

## Inteligência Artificial

Lista de Exercícios - Resolução de Problemas por Busca Desinformada – Profa. Heloisa

=====

### Recomendações

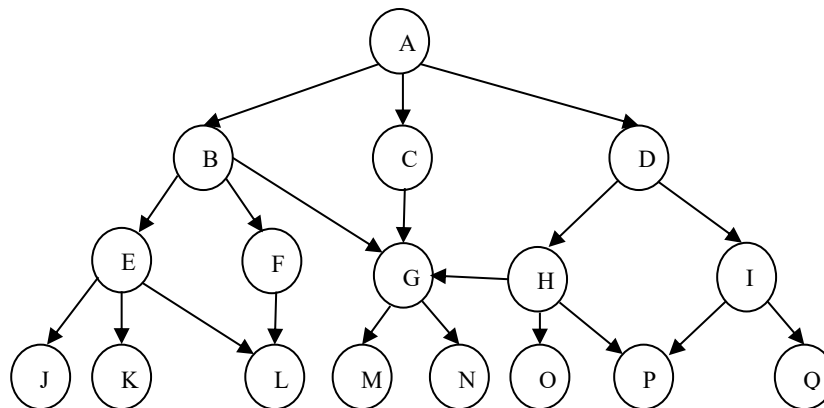
#### **Informe-se e siga as convenções de notação e figuras adotadas:**

- Em um grafo que representa um espaço de estados, um arco entre dois nós pode ou não ser direcionado. Se o arco tiver uma seta em um dos sentidos, a ação correspondente pode ser aplicada apenas no sentido da seta. Se o arco não for direcionado, a ação correspondente pode ser aplicada em qualquer um dos dois sentidos;
- A árvore de busca acompanha os passos realizados durante a busca. Portanto, na árvore de busca, são indicados apenas os movimentos que foram de fato aplicados durante a busca. Como é uma estrutura de árvore, cada nó tem apenas um pai;
- Faça tudo e somente o que foi pedido no enunciado (mostrar conteúdo das listas, desenhar árvore de busca, indicar e justificar mudanças que aconteceram nos conteúdos ou nos desenhos, mostrar o caminho da solução, responder outras perguntas).

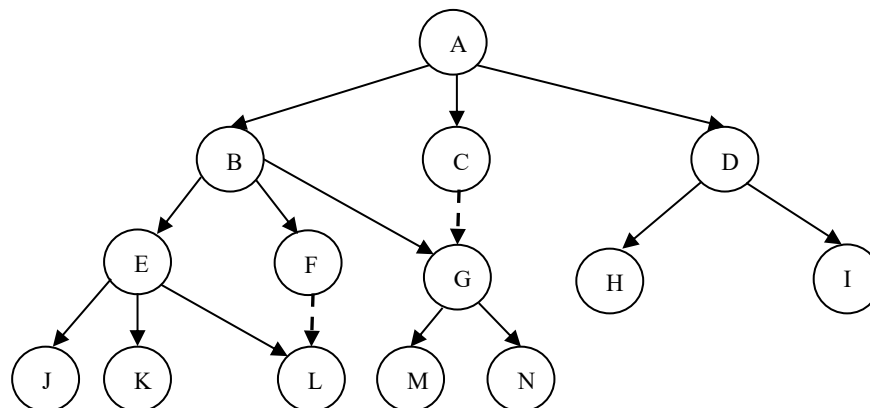
#### **Para resolver os exercícios dessa lista você precisa:**

- saber aplicar corretamente os algoritmos, com atenção para as diferenças entre eles, principalmente quanto a forma de tratar a lista OPEN e o momento de testar se o nó atual é um nó objetivo;
- o que fazer quando um nó não gera sucessores;
- o que fazer quando um nó gerado já aparecia na lista OPEN ou CLOSED, inclusive com relação ao desenho da árvore de busca;
- lembrar que quando a ordem dos movimentos está definida, ela deve ser seguida inclusive no desenho da árvore de busca;
- lembrar que movimentos que retornam a estados que já estão nas listas OPEN ou CLOSED (já aparecem na árvore de busca) não são representados na árvore.
- seguir as indicações de representação dos conteúdos das listas OPEN e CLOSED (com ou sem custo) e da forma de desenhar a árvore de busca.

- 1) Considere o grafo a seguir, representando o espaço de estados para um problema. Sendo o nó A inicial, e o nó N o objetivo, utilize o algoritmo de **busca em largura** para encontrar a solução desse problema. Apresente o conteúdo das listas OPEN e CLOSED a cada passo e construa a árvore de busca. Indique as ações tomadas quando um nó gerado já estiver em uma das duas listas e diga qual é o caminho da solução



**Solução:**



OPEN	CLOSED	Nó selecionado
A		A
B C D	A	B
C D E F G	A B	C
D E F G G foi gerado novamente por C, mas já estava em OPEN. Permanece onde estava, não é inserido novamente.	A B C	D
E F G H I	A B C D	E
F G H I J K L	A B C D E	F
G H I J K L L foi gerado novamente por F, mas já estava em OPEN. Permanece onde estava, não é inserido novamente.	B C D E F	G
H I J K L M N	A B C D E F G	N É OBJETIVO - PARE

**SOLUÇÃO: A – B – G – N**

- 2) Considere o mesmo grafo de espaço de estados do exercício anterior. Sendo o nó A inicial, e o nó N o objetivo, utilize o algoritmo de **busca em profundidade** para encontrar a solução desse problema. Apresente o conteúdo das listas OPEN e CLOSED a cada passo e construa a árvore de busca. Indique as ações tomadas quando um nó gerado já estiver em uma das duas listas e diga qual é o caminho da solução.
- 3) Construa a representação dos estados, o estado inicial, final e os operadores para o problema dos baldes de água, definido como:

Existem dois baldes de água, o primeiro com capacidade para 5 litros e o segundo com capacidade para 3 litros. É possível encher qualquer um dos baldes totalmente de uma torneira ou esvaziá-los totalmente independente da quantidade de água que eles tenham, e despejar água de um para o outro, até que o que recebe a água esteja totalmente cheio ou até que aquele do qual se despeja a água esteja totalmente vazio. Supondo que os baldes estão inicialmente cheios, o objetivo é deixar 4 litros de água no balde maior e qualquer quantidade de água no balde menor.

**Solução:**

Representação: (M,m) em que M é a quantidade de água no balde maior e m é a quantidade de água no balde menor

Estado inicial: (5,3) (O balde maior tem 5 litros e o balde menor tem 3 litros)

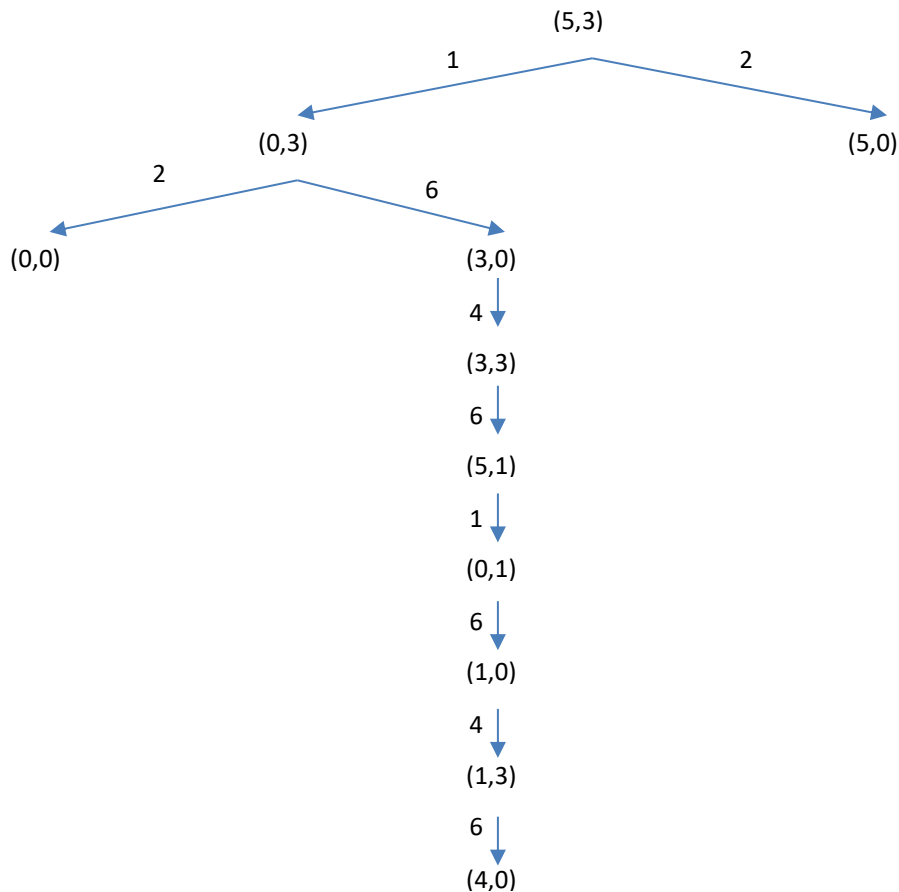
Estado final: (4,x) (O balde maior tem 4 litros e o balde menor tem qualquer quantidade)

Operadores:

- 1) Esvaziar balde maior
- 2) Esvaziar balde menor
- 3) Encher balde maior
- 4) Encher balde menor
- 5) Despejar do balde maior para o balde menor
- 6) Despejar do balde menor para o balde maior

- 4) Considere o problema dos baldes de água e assuma que os operadores são aplicados na ordem: 1) esvaziar balde maior; 2) esvaziar balde menor; 3) encher balde maior; 4) encher balde menor; 5) despejar do balde maior para o menor; 6) despejar do balde menor para o maior. Aplique o algoritmo de **busca em largura** para encontrar a solução para esse problema. Apresente o conteúdo das listas OPEN e CLOSED a cada passo e construa a árvore de busca. Indique as ações tomadas quando um nó gerado já estiver em uma das duas listas e diga qual é o caminho da solução.
- 5) Considere o problema dos baldes de água e assuma que os operadores são aplicados na ordem: 1) esvaziar balde maior; 2) esvaziar balde menor; 3) encher balde maior; 4) encher balde menor; 5) despejar do balde maior para o menor; 6) despejar do balde menor para o maior. Aplique o algoritmo de **busca em profundidade** para encontrar a solução para esse problema. Apresente o conteúdo das listas OPEN e CLOSED a cada passo e construa a árvore de busca. Indique as ações tomadas quando um nó gerado já estiver em uma das duas listas e diga qual é o caminho da solução.

### Solução:



OPEN	CLOSED	X
(5,3)		(5,3)
(0,3) (5,0)	(5,3)	(0,3)
(0,0) (3,0) (5,0) (5,3) é gerado novamente por (0,3) mas já está em CLOSED. Permanece em CLOSED e não é inserido novamente em OPEN	(5,3) (0,3)	(0,0)
(3,0) (5,0) (0,0) gera (5,0) que já está em OPEN e (0,3) que já está em CLOSED. Esses nós ficam onde estão e não são inseridos novamente em OPEN. (0,0) não gera estados novos	(5,3) (0,3) (0,0)	(3,0)
(3,3) (5,0) (3,0) gera (0,0) e (0,3) que já estão em CLOSED e (5,0) que está em OPEN. Esses nós ficam onde estão e não são inseridos novamente em OPEN.	(5,3) (0,3) (0,0)(3,0)	(3,3)
(5,1) (5,0) (3,3) gera (0,3), (3,0) e (5,3) que já estão em CLOSED. Esses nós ficam onde estão e não são inseridos novamente em OPEN	(5,3) (0,3) (0,0)(3,0)(3,3)	(5,1)
(0,1) (5,0) (5,1) gera (5,0), (5,3) e (3,3) que já estão em CLOSED. Esses nós ficam onde estão e não são inseridos novamente em OPEN	(5,3) (0,3) (0,0)(3,0)(3,3)(5,1)	(0,1)
(1,0) (5,0)	(5,3) (0,3) (0,0)(3,0)(3,3)(5,1)(0,1)	
(continuar...)		

Caminho da solução: (5,3)-(0,3)-(3,0)-(3,3)-(5,1)-(0,1)-(1,0)-(1,3)-(4,0)

Movimentos da solução: Esvaziar maior, Despejar do menor para maior, Encher menor, despejar do menor para maior, Esvaziar maior, Despejar do menor para maior, Encher menor, Despejar do menor para maior.

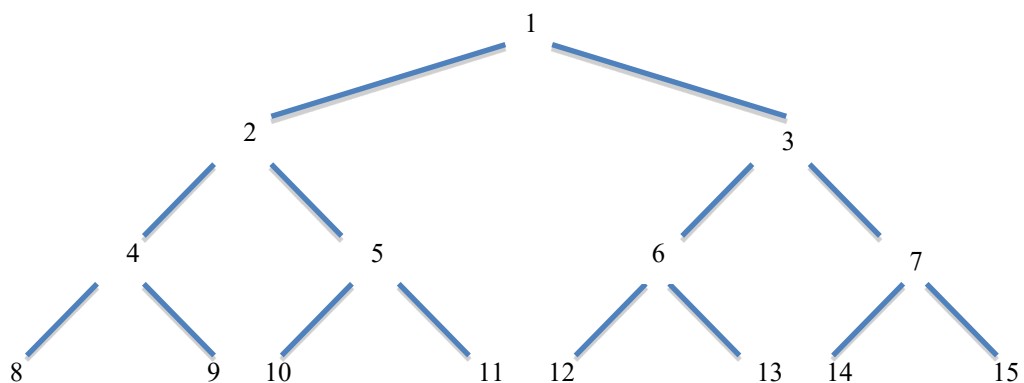
- 6) Considere outra versão para o problema dos baldes de água em que a capacidade dos baldes maior e menor é 7 e 5 litros, respectivamente. Considerando que os baldes estão inicialmente cheios e que o objetivo é deixar 4 litros de água no balde maior e qualquer quantidade no balde menor, aplique os algoritmos de **busca em profundidade** e **busca em profundidade fixa até nível 7**. Para cada um dos algoritmos, construa a árvore de busca e mostre a sequência de movimentos que determina a solução. Compare e comente as duas soluções encontradas quanto ao nível em que estava o objetivo encontrado, o tamanho da solução (número de movimentos) e o número de nós gerados. Não é necessário mostrar o conteúdo das listas OPEN e CLOSED.

**Solução parcial:**

Ao comparar as duas soluções encontradas, verifica-se que, na busca em profundidade, o objetivo (4,0) é atingido no nível 16, são gerados 19 nós e a solução tem 16 movimentos, enquanto na busca em profundidade fixa até nível 7 o objetivo (4,5) é atingido no nível 6, são gerados 15 nós e a solução tem tamanho 6.

- 7) Considere um espaço de estados onde o estado inicial é o número 1 e cada estado  $k$  tem dois sucessores: os números  $2k$  e  $2k+1$ . Desenhe a parte do espaço de estados que vai do estado 1 ao 15. Supondo que o estado objetivo é 11, liste a ordem em que os estados são gerados pelo algoritmo de **busca em largura**; liste a ordem em que os estados são gerados pelo algoritmo de **busca em profundidade iterativa** em cada uma das iterações do algoritmo.

**Solução:**



Sequência de nós gerados pela busca em largura:

1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11

Sequência de nós gerados pela busca em profundidade iterativa:

Iteração 1: 1

Iteração 2: 1, 2, 3

Iteração 3: 1, 2, 3, 4, 5, 6, 7

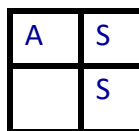
Iteração 4: 1, 2, 3, 4, 5, 8, 9, 10, 11

- 8) O problema do mundo do aspirador de pó pode ser formulado como descrito abaixo. Construa a árvore de busca para o algoritmo de **busca em largura** e **busca em profundidade limitada** com **lim=3**. Enumere os nós da árvore de busca acordo com a sequência em que foram expandidos. Indique as ações tomadas quando um nó gerado já estiver em uma das listas OPEN ou CLOSED. Mostre a sequência de movimentos que determina a solução. Compare e comente as duas soluções encontradas quanto ao nível em que estava o objetivo encontrado, o tamanho da solução (número de movimentos) e o número de nós gerados. Não é necessário descrever os conteúdos das listas OPEN e CLOSED passo a passo.

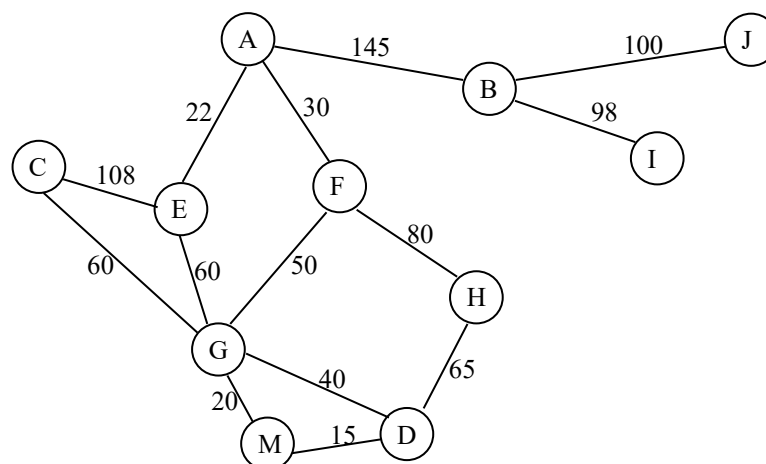
**Problema do Mundo do Aspirador de Pó:** Um cenário é representado por uma grade de 2X2, sendo que cada quadrado pode ter ou não sujeira. Um aspirador de pó pode se mover nesse cenário, com os seguintes movimentos: aspirar a sujeira; andar para a esquerda; andar para a direita; andar para cima; andar para baixo. Os operadores devem ser aplicados na ordem: 1) aspirar; 2) mover à direita; 3) mover para baixo; 4) mover à esquerda; 5) mover para cima. Encontre a sequência de movimentos para chegar a um estado onde todos os quadrados estão limpos, supondo que no início, e o aspirador de pó está no quadrado superior esquerdo e que os dois quadrados da direita estão sujos.

**Sugestão:**

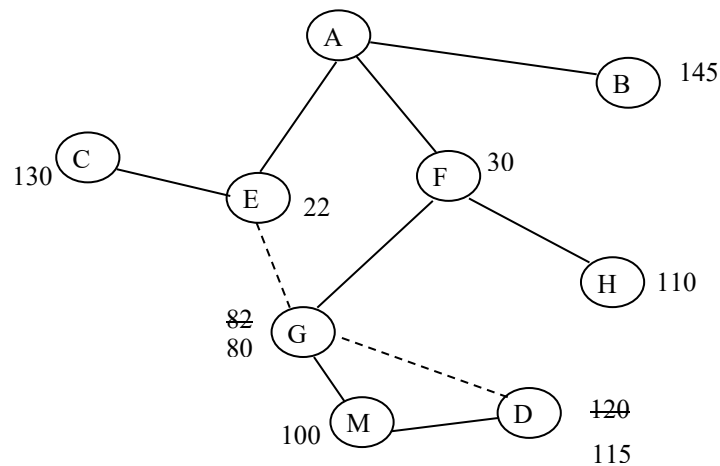
Neste exercício não é necessário mostrar os conteúdos das listas OPEN e CLOSED, portanto os estados podem ser representados com a figura do cenário. A letra A representa o aspirador e a letra S representa que o quadrado em que aparece, tem sujeira.



- 9) A figura dada a seguir representa um conjunto de cidades ligadas por estradas com suas respectivas quilometragens. Encontre o melhor caminho que vai da cidade A até a cidade D, aplicando o **algoritmo de busca de custo uniforme**. Considerar a quilometragem de cada trecho como o custo do movimento de ir de uma cidade a outra. Construa a árvore de busca e mostre o conteúdo das listas OPEN e CLOSED a cada passo indicando claramente o que foi feito nos casos em que um nó que já estava em uma das listas é gerado novamente. Apresente a sequência de nós expandidos e o caminho da solução



Na árvore de busca, os números ao lado dos estados representam a função de custo dos estados. Quando o valor é alterado, o valor antigo está riscado.



OPEN	CLOSED	Nó selecionado
A		A
E-22, F-30, B-145	A	E
F-30, G-82, C-130, B-145	A, E	F(gera novamente G-80; valor menor, G fica em OPEN com o valor novo)
G-80, H-110, C-130, B-145	A, E, F	G (gerado por F, gera novamente E-140, e C-140; E já estava em CLOSED, é descartado; valor de C maior, C fica em OPEN com o valor antigo)
M-100, H-110, D-120, C-130, B-145	A, E, F, G	M (gera novamente D-115; valor menor, D fica em OPEN com valor novo)
H-110, D-115, C-130, B-145	A, E, F, B, G	H(gera novamente D-175; valor maior, D fica em OPEN com valor antigo)
D-115, C-130, B-145	A, E, F, B, G, M	D objetivo!

Caminho da solução: A – F – G – M – D