CS 1571: Homework 2

Search (Chapter 3 and Chapter 4)

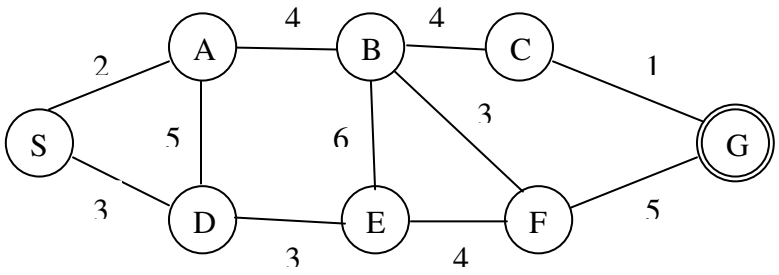
Answer key:

1. More Blind Search (30 pts)

i) (15 pts)
Depth Limited Search (limit = 4): 1, 2, 4, 8, 16, 17, 9, 18, 19, 5, 10, 20, 21, 11 or (1, 3, 7, 15, 31, 30, 14, 29, 28, 6, 13, 27, 26, 12, 25, 24, 2, 5, 11. if you visit the right child first)

ii) (15 pts)
Iterative Deepening Search:
Iteration 0: 1;
Iteration 1: 1, 2, 3; or (1, 3, 2)
Iteration 2: 1, 2, 4, 5, 3, 6, 7; or (1, 3, 7, 6, 2, 5, 4)
Iteration 3: 1, 2, 4, 8, 9, 5, 10, 11. or (1, 3, 7, 15, 14, 6, 13, 12, 2, 5, 11)

2. Greedy Search (35pts)



i) (15 pts)
Each line is a list of nodes generated due to expanding the leftmost node. The lines are listed in the order of node expansion:
S: A(7), D(5)
D: S(10), A(7), E(4)
E: D(5), B(3), F(2)
F: E(4), B(3), G(0)
G: Goal reached

ii) (5 pts)
Solution path: S, D, E, F, G

iii) (5pts)
It has cost 15. It is not optimal.

iv) A* (10 pts)
S: A(9), D(8)
D: S(16), A(15), E(10) (expanded from D(8))
A: S(14), D(12), B(9) (expanded from A(9))
B: A(17), E(16), F(11), C(11) (expanded from B(9))
E: B(15), D(14), F(12)(expanded from E(10))
C: B(17), G(11) (expanded from C(11))
G: Goal reached

Solution path: S, A, B, C, G. (optimal)

3. A Search (35pts)*

i) (15 pts)
The given heuristic is admissible because those white tiles would need to move to the left of the leftmost black tile to have a goal state. The cost of doing this at least is the #white tiles for the black tile to hop over.

(b) (20 pts)

From the initial configuration
(1) |B|B|B|W|W|E|W| (switch E and the rightmost W), f(n) = g(n) + h(n) = 1 + 3 = 4;
(2) |B|B|B|W|E|W|W| (The second rightmost W hops over into the empty cell), f(n) = 1 + 3 = 4;
(3) |B|B|B|E|W|W|W| (The leftmost W hops over into the empty cell), f(n) = 2 + 3 = 5;

From node (1)
(4) |B|B|B|W|E|W|W| (switch E and the second rightmost W), f(n) = g(n) + h(n) = 2 + 3 = 5; (same as node (2))
(5) |B|B|B|W|W|W|E| (switch E and the rightmost W), f(n) = g(n) + h(n) = 2 + 3 = 5; (same as the initial node)
(6) |B|B|B|E|W|W|W| (The leftmost W hops over into the empty cell), f(n) = 2 + 3 = 5; (same as the node (3))
(7) |B|B|E|W|W|B|W| (The rightmost B hops over into the empty cell), f(n) = 3 + 3 = 6;

From node (2)
(8) |B|B|B|E|W|W|W| (switch E and the leftmost W), f(n) = g(n) + h(n) = 2 + 3 = 5; (same as node (3))
(9) |B|B|B|W|W|E|W| (switch E and the second rightmost W), f(n) = g(n) + h(n) = 2 + 3 = 5; (same as node (1))
(10) |B|B|E|W|B|W|W| (The rightmost B hops over into the empty cell), f(n) = 2 + 3 = 5;
(11) |B|E|B|W|B|W|W| (The second rightmost B hops over into the empty cell), f(n) = 3 + 3 = 6;
(12) |B|B|B|W|W|W|E| (The rightmost W hops over into the empty cell), f(n) = 2 + 3 = 5;
 (same as the initial node)