3CS3243 Assignment 2

Goh Kheng Xi, Jevan

A0199806L

Q5a.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | S | A | B | G | Admissible | Consistent |
|  | 0 | 0 | 0 | 0 | Yes | Yes |
|  | 8 | 1 | 1 | 0 | Yes | No |
|  | 9 | 3 | 2 | 0 | Yes | Yes |
|  | 6 | 3 | 1 | 0 | Yes | No |
|  | 8 | 4 | 2 | 0 | No | No |

5b.

S-A-B-G

5c. I would use h3. It is admissible and consistent which allows for optimal and complete limited graph search compared to only admissible heuristics which is only optimal in tree-based search which is slower compared to graph-based search.

Between h3 and h1, h3 is the dominant heuristic as h3(n) >= h1(n), hence, it is more efficient as the more dominant heuristic consider lesser number of paths (equal number for worst case). Thus, I would choose h3.

5d.

Since h5 is not admissible, there exist a node, n where h5(n) > h\*(n). For example, when n = A, h5(A) = 4 > h\*(A) = 3.

As h3 is admissible, h3(n) <= h\*(n) and therefore h3(n) <= h\*(n) < h5(n) which also means h3(n) < h5(n). Using the same example node n=A, h3(A) = 3 < h5(A) = 4

Hence max(h3(n), h5(n)) = h5(n). And since h5(n) > h\*(n) as established in the first statement of this proof, max(h3, h5) is not admissible.

Using this particular search problem as an example, when n = A,

h5(A) = 4 > h\*(A) = 3 ---- (1)

h3(A) = 3 <= h\*(A) = 3

since h5(A) > h3(A), max( h3(A), h5(A) ) = h5(A) = 4

From (1), h5(A) > h\*(A). Therefore, max(h3, h5) is not admissible.