Universidade Federal de Viçosa Campus Rio Paranaíba
SIN 323 - Inteligência Artificial
Profa. Dra. Larissa Ferreira Rodrigues Moreira
Período: 2024-2

Lista 2

Data de Entrega: 23/10/2024 | 23h59min | PVANet Moodle

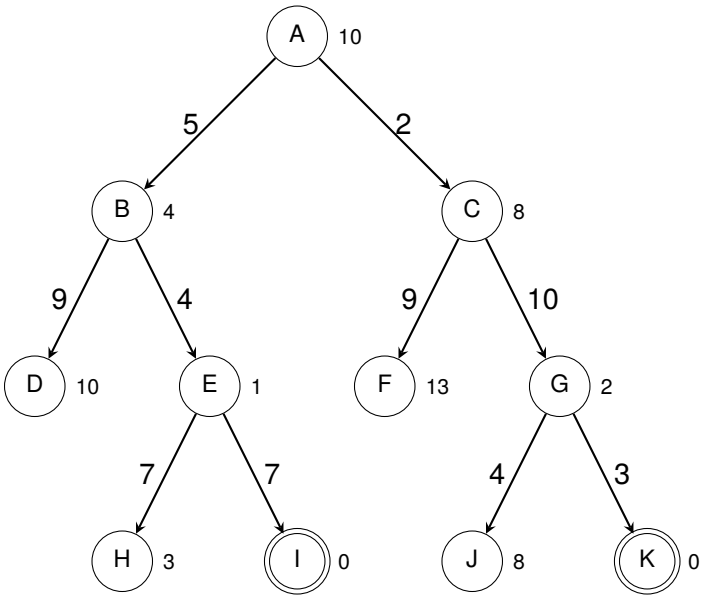
INSTRUÇÕES

- Para compreender os conceitos necessários para a resolução dos exercícios, leia os Capítulos 3 e 5 do livro "Inteligência Artificial - Uma Abordagem Moderna" de Russell e Norvig.
- A resolução da Lista (exercícios teóricos e os códigos para as questões que envolvem implementação) deve ser entregue em um único arquivo compactado (.zip) com o nome e matrícula: Nome_Matricula.zip.
  Ex.: Fulano_1234.zip
- Certifique-se de incluir comentários nos códigos para explicar a lógica implementada.

Busca

1. Qual é a diferença entre os algoritmos de busca: gulosa, A*, em largura e em profundidade?

2. Considere o espaço de busca a seguir. Cada nó é rotulado por uma letra. Cada nó objetivo é representado por um círculo duplo. Existe uma heurística estimada para cada dado nó (indicada por um valor ao lado do nó). Arcos representam os operadores e seus custos associados.



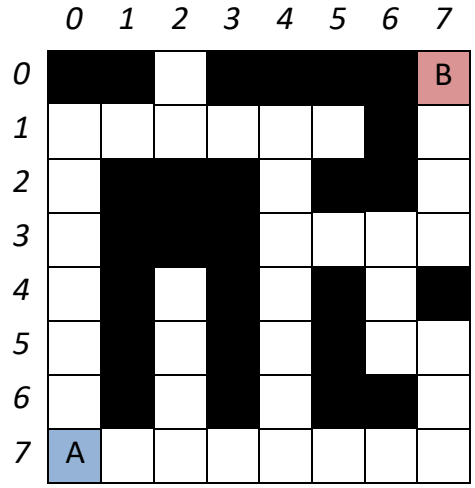
Para cada um dos algoritmos a seguir, liste os nós visitados na ordem em que eles são examinados, começando pelo nó A. No caso de escolhas equivalentes entre diferentes nós, prefira o nó mais próximo da raiz, seguido pelo nó mais à esquerda na árvore.

(a) Algoritmo de Busca Gulosa

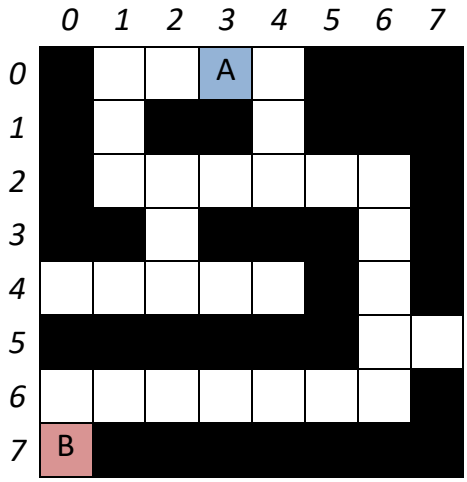
(b) Algoritmo de Busca A*

3. O que significa dizer que uma heurística h_1 domina uma heurística h_2? O que isto quer dizer em termos de eficiência de uma busca A* usando h_1 e h_2?

Considere os seguintes labirintos para resolver as questões 4 e 5, considerando A como origem e B como destino.



Labirinto 1



Labirinto 2

4. Preencha o caminho percorrido para resolver os labirintos usando **busca gulosa**. Considere a distância de Manhattan.

   (a) Qual o número de passos percorridos pelo algoritmo?

   (b) A solução encontrada pelo algoritmo é ótima? Discorra sobre.

   (c) Compare com as soluções obtidas com os métodos de largura e profundidade aplicados na Lista 1.

5. Preencha o caminho percorrido para resolver os labirintos usando o **algoritmo A***. Considere a distância de Manhattan como heurística 1 e o número de passos como heurística 2.

   (a) Qual o número de passos percorridos pelo algoritmo?

   (b) A solução encontrada pelo algoritmo é ótima? Discorra sobre.

   (c) Compare com as soluções obtidas com os métodos de largura, profundidade (aplicados na Lista 1) e busca gulosa (aplicado na questão anterior).

6. Implemente, utilizando a linguagem de programação de sua preferência, o algoritmo A* para encontrar o caminho mais curto em um grafo ponderado. O grafo representa um labirinto, onde cada nó é uma sala e cada aresta (com um peso) representa o custo de mover de uma sala para outra. Além disso, você deve usar a heurística da distância de Manhattan para estimar o custo de cada nó até o objetivo.

   Considere o seguinte grafo e as conexões:

   - O nó A está conectado a B (custo 1) e C (custo 4).
   - O nó B está conectado a D (custo 2).
   - O nó C está conectado a D (custo 1) e E (custo 5).
   - O nó D está conectado a F (custo 3).
   - O nó E está conectado a F (custo 2).

   O objetivo é encontrar o caminho mais curto de A até F.

(a) Implemente uma função `a_star(graph, start, goal, heuristic)` que receba o grafo, o nó inicial `start`, o nó objetivo `goal` e uma função `heuristic` para calcular a heurística. A função deve retornar o caminho mais curto entre `start` e `goal`.

(b) Use a distância de Manhattan como heurística, assumindo as seguintes coordenadas para os nós:

- A: (0, 0)
- B: (1, 0)
- C: (0, 1)
- D: (1, 1)
- E: (0, 2)
- F: (1, 2)

(c) Execute a função `a_star` para encontrar o caminho de A até F e imprima o caminho e o custo total.

Dica

A distância de Manhattan entre dois pontos (x_1, y_1) e (x_2, y_2) é dada por:

$$d(x_1, y_1, x_2, y_2) = |x_1 - x_2| + |y_1 - y_2|$$

Na busca A*, para cada nó, é necessário calcular o custo real de chegar até ele (g) e a estimativa de custo até o objetivo (h), sendo a função de avaliação $f(n) = g(n) + h(n)$.

Busca Competitiva

Considere os seguintes estados em uma partida de jogo da velha.

X		O
	O	O
X		X

Jogo A

X		O
	X	X
X	O	O

Jogo B

X	O	X
		O
	O	X

Jogo C

X	O	
O	X	
		X

Jogo D

7. Considere o algoritmo Minimax. Quais as saídas das seguintes funções considerando os estados A, B, C e D dos jogos-da-velha. E = {A, B}.

(a) Jogador(E)

(b) Ações(E)

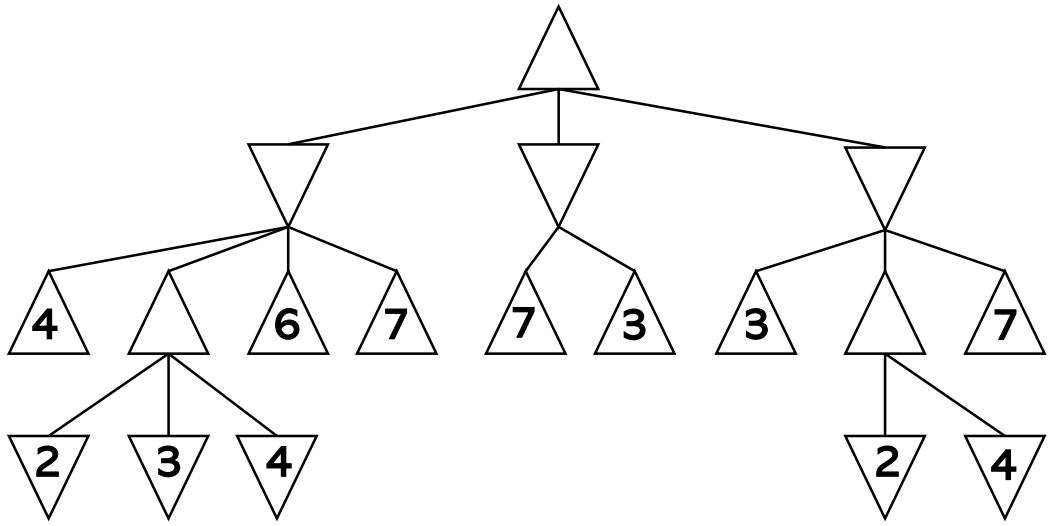
(c) Resultado(E, *)
*Considere um dos elementos resultantes de Ações(E)

(d) Terminal(E)

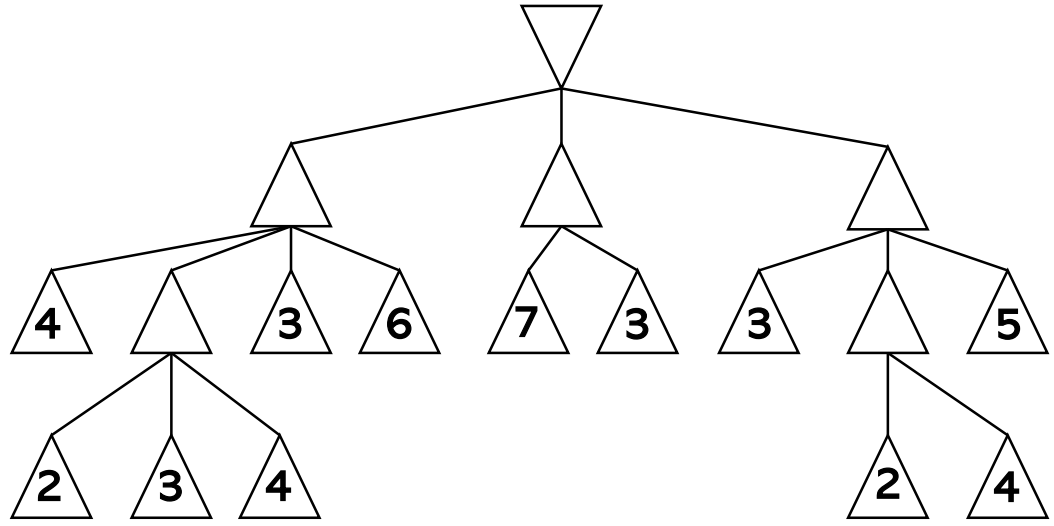
(e) Utilidade(E)
(Obs.: apenas se E é terminal)

8. Considere o algoritmo Minimax. Monte a árvore de todas as possíveis jogadas a partir dos estados E = {A, B, C, D}. Analise as melhores (e piores) decisões que cada jogador (maximizador e minimizador) podem tomar.

9. Considere as seguintes árvores com os valores finais de cada caminho estabelecidos. Preencha os valores ótimos para os maximizadores e minimizadores, de acordo com o algoritmo Minimax.



(a)



(b)