Name : ABID ALI
Student_No : 2019380141

Assignment 4

3.14 Which of the following are true and which are false? Explain your answers.

a. Depth-first search always expands at least as many nodes as A∗ search with an admissible heuristic.
Answer:
False: a lucky DFS might expand exactly d nodes to reach the goal. A∗ largely dominates any graph-search algorithm that is *guaranteed to find optimal solutions*.

b. h(n) = 0 is an admissible heuristic for the 8-puzzle.
Answer:
True: h(n) = 0 is always an admissible heuristic, since costs are nonnegative.

c. A∗ is of no use in robotics because percepts, states, and actions are continuous.
Answer:
True: A* search is often used in robotics; the space can be discretized or skeletonized.

d. Breadth-first search is complete even if zero step costs are allowed.
Answer:
True: depth of the solution matters for breadth-first search, not cost.

e. Assume that a rook can move on a chessboard any number of squares in a straight line, vertically or horizontally, but cannot jump over other pieces. Manhattan distance is an admissible heuristic for the problem of moving the rook from square A to square B in the smallest number of moves.
Answer:
False: a rook can move across the board in move one, although the Manhattan distance from start to finish is 8.

3.21 Prove each of the following statements, or give a counterexample:
a. Breadth-first search is a special case of uniform-cost search.
Answer:
When all step costs are equal, g(n) ∝ depth(n), so uniform-cost search reproduces breadth-first search.

b. Depth-first search is a special case of best-first tree search.
Answer:
Breadth-first search is best-first search with f(n) = depth(n); depth-first search is best-first search with f(n)= −depth(n); uniform-cost search is best-first search with f(n) = g(n).

c. Uniform-cost search is a special case of A∗ search.
Answer:
Uniform-cost search is A∗ search with h(n) = 0.

3.23 Trace the operation of A∗ search applied to the problem of getting to Bucharest from Lugoj using the straight-line distance heuristic. That is, show the sequence of nodes that the algorithm will consider and the f, g, and h score for each node.

Answer:

The sequence of queues is as follows:

L[0+244=244]

M[70+241=311], T[111+329=440]

L[140+244=384], D[145+242=387], T[111+329=440]

D[145+242=387], T[111+329=440], M[210+241=451], T[251+329=580]

C[265+160=425], T[111+329=440], M[210+241=451], M[220+241=461], T[251+329=580]

T[111+329=440], M[210+241=451], M[220+241=461], P[403+100=503], T[251+329=580], R[411+193=604], D[385+242=627]

M[210+241=451], M[220+241=461], L[222+244=466], P[403+100=503], T[251+329=580], A[229+366=595],R[411+193=604], D[385+242=627]

M[220+241=461], L[222+244=466], P[403+100=503], L[280+244=524], D[285+242=527], T[251+329=580], A[229+366=595], R[411+193=604], D[385+242=627]

L[222+244=466], P[403+100=503], L[280+244=524], D[285+242=527], L[290+244=534], D[295+242=537], T[251+329=580], A[229+366=595], R[411+193=604], D[385+242=627]

P[403+100=503], L[280+244=524], D[285+242=527], M[292+241=533], L[290+244=534], D[295+242=537], T[251+329=580], A[229+366=595], R[411+193=604], D[385+242=627], T[333+329=662]

B[504+0=504], L[280+244=524], D[285+242=527], M[292+241=533], L[290+244=534], D[295+242=537], T[251+329=580],

A[229+366=595], R[411+193=604], D[385+242=627], T[333+329=662], R[500+193=693], C[541+160=701]

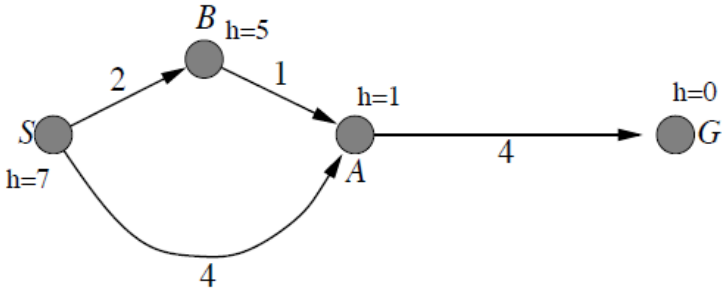

Figure S3.2 A graph with an inconsistent heuristic on which GRAPH-SEARCH fails to return the optimal solution. The successors of S are A with $f = 5$ and B with $f = 7$. A is expanded first, so the path via B will be discarded because A will already be in the closed list.