**Problem assignment 1**

*Due: Wednesday, September 18, 2019*

Problem 1

Part a.

The nodes will be expanded as:

S-A-D-S-B-D-E-S-A-A-D-A-C-E-E-S-A-D-F-B-A-D-S-B-D-S-B-D-E-S-A-S-B-D-G

The path would be: S-A-B-C-G. It’s the optimal path.

Part b.

The nodes will be expanded as:

S-D-A-E-A-S-B-D-S-F-B-D-B-D-S-D-A-C-E-A-E-A-S-D-A-C-E-C-E-A-E-A-S-C-E-A-E-A-S-D-A-E-A-S-B-D-S-G

The path would be S-A-B-C-G. It’s still the optimal path,

Part c.

The nodes will be expanded as:

S-A-D-B-D-F-R-D-S-V-R-S-S-F-R-D-S-G-B-F-S-F-B-S-D-G

The path would be S-A-B-C-G. it’s the optimal path.

Problem 2

Part a.

Yes, the bidirectional A\* is complete.

The bidirectional A\* algorithm is based on two A\* search expanded from beginning and end note. So, we only need to prove the completeness of A\* search.

For A\* search, the algorithm won’t stuck into infinite loops since when we expand the nodes, it’s based on breath-first rule.

Thus, the A\* search is complete, and the bidirectional A\* is also complete.

Part b.

Yes, the bidirectional A\* is optimal.

The start node is S, the end node is E.

Let’s suppose there’s a sub-optimal state G, and the optimal state is G’.

f(G) = fS(G) +fE(G) = gS(G)+gE(G)

Thus, before reaching the G’, at least from one side, there would be a middle state M, let’s suppose it’s on the forward direction.

Since G’ is optimal and G is sub-optimal, gS(G’)<gS(G).

M is the the middle state, and the heuristic is admissible, so fS(M)≤gS(G’).

Therefore, fS(M)<gS(G).

Thus, when expand nodes, state M would be considered first than state G, which means the algorithm won’t choose a sub-optimal result.

If the state is on the backward direction, the situation is also the same.

Problem 3