhou.

2 Representação do Estado do Jogo

O tabuleiro de jogo irá ser representado utilizando uma lista de listas, de dimensões 5x5. Cada posição desta lista irá ser ocupada com os valores representativos de cada peça:

- a3 - pilha completa do jogador A
- b3 - pilha completa do jogador B
- p[num] - número de peças na casa
- a[num] - posição do jogador A. Possui [num] peças.
- b[num] - posição do jogador B. Possui [num] peças.
- vv - posição vazia

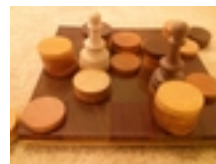
Assim, no início de jogo, o tabuleiro será representado da seguinte forma:

$[[p1, p1, p1, p1, b1], [p1, p1, p1, p1, p1], [p1, p1, p1, p1, p1], [p1, p1, p1, p1, p1], [a1, p1, p1, p1, p1]]$



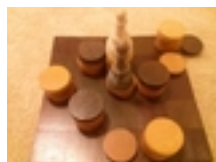
Posteriormente, numa fase intermédia do jogo, a representação poderá ser como a seguinte:

$[[vv, vv, vv, vv, p2], [vv, b3, p2, p1, p1], [a3, a1, p1, b3, vv], [vv, p2, vv, b1, p1], [p1, vv, vv, a3, vv]]$



No final de jogo, com o jogador A vencedor, uma representação possível será:

$[[vv, vv, vv, vv, a3], [vv, b3, a3, vv, p1], [a3, vv, b2, b3, vv], [vv, b3, vv, vv, vv], [vv, vv, p1, a3, vv]]$



3 Visualização do Tabuleiro

O tabuleiro irá ser visualizado em modo de texto, no terminal, sendo imprimidas para o ecrã a representação do estado de cada posição do tabuleiro. Para tal, é implementado o predicado `printBoard(B)`, que recorre aos predicados `printBoardAux([X — B])` e `printLine([X — B])` para produzir o efeito desejado.

Output produzido pelo predicado de visualização:

```
| ?- printBoard([
    [ vv, vv, vv, vv, a3 ],
    [ vv, b3, a3, vv, p1 ],
    [ a3, vv, b2, b3, vv ],
    [ vv, b3, vv, vv, vv ],
    [ vv, vv, p1, a3, vv ]
]).
vv | vv | vv | vv | a3 |
vv | b3 | a3 | vv | p1 |
a3 | vv | b2 | b3 | vv |
vv | b3 | vv | vv | vv |
vv | vv | p1 | a3 | vv |
yes
```

4 Movimentos

Cada jogador, na sua vez de jogar, pode pegar na peça que se encontra na sua casa e mover para outra casa. Por outro lado, quando completa uma pilha, o jogador pode declarar uma stack. Também há a possibilidade de propor um empate.

Assim, os cabeçalhos para estes movimentos serão os seguintes:

- `carryDisc()`
- `Move()`
- `claimStack()`

- OfferDraw()