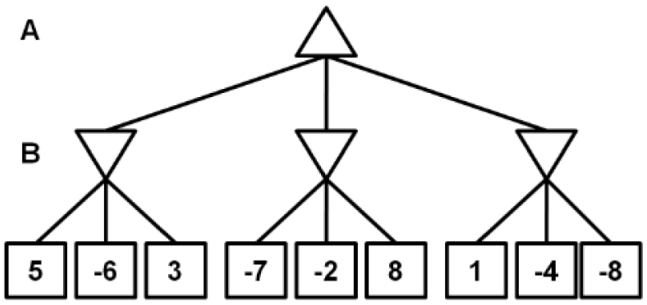
.

CEFET/RJ
Disciplina: Inteligência Artificial
Professor: Eduardo Bezerra
Lista de exercícios 02

**Créditos**: alguns itens desta lista são adaptados do material da disciplina *CS188 - Artificial Intelligence*[1], da University of Berkeley. Outros são adaptações de exercícios propostos no livro-texto da disciplina, *AIMA*[2].

1. (Busca Competitiva) Considere a seguinte árvore de um jogo de soma zero, no qual as utilidades mostradas nos nós-folha são para o primeiro jogador (A) que é um MAXimizador. Suponha que o segundo jogador (B) é um MINimizador.




   (a) Escreva nos nós internos da árvore o valor da utilidade $U_A(s)$ do jogador $A$ (isto é, o valor minimax desses nós).

   (b) Circule as arestas da árvore correspondentes às jogadas escolhidas por A e por B de acordo com o valor minimax.

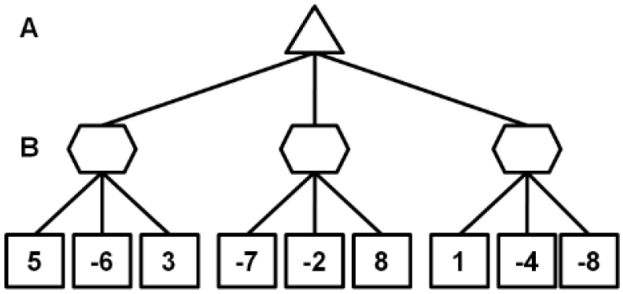
   (c) Faça um X em cima dos nós que seriam podados pela poda alfa-beta, supondo que os nós são percorridos da esquerda para a direita.

2. Ainda considerando o jogo de soma zero do exercício anterior, suponha agora que o segundo jogador (B) é um equilibrador. Um equilibrador não tenta minimizar a utilidade de A. Ao invés disso, ele quer que o resultado do jogo seja o mais equilibrado possível, isto é, o mais próximo possível de zero. A figura abaixo mostra a árvore do jogo com hexágonos indicando os nós de equilíbrio.

   (a) Escreva nos nós internos da árvore o valor da utilidade $U_A(s)$ do jogador A, supondo que B é um equilibrador.

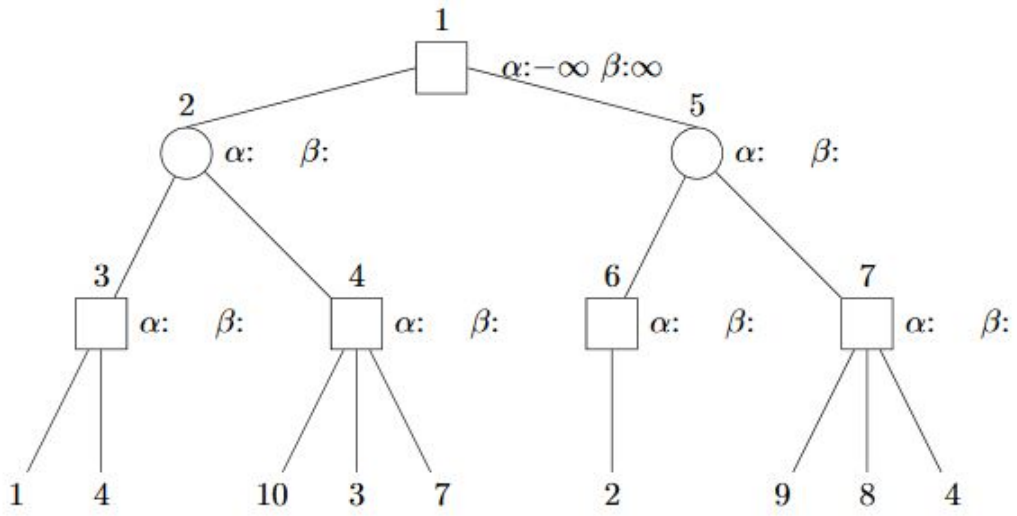
   (b) Circule as arestas da árvore correspondentes às jogadas escolhidas por A e por B.

[1] `http://ai.berkeley.edu/home.html`
[2] `http://aima.cs.berkeley.edu/`




(c) Repita esse exercício considerando agora que a utilidade de $B$ é dada por $U_B(s) = -|U_A(s)|$.

3. A árvore a seguir representa todos os resultados possíveis de um jogo hipotético de soma zero. Esta árvore foi construída da perspectiva do jogador MAX; Os nós MAX são representados por quadrados e os nós MIN são representados por meio de círculos. As folhas da árvore representam o valor do jogo para o jogador MAX. O número em cada nó indica a ordem em que são considerados pelo algotimos Minimax e $\alpha - \beta$.




(a) Calcule os valores de *back-up* de cada nó na árvore usando a estratégia Minimax, e escreva esses valores no espaço dentro de cada nó.

(b) Nota: esta pergunta tem duas partes; Leia as duas partes completamente antes de começar a responder.

i. Execute o algoritmo $\alpha - \beta$ e circule cada folha e nó que NÃO for considerado pelo algoritmo. (Suponha que as folhas são consideradas na ordem da esquerda para a direita.)

ii. Além disso, ao fazê-lo, escreva os valores de $\alpha$ e de $\beta$ que são passados como argumentos para a chamada recursiva em cada nó no espaço fornecido. (Por exemplo, o algoritmo de poda $\alpha - \beta$ é inicializado com $\alpha = -\infty$ e $\beta = -\infty$,

portanto estes são os valores passados como argumentos para a primeira chamada no nó raiz.) Nota: Se algum nó não receber valores $\alpha$ ou $\beta$ devido à poda, basta escrever X para seus valores $\alpha$ e $\beta$.

4. Neste problema, você vai investigar a relação entre as árvores expectimax e árvores minimax para jogos de soma-zero com dois jogadores. Imagine que você tem um jogo que alterna entre o jogador 1 (MAX) e o jogador 2. O jogo começa no estado $s_0$, com o jogador 1. O jogador 1 pode escolher um movimento usando ou a busca minimax, ou a busca expectimax, onde os nós do jogador 2 são nós de acaso em vez de serem nós min.
   (a) Desenhe uma árvore de jogo (pequena) na qual o nó raiz tem um valor maior se a busca **expectimax** for usada do que se a busca **minimax** for usada, ou argumente por que não seria possível.
   (b) Desenhe uma árvore de jogo (pequena) na qual o nó raiz tem um valor maior se a busca **minimax** for usada do que se busca **expectimax** é usada, ou argumente por que não seria possível.
   (c) Sob quais suposições acerca do jogador 2 o jogador 1 deve usar busca minimax em vez da busca expectimax para selecionar um movimento?
   (d) Sob quais suposições acerca do jogador 2 o jogador 1 deve usar a busca de expectimax em vez da busca minimax?
   (e) Imagine que o jogador 1 deseja agir de forma ótima (i.e., racionalmente), e o jogador 1 sabe que o jogador 2 também pretende de forma ótima. No entanto, o jogador 1 também sabe que o jogador 2 (equivocadamente) acredita que o jogador 1 está se movendo uniformemente ao acaso em vez de otimamente. Explique como o jogador 1 deve usar esse conhecimento para selecionar um movimento. Sua resposta deve ser um algoritmo preciso envolvendo uma pesquisa de árvore de jogo, e deve incluir um esboço de uma árvore de jogo apropriada com o movimento do jogador 1 na raiz. Seja claro acerca de que tipo de nós estão em cada camada (ply) e de quem é a vez em cada camada.

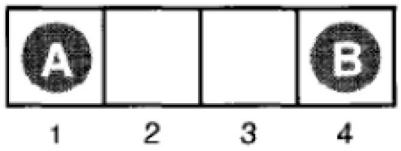
5. Definimos $X_n$ como o número de linhas, colunas ou diagonais com exatamente $n$ valores de $X$ e nenhum valor de $O$ (análogo para O). A função de utilidade atribui $+1$ a qualquer posição com $X_3 = 1$ e $-1$ a qualquer posição com $O_3 = 1$. Todas as outras posições terminais tem utilidade 0. No caso de posições não-terminais, utilizamos uma função de avaliação linear definida como

$$Aval(s) = 3X_2(s) + X_1(s) - (3O_2(s) + O_1(s))$$

   (a) Aproximadamente, quantas possibilidades de jogos existem no jogo da velha?
   (b) Mostre a árvore de jogo inteira a partir de um tabuleiro vazio até a profundidade 2, levando em conta a simetria.
   (c) Marque em sua árvore as avaliações de todas as posições na profundidade 2.
   (d) Usando o algoritmo minimax, marque em sua árvore os valores propagados, e utilize esses valores para escolher o melhor movimento inicial.

(e) Destaque os nós na profundidade 2 que não seriam avaliados se a poda alfa-beta fosse aplicada, supondo que os nós fossem gerados na ordem ótima para poda alfa-beta.

6. Considere o seguinte jogo de dois jogadores:




- O jogador A joga primeiro.
- Os dois jogadores de revezam na movimentação.
- Cada jogador deve mover sua ficha para um espaço adjacente aberto em qualquer direção.
- Se o oponente ocupar um espaço adjacente, o jogador pode saltar sobre o oponente até o próximo espaço aberto, se houver.
- O jogo termina quando um jogador chega à extremidade oposta.

(a) Desenhe a árvore de jogo completa, usando as convenções a seguir:

- Escreva cada estado com (sA, sB), onde sA e sB denotam as posições das fichas.
- Coloque cada estado terminal em um quadrado e escreva o seu valor em um círculo.
- Insira os estados repetidos em quadrados duplos. Tendo em vista que não esta clara a maneira de atribuir valores a esses estados, identifique cada um com um "?".

(b) Agora marque cada nó com seu valor minimax propagado. Explique como você tratou os valores "?" e por que.

(c) Explique por que o algoritmo minimax padrão falharia nessa árvore e faça um resumo de como corrigi-lo, baseando-se em sua resposta ao item (b).