

# Inteligência Artificial

## Primeira Lista de Exercícios

### Busca

Prof. Norton Trevisan Roman

11 de março de 2019

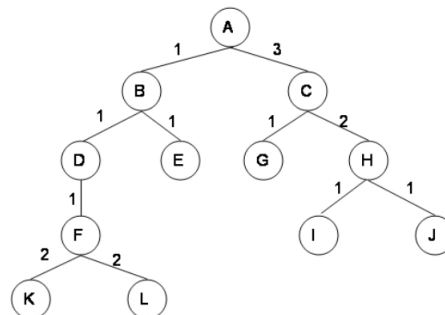
1. Um robô explorador em Marte precisa deixar o módulo de aterrissagem, coletar amostras de rochas de 3 lugares conhecidos (em qualquer ordem) e voltar ao módulo. Assuma que ele tem um módulo de navegação que pode levá-lo diretamente, e sem erros, de um lugar de interesse a outro. Assim, ele conhece as ações vá-ao-módulo-de-aterrissagem, vá-à-pedra<sub>1</sub>, vá-à-pedra<sub>2</sub> e vá-à-pedra<sub>3</sub>. Sabemos o tempo que leva para andar de um desses locais ao outro. Nosso objetivo é encontrar a seqüência de ações que executará essa tarefa no menor espaço de tempo.
  - (a) Formule esse problema como um problema de busca, definindo os possíveis estados, o estado objetivo, estado inicial, ações, o teste usado para ver se atingiu o estado final e a avaliação do custo.
  - (b) Construa o grafo de busca para o problema (faça-o dirigido)
  - (c) Que buscas seriam mais apropriadas nesse caso (note que os tempos para se locomover entre os locais não necessariamente são iguais). Justifique.
2. Suponha que dois amigos vivem em cidades distintas. A cada iteração, podemos simultaneamente mover cada amigo para uma cidade vizinha. O tempo necessário para ir da cidade  $i$  à sua vizinha  $j$  é igual à distância  $d(i, j)$  entre elas. Contudo, a cada iteração o amigo que chegar antes em uma cidade deve esperar até que o outro chegue em seu destino, para só então a próxima iteração começar. Queremos que os amigos se encontrem o quanto antes.
  - (a) Escreva uma formulação detalhada para esse problema
  - (b) Seja  $D(i, j)$  a distância em linha reta entre as cidades  $i$  e  $j$ . Quais das seguintes heurísticas é admissível?
    - i.  $D(i, j)$
    - ii.  $2D(i, j)$
    - iii.  $D(i, j)/2$
  - (c) Existe algum mapa completamente conectado para o qual não há solução?
  - (d) Existe algum mapa no qual todas as soluções exigem que um amigo visite a mesma cidade duas vezes?
3. Considere uma versão mais real do explorador em Marte. Agora, ele deve andar pela superfície, coletar algumas amostras de rochas, e voltar ao módulo de aterrissagem. Queremos construir um plano para sua exploração. Considere agora que:
  - Ele tem baterias. As baterias podem ser recarregadas se ele parar e abrir os coletores solares (suponha que é sempre dia). Uma hora de exposição solar resulta em uma unidade de carga da bateria. As baterias podem armazenar um total de 10 unidades de carga.
  - Ele pode andar. E tem um mapa com resolução de 10 metros indicando quantas unidades de carga da bateria e quanto tempo (em horas) será necessário para chegar a uma rocha aceitável em cada quadrado do mapa.

- Ele pode pegar uma rocha. Isso requer uma unidade de carga da bateria. O robô tem um mapa com resolução de 10 metros que indica o tipo de rocha esperado em cada local e o peso esperado das rochas. Assuma que somente um tipo de rocha e tamanho podem ser encontrados em cada quadrado do mapa.

O objetivo do robô é pegar 10 tipos diferentes de rochas em 3 dias, enquanto minimiza a combinação de seu peso total e a distância viajada. Você recebe um parâmetro de conversão  $\alpha$  que converte unidades de peso em unidades de distância (ou seja, carregar tanto de peso equivaleriam a andar tantos metros sem o peso). O robô começa do módulo com a bateria cheia, e deve retornar a ele. Eis uma lista de variáveis que podem ser usada para descrever o mundo do robô:

- tipos de rocha já coletados
  - localização atual do robô (quadrado no mapa)
  - localização atual do módulo (quadrado no mapa)
  - peso das rochas no local atual (quadrado no mapa)
  - custo para atravessar o local atual (quadrado no mapa)
  - tempo decorrido desde a última recarga da bateria
  - tempo decorrido desde a saída do módulo
  - data atual
  - nível atual da carga da bateria
  - capacidade total da bateria
  - distância ao módulo
  - peso total das rochas coletadas
- Use um conjunto das variáveis acima para descrever o estado do robô. Não inclua informação externa.
  - Especifique o teste para ver se o objetivo foi atingido.
  - Especifique as ações. Indique como elas modificam o estado e as pré-condições para serem usadas.
  - Especifique uma função que determine o custo de cada ação.
  - Este problema pode ser tratado como uma busca de caminho em grafo. Gostaríamos de achar uma heurística. Diga, para cada uma das heurísticas abaixo, se ela seria útil para encontrar o caminho ótimo ou, se não, o que há de errado com ela. considere  $n$  o número de rochas já coletadas:
    - $H_1$ : A soma das distâncias (no mapa), a partir do robô, até os  $10 - n$  locais mais próximos para os tipos de rocha que faltam
    - $H_2$ : O comprimento do circuito mais curto através dos  $10 - n$  locais mais próximos para os tipos de rocha que faltam
    - $H_3$ : a distância de volta ao módulo

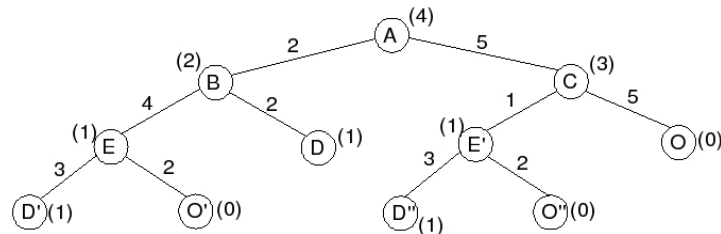
4. Considere a árvore de busca abaixo:



Assuma que o objetivo é o nó G. Quais seriam os estados expandidos (visitados) pelas seguintes buscas? Pare quando expandir G.

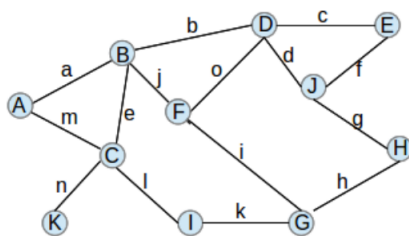
- (a) De custo uniforme, com preferência para nós à esquerda em caso de empate
- (b) Quais seriam os nós visitados pela Subida da Encosta, com a heurística de sempre se mover para a aresta de maior peso, e com preferência para nós à esquerda em caso de empate? A solução (nó G) foi encontrada? Qual foi o problema?

5. Considere a árvore de busca abaixo:



Assuma que o objetivo é qualquer nó O (O, O' ou O''). Os números junto às arestas correspondem a seus respectivos pesos, enquanto que os números entre parênteses correspondem às distâncias estimadas ao estado objetivo (representado por qualquer um dos nós O), ou seja, o valor da função heurística

- (a) Quais seriam os estados expandidos pela busca A\*, com preferência para nós à esquerda da árvore em caso de empate? Pare quando expandir um nó objetivo.
  - (b) Quais seriam os nós visitados pela Subida da Encosta, com as mesmas características da questão 4?
6. Escolha o método de busca mais eficiente, de acordo com os critérios abaixo (explique sua escolha):
- (a) Você tem um grafo com pesos nas arestas. O algoritmo deve garantir que o caminho de menor custo é encontrado.
  - (b) Você tem um grafo com pesos nas arestas e valores heurísticos consistentes em cada estado. O algoritmo deve garantir que o caminho de menor custo é encontrado.
  - (c) Você tem um grafo sem pesos nas arestas ou valores heurísticos. O algoritmo deve encontrar caminhos até o objetivo com o menor número de estados, tomando cuidado de que há necessidades de espaço.
7. Considere o grafo abaixo, onde cada estado tem o valor de *fitness* (função objetivo) conforme a tabela adjacente. Cada ação, nesse caso, nos leva de um estado a outro. As ações estão indicadas nas arestas do grafo, arestas essas que são bidirecionais (ou seja, funcionam em ambas as direções). O objetivo é executar uma sequência de ações que maximize a função objetivo. Assuma que A é o estado inicial.



Estado	<i>Fitness</i>
A	1
B	5
C	4
D	7
E	9
F	5
G	4
H	6
I	3
J	6
K	10

- (a) Usando *hill climbing*, qual será o estado final atingido?
- (b) Esse estado é um ótimo local ou global? Justifique.