Lista 08 – Felipe Campolina

1)

**1.1** A Nós explorados: A B C D E F G H I J K Solução: [A B E I] e [A C G K] Não utiliza heurística.

**1.2** B Nós explorados: A B D E H I C F G J K Solução: [A B E I] e [A C G K] Não utiliza heurística.

**1.3** C Nós explorados: A C B E F G D K H I J Solução: [A B E I] e [A C G K] Não utiliza heurística.

**1.4** D Nós explorados: A B E I C G K Solução: [A B E I] e [A C G K] A heurística é apropriada, pois a função que mapeia o nó para uma determinada heurística, *h*(*x*), produz valores menores ou iguais à distância real do nó atual para o nó alvo.

**1.5** E Nós explorados: A B E C G K I Solução: [A B E I] e [A C G K] A heurística é apropriada, pois a função que mapeia o nó para uma determinada heurística, *h*(*x*), produz valores menores ou iguais à distância real do nó atual para o nó alvo.

2)

**2.1** A heurística que emprega a distância de Manhattan no quebra-cabeça de 8 peças é admissível, pois ao calcular a distância entre cada bloco do quebra-cabeça e sua posição final na solução, sempre obtemos um valor menor ou igual à distância real necessária para posicionar a peça corretamente. Além disso, podemos garantir que as peças mais próximas do estado final sempre terão uma heurística menor do que aquelas que estão mais distantes do resultado final.

**2.2** Uma heurística alternativa, embora impraticável, seria calcular a distância exata para o resultado final e então aplicar essa distância como a heurística do estado. Infelizmente, embora essa heurística seja admissível, pois atende aos mesmos requisitos satisfeitos pela distância de Manhattan, ela não é viável de ser utilizada, uma vez que resolver o quebra-cabeça seria necessário para obtê-la. Outra heurística possível seria calcular a distância euclidiana de cada peça até sua posição final no quebra-cabeça resolvido e somar essas distâncias para obter a heurística de cada estado. Pelos mesmos motivos apresentados na alternativa anterior, essa heurística é aceitável e possível de ser implementada.

3) B

4) A

5) E

6) A

7) B

8)B

9) O algoritmo de custo uniforme visa minimizar exclusivamente a função de custo *g*(*n*). Quando *w*=1, a função faz com que o algoritmo execute uma busca semelhante àquela do algoritmo A\*, considerando tanto)*g*(*n*) quanto)*h*(*n*) para determinar a prioridade. Finalmente, no caso de w=2, a busca será puramente uma busca gulosa, tentando minimizar apenas o valor de *h*(*n*).

10)   
10.1 A Incorreta. A busca em largura encontra a solução ótima apenas quando todos os custos são iguais.

10.2 B Incorreta. A estratégia de busca heurística pode não encontrar o melhor caminho possível se a heurística estiver incorreta.

10.3 C Correta. Como a busca de custo uniforme utiliza uma fila de prioridade baseada no mapeamento fornecido por *g*(*n*), os valores de *g*(*n*) são minimizados durante os caminhamentos.

11)A

12) MAX = 3

13) Haverá um corte na divisão <=2

14) Diagrama

Descrição gerada automaticamente