

IA

Nome: \_\_\_\_\_

RA: \_\_\_\_\_

**Lista 03**

Professor: Vinicius Pereira

---

Esta lista contém 3 páginas e 3 questões.

---

**Algoritmo  $A$  :**

Considere a função de avaliação  $f(n) = g(n) + h(n)$ , onde

$n$  é um estado qualquer encontrado na busca

$g(n)$  é o custo de chegar até  $n$  a partir do estado inicial

$h(n)$  é a estimativa heurística do custo de ir de  $n$  até um objetivo

Se esta função de avaliação for usada com a heurística de busca pela melhor escolha, o resultado é chamado algoritmo  $A$ .

**Algoritmo  $A^*$  :**

Seja  $h^*(n)$  a função de avaliação heurística que retorna o valor exato da distância da melhor solução do vértice  $n$  até algum objetivo.

Então uma heurística  $A^*$  usa uma heurística de avaliação  $h(n) \leq h^*(n)$ .

Note que não temos como saber exatamente  $h^*(n)$ , só como estimar uma  $h(n) \leq h^*(n)$ .

Então a função de avaliação  $f(n)$  de  $A^*$  é  $f(n) = g(n) + h(n)$ , onde  $h(n) \leq h^*(n)$ .

**Admissibilidade :**

Uma Busca  $A$  com heurística de avaliação  $f(n) = g(n) + h(n)$ , tal que  $h(n) \leq h^*(n)$ , é uma heurística admissível.

Heurísticas admissíveis chegam, eventualmente, na resposta ótima.

Note que o Algoritmo  $A^*$  é admissível.

**Grau de informação :**

Para duas Heurísticas  $A^*$ ,  $h_1$  e  $h_2$ , se  $h_1(n) \leq h_2(n) \leq h^*(n)$  para todo vértice  $n$ , então se diz que a heurística  $h_2$  é mais bem informada que a heurística  $h_1$ .

1. Um quebra-cabeça de peças deslizantes consiste em um tabuleiro unidimensional com 7 espaços, três peças pretas no extremo esquerdo, três peças brancas no extremo direito, e um espaço vazio no meio, figura 1.



Figura 1: Quebra-cabeça de peças deslizantes

Uma peça pode ser movida para uma casa adjacente, este movimento tem custo 1. Uma peça pode pular uma ou duas peças para um espaço em branco, este movimento tem o custo da quantidade de peças puladas.

O objetivo é ter todas as peças brancas à esquerda de todas as peças pretas, a posição do vazio não é importante.

Defina um espaço de estado para o problema.

Use a busca da melhor escolha como heurística de busca e desenvolva **duas** heurísticas de avaliação.

Argunte se as suas heurísticas de avaliação são admissíveis e/ou monotônicas. Qual tem mais informação?

2. Discuta a admissibilidade e grau de informação para as seguintes heurísticas do 8-puzzle:  $f(n) = g(n) + h(n)$ , onde  $g(n)$  é o número de passos dados até chegar em  $n$  e  $h(n)$  é

1. O número de peças fora do lugar.
2. A soma da distância de cada peça até o seu lugar objetivo
3. Gere três instâncias do estado  $n$ . Nas três instâncias o espaço vazio está no mesmo lugar que na instância original; na primeira as três peças de cima são iguais, as outras peças com um símbolo x; na segunda com as três peças do meio (em relação à altura) são iguais, as outras peças com um símbolo x; na terceira as três peças de baixo são iguais, as outras com um símbolo x (um exemplo é visto na figura 2). O valor é a soma do menor número de passos para resolver cada um dos três problemas, deixando o espaço vazio no meio, peças com símbolo x em qualquer lugar e peças com números específicos em seus lugares específicos.

OBS: Caso a heurística não seja admissível mostre um contra exemplo. Neste caso você deve criar uma instância que não seja admissível e mostrar.

OBS2: Não use o exemplo dado para explicar a heurística, mesmo por que é possível que ele nem tenha solução.

$$h(n) = h^* + h^* + h^*$$

3	4	8	→	3	4	8	X	X	X	X	X	X
1	2			X	X		1	2		X	X	
7	6	5		X	X	X	X	X	X	7	6	5

$$25 = 11 + 13 + 1$$

Figura 2: Um exemplo de uma das heurísticas de avaliação.

3. Considere o problema em que você deve programar um robô para empilhar e desempilhar caixas em uma sala. Considere que na sala existe, pelo menos, três espaços para empilhar as caixas, a altura que a pilha alcança é ilimitada. No estado inicial já existem caixas empilhadas. Cada caixa tem uma identificação. O robô consegue empilhar ou desempilhar uma caixa por vez. O objetivo é, dado duas identificações,  $X$  e  $Y$ , Colocar a caixa  $X$  logo em cima da caixa  $Y$ .

Desenvolva uma heurística de avaliação não trivial para uma busca do algoritmo  $A$ . Discuta se a sua heurística é admissível. .