

REVERSI

Relatório

Laboratórios de Informática II

Mestrado Integrado em Engenharia Informática

Grupo 1

Filipa Fernandes Guimarães Cunha e Castro, 89459

Manuel João Ferreira Moreira, 89471

Sara João Carvalho Dias, 89544

REVERSI

Introdução.....	2
Reversi – o jogo.....	3
Comandos.....	3
Sugestão e validação de jogada	4
Modos de Jogo	5
Estratégias do “bot”	6
Nível 1	6
Nível 2	6
Nível 3	7

Introdução

No âmbito do projeto proposto na unidade curricular de Laboratórios de Informática II, procedeu-se à realização de um programa que simulasse fielmente o tão conhecido jogo Reversi (ou Othello, como também é designado).

O conteúdo deste relatório aborda a projeção e explicação dos comandos e modos de jogo e, principalmente, as estratégias utilizadas para a elaboração do “bot” em cada nível de jogo, que permite ao utilizador iniciar a partida contra o computador em três níveis distintos (fácil, médio e difícil).

Reversi – o jogo

Comandos

O jogo apresenta um menu com 10 comandos disponíveis, através dos quais o jogador pode seleccionar diversas opções inerentes ao jogo, nomeadamente iniciar um jogo e o modo em que pretende jogar (manual “humano vs. humano” e automático “humano vs. bot”), o nível de jogo, efetuar a seleção específica da posição de jogada, entre outras relacionadas com a escolha da peça (“X” ou “O”), jogadas válidas, sugestões de jogada, anulação da jogada anterior e leitura e escrita do estado de jogo em ficheiros, para poder retomar a qualquer momento um jogo pendente.

Estes comandos podem ser observados mais detalhadamente na figura abaixo.

```
-> N <peca> : começar novo jogo
-> L <ficheiro> : ler um jogo do ficheiro
-> E <ficheiro> : escrever estado do jogo no ficheiro
-> J <L> <C> : jogar na posição (L,C)
-> S : mostrar no tabuleiro jogadas possíveis
-> H : receber sugestão de jogada
-> U : desfazer última jogada
-> A <peca> <nivel> : começar jogo com bot com peça <peca> e nível <nivel> (3 níveis disponíveis)
-> C <ficheiro> : começar um campeonato
-> Q : sair do jogo
```

Sugestão e validação de jogada

Ao longo da partida, poderá não ser totalmente perceptível ao jogador avaliar instantaneamente todas as jogadas disponíveis na sua vez. Para colmatar essa situação, existe a possibilidade de este ativar do comando “S”.

Adicionalmente, caso o jogador apenas necessite de uma sugestão para colocar a sua peça, pode ativar o comando “H”, que indica apenas uma jogada de entre todas as possíveis.

O código elaborado para a execução deste último comando faz com que, sempre que o jogador o ative, apareça como sugestão a posição que mais favorece o jogo do utilizador, com base no algoritmo minmax. Exemplificando:

Reversi > s	
Vez do jogador 0	
	1 2 3 4 5 6 7 8
1	- - - - - - -
2	- - - - - - -
3	- - - - 0 - - -
4	- . - 0 0 . - -
5	- - X X X X - -
6	- . 0 . . . -
7	- - - - - - -
8	- - - - - - -
Score 0: 4	
Score X: 4	

Reversi > h	
Vez do jogador 0	
	1 2 3 4 5 6 7 8
1	- - - - - - -
2	- - - - - - -
3	- - - - 0 - - -
4	- ? - 0 0 - - -
5	- - X X X X - -
6	- - - 0 - - - -
7	- - - - - - -
8	- - - - - - -
Score 0: 4	
Score X: 4	

Na primeira figura, temos uma situação de jogo com a ativação do comando “S”, em que aparece com um ponto final, todas as jogadas possíveis (posições 4-2, 4-6, 6-2, 6-3, 6-5, 6-6 e 6-7, neste caso).

Na segunda figura temos a mesma situação de jogo, só que com o comando “H” ativado, e como se pode observar, a posição sugerida (visível com um ponto de interrogação) corresponde à melhor (posição 4-2) de entre todas as possíveis.

Finalmente, para que seja possível a validação de uma determinada jogada, esta terá de cumprir de antemão os seguintes critérios:

- tratar-se de uma jogada possível e/ou sugerida;
- cumprir os limites do tabuleiro (8x8).

Modos de Jogo

Tal como referido anteriormente, a partida dispõe de dois modos de jogo principais: o modo manual, no qual o jogo é desenrolado entre dois jogadores humanos, e o modo automático, em que o utilizador joga contra o próprio computador.

Em ambos os modos de jogo, pode ser escolhida a peça de cada jogador, sendo que, no modo automático, o jogador que tiver a peça “X” é sempre aquele que inicia o jogo.

Neste mesmo modo, o jogador terá a possibilidade de testar e aprimorar as suas capacidades de jogo escolhendo iniciar a partida contra o computador (bot). Aqui, depara-se com 3 níveis de dificuldade, cujas estratégias individuais serão explanadas no tópico seguinte.

Estratégias do “bot”

Em cada nível de jogo foi adotada uma diferente estratégia para a realização do bot, que confere diferentes graus de dificuldade ao utilizador. Como já referido, existem 3 níveis de jogo: fácil, médio e difícil.

Nível 1

No primeiro nível, a estratégia aplicada é bastante simples: num jogo entre o bot e o utilizador, o computador escolhe uma posição totalmente aleatória de entre as localizações possíveis na sua vez de jogar.

Como se pode presumir, este nível não confere quase nenhum grau de dificuldade em particular ao jogador, uma vez que o bot escolhe uma posição meramente aleatória para a sua peça, sem “pensar” propriamente se está a efetuar a melhor jogada ou não, não tendo em conta quer o número de peças do adversário, quer eventuais posições estratégicas.

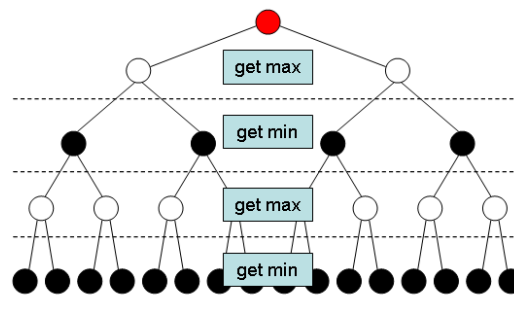
Nível 2

Aqui já é conferida uma linha de pensamento lógico ao bot. Imaginemos que o bot joga com a peça “X” e o utilizador com a peça “O”: neste caso, o bot avalia as jogadas possíveis e calcula qual delas transforma o maior número de peças “O” do adversário em peças “X”. Seguindo esta máxima, a cada jogada o computador procura sempre aquela que lhe dá mais pontuação.

No entanto, este nível é considerado médio, uma vez que a estratégia do Reversi vai muito além das jogadas por um maior número de pontos. A posição em que se joga, tendo em consideração a minimização de perda de peças nas jogadas subsequentes, influencia totalmente o desfecho do jogo, como poderemos constatar seguidamente no nível 3.

Nível 3

Este nível leva-nos a abordar sucintamente o conceito do algoritmo minmax aplicado a jogos de tabuleiro. A premissa da aplicação deste algoritmo passa essencialmente por minimizar a perda máxima possível e maximizar os ganhos.



No contexto do jogo aqui abordado, imaginemos o cenário de uma partida entre o bot (peça X) e o jogador (peça O): o algoritmo vai quantificar as jogadas possíveis do bot e avaliar qual o melhor caminho a tomar de forma a que, nas sucessivas jogadas, o jogador transforme o menor número de peças “X” do bot em peças “O”. Desta forma, no final do jogo, o jogador ficará com o menor número possível de peças “O”, logo, com menor pontuação.

Aqui o nível já pode ser considerado difícil, pois além do bot se focar na obtenção de um maior número de pontos (já presente no nível 2), são estudados e previstos todos os movimentos possíveis até 5 jogadas posteriores, aplicando o algoritmo de forma a minimizar a perda das peças do bot, e consequentemente, reduzir ao máximo as hipóteses de este obter uma pontuação baixa, maximizando o seu *score* e criando uma desvantagem ao oponente.