

# Sixteen Stones

Relatório Intercalar



Mestrado Integrado em Engenharia Informática e Computação

Programação em Lógica

**Grupo Sixteen\_Stones\_3:**

Diogo Filipe Costa - ei11014

Maria Teresa Chaves - up201306842

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

10 de Outubro de 2015

# Conteúdo

<b>1</b>	<b>O Jogo Sixteen Stones</b>	<b>3</b>
1.1	Regras . . . . .	3
1.1.1	Jogadas possíveis . . . . .	3
1.1.2	<i>Push</i> . . . . .	4
1.1.3	<i>Move</i> . . . . .	5
1.1.4	<i>Sacrifice</i> . . . . .	6
<b>2</b>	<b>Representação do Estado do Jogo</b>	<b>7</b>
<b>3</b>	<b>Visualização do Tabuleiro</b>	<b>7</b>
<b>4</b>	<b>Movimentos</b>	<b>7</b>

# 1 O Jogo Sixteen Stones

Sixteen Stones é o nome de um jogo de tabuleiro abstrato <sup>1</sup> de forma quadrada, jogado normalmente num tabuleiro de 5x5, para dois jogadores. Inicialmente, cada jogador tem 8 pedras vermelhas ou azuis, colocando-as de forma alternada em qualquer uma das células livres, sendo que o jogador vermelho é o primeiro a jogar. Assim que todas as pedras estejam no tabuleiro, o jogo começa. O jogo termina quando o jogador vencedor consegue reduzir para um o número de pedras em jogo do adversário.

## 1.1 Regras

No início do jogo distribui-se 8 pedras (vermelhas ou azuis) para cada jogador. De forma alternada os jogadores devem colocá-las em qualquer uma das células livres, começando as pedras vermelhas.

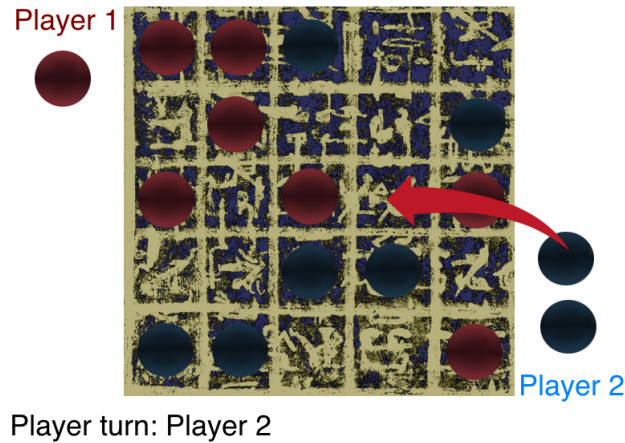


Figura 1: Preparação do tabuleiro.

Quando as pedras estiverem todas colocadas no tabuleiro, o jogador vermelho começa a jogar e alterna de turno com o jogador adversário.

### 1.1.1 Jogadas possíveis

Um jogador pode no seu respetivo turno realizar cada uma das seguintes jogadas: *Push*, *Move* e *Sacrifice*. No primeiro turno de cada jogador, deve ser feito um *push* ou um *move*, mas nunca ambos.

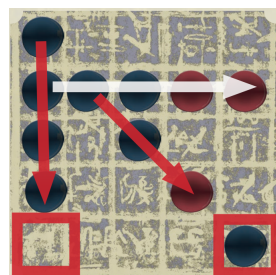


Figura 2: Primeiro turno de um jogador.

---

<sup>1</sup>Jogo abstrato - jogo de estratégia que tenta minimizar a sorte e não possui tema.

### 1.1.2 *Push*

Para realizar um *Push* é necessário que o jogador tenha mais pedras nessa linha que o adversário. Assim, se após um *Push*, a pedra do adversário é "empurrada" para fora do tabuleiro, então vai para o "banco" do respectivo jogador. As pedras que são "empurradas" apenas se movem uma célula na direção do *Push*, que pode ser realizado em qualquer direção.

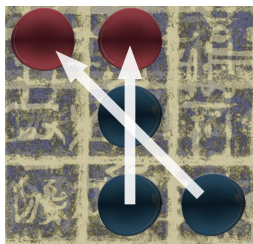


Figura 3: Direções possíveis de um *Push* e representação do "banco".

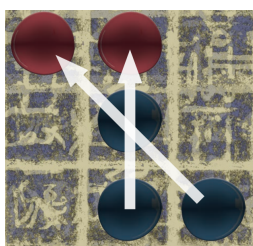


Figura 4: Exemplo de *Push* sem que a pedra adversária seja "empurrada" para fora do tabuleiro.

O número de pedras máximo (PE) que um jogador pode "empurrar" é igual:

$$PE = P - 1 \quad (\text{sendo } P \text{ o número das suas pedras nessa linha}) \quad (1)$$

Isto é, por exemplo, caso numa linha o jogador azul tenha três pedras, então pode realizar um *Push* a uma pedra ou a duas do adversário. Mas caso tenha duas apenas pode realizar um *Push* a uma pedra vermelha.

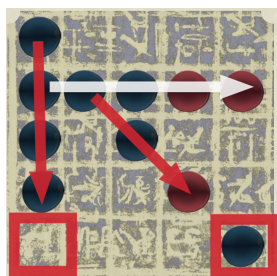


Figura 5: *Push* com três pedras.

Além das regras referidas anteriormente, um jogador não pode realizar um *Push* nas suas próprias pedras.

Por fim, para realizar um *Push* é necessário que exista uma pedra para ser empurrada. Por exemplo caso numa linha o jogador azul tenha 3 pedras, mas nessa linha o jogador vermelho não tem pedras, então é realizar um *Push* é uma jogada inválida.

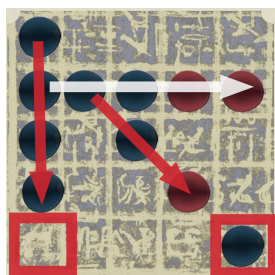


Figura 6: Push inválido por realizar um *Push* nas suas próprias pedras.

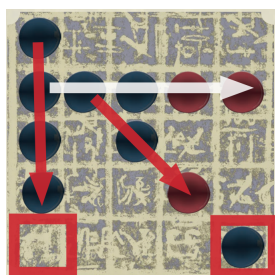


Figura 7: Push inválido por falta de pedras do adversário.

### 1.1.3 *Move*

É possível realizar um *Move* para uma célula vazia em qualquer direção. Se o jogador tentar realizar um *Move* para uma célula ocupada, então é considerado uma jogada inválida.

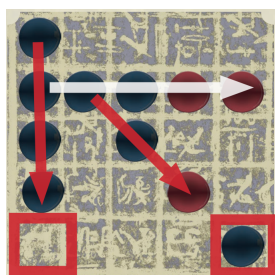


Figura 8: Exemplo de um *Move* válido (a) e um *Move* inválido (b).

Após um *Move* se alguma das pedras adversárias ficar cercada<sup>2</sup>, então esta é capturada e substituída por uma pedra do "banco" do jogador que a capturou.

---

<sup>2</sup>Uma pedra fica cercada quando numa das direções existem pedras adversárias nos dois sentidos.

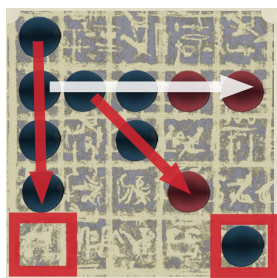


Figura 9: Exemplo de uma pedra capturada após um *Move*.

Se um jogador realizar um *Move* e a sua pedra ficar voluntariamente cercada, esta não é capturada.

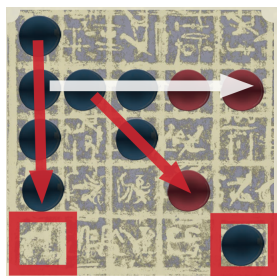


Figura 10: Exemplo de uma pedra que após um *Move* não é capturada.

#### 1.1.4 *Sacrifice*

Um jogador pode sacrificar uma pedra do seu "banco", permanentemente, para poder realizar um *Push* ou um *Move* adicional.

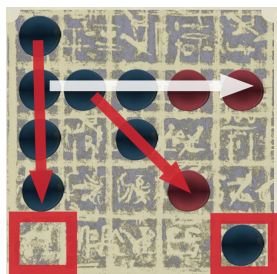


Figura 11: Jogador azul realiza um *Sacrifice* e depois um *Push*

- No primeiro turno de cada jogador, deve ser feito um *push* ou um *move*, mas nunca ambos
- Se após um jogador realizar um *Push*, uma das suas peças movidas não ocupa uma "casa" anteriormente ocupada por outra das suas peças, e esta cercar uma peça adversária, então esta é capturada
- Se um jogador empurrar uma peça adversária para uma posição de cerco, então esta é capturada
- Quando um jogador após uma captura substitui a peça capturada por uma do seu banco, esta não pode ser utilizada para capturar outra peça enquanto não for movida
- Se um jogador não tem peças no banco então este não pode capturar peças adversárias

Descrever detalhadamente o jogo, a sua história e, principalmente, as suas regras. Devem ser incluídas imagens apropriadas para explicar o funcionamento do jogo. Devem ser incluídas as fontes de informação (e.g. URLs em rodapé).

## 2 Representação do Estado do Jogo

Descrever a forma de representação do estado do tabuleiro (tipicamente uma lista de listas), com exemplificação em Prolog de posições iniciais do jogo, posições intermédias e finais, acompanhadas de imagens ilustrativas.

## 3 Visualização do Tabuleiro

Descrever a forma de visualização do tabuleiro em modo de texto e o(s) predicado(s) Prolog construídos para o efeito. Deve ser incluída pelo menos uma imagem correspondente ao output produzido pelo predicado de visualização.

## 4 Movimentos

Elencar os movimentos (tipos de jogadas) possíveis e definir os cabeçalhos dos predicados que serão utilizados (ainda não precisam de estar implementados).

## Bibliografia