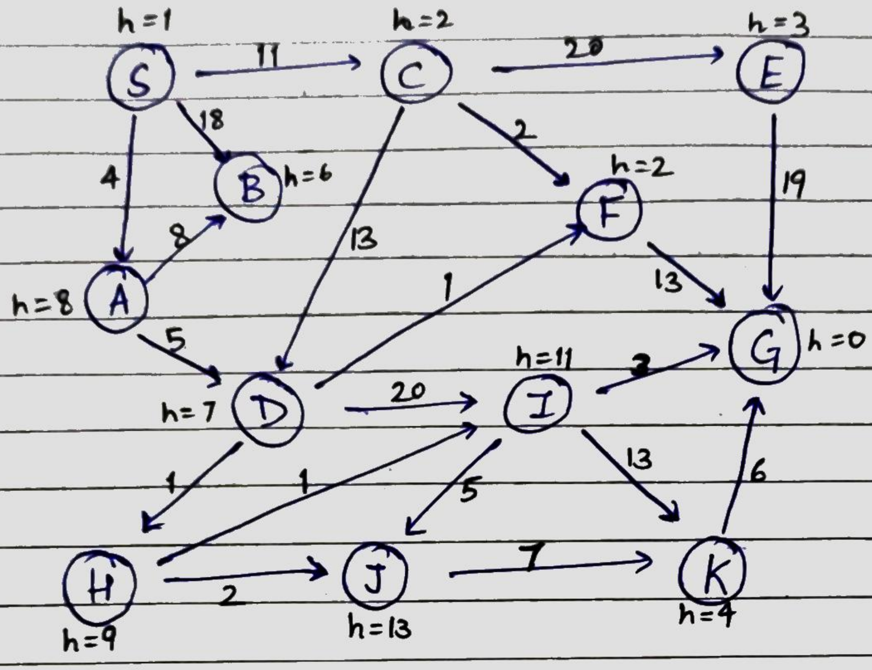
THEORY

Q1) (a) If the heuristic is admissible, the algorithm will still give the optimal solution even if the set of explored sets is not maintained. This is because the purpose of the set is to make the algorithm efficient by not revisiting states that have already been explored; the completeness of the algorithm depends only on the heuristic, which if admissible, is guaranteed to give the optimal soln.

(b) Yes, the new algorithm is complete. This is because of the same reasoning as explained above. The set of explored nodes is only used to make the algorithm more efficient. Its omission still guarantees that if a solution exists, using an admissible heuristic will find the solution.

(c) No, the new algorithm will not be faster as the same states will be reached in the process of the search in some cases, and explored multiple times thus leading to repetitive operations. This is avoided when keeping track of the explored states/nodes.

Q2)



$S \rightarrow \ldots \rightarrow G$

(i) **A* search**

$f(n) = g(n) + h(n)$ for node n

where $g(n) =$ Cost of path from S to n

$h(n) =$ Estimated cost of path from n to G

① Path = S

Frontier = ((S,0)) ; Visited = {S}

$f(A) = g(A) + h(A) = 4 + 8 = 12$

$f(B) = g(B) + h(B) = 18 + 6 = 24$

$f(C) = g(C) + h(C) = 11 + 2 = 13$

$\Rightarrow$ Frontier = [(A,12), (C,13), (B,24)]

② $f(B) = g(B) + h(B)$

$= (4+8) + 6 = 18 \longrightarrow < 24$ so (B,24) popped

$f(D) = g(D) + h(D) = (4+5) + 7 = 16$

$\Rightarrow$ Visited $= \{S, A\}$

Frontier = [(C,13), (D,16), (B,18), ~~(B,24)~~]

③ $f(D) = g(D) + h(D) = (11+13) + 7 = 31$

$f(E) = g(E) + h(E) = (11+20) + 3 = 34$

$f(F) = g(F) + h(F) = (11+2) + 2 = 15$

$\Rightarrow$ Visited $= \{S, A, C\}$

Frontier = [(F,15), (D,16), (B,18), ~~(B,24)~~, (D,31), (D,34)]

④ $f(G) = g(G) + h(G)$

$= (11+2+13) + 0 = 26$

Although it has reached the goal, there exist nodes in the frontier with f-value < cost of current path.

$\Rightarrow$ Visited $= \{S, A, C, F\}$

Nodes with f-value > cost of current optimal path are removed from Frontier.

$\Rightarrow$ Frontier = [(D,16), (B,18), ~~(B,24)~~]

⑤ $f(H) = g(H) + h(H)$

$= (4+5+1) + 9 = 19$

$f(I) = g(I) + h(I)$

$= (4+5+20) + 11 = 40 > 26 \longrightarrow$ Not considered

$f(F) = g(F) + h(F)$

$= (4+5+1) + 2 = 12$

$\Rightarrow$ Visited $= \{S, A, C, D\}$

Frontier = [(F,12), (B,18), (H,19)]

(6) $f(G) = g(G) + h(G)$
$= (4+5+1+13) + 0 = 23$
New most optimal f-value = 23
No nodes in frontier have f-value > 23

(7) B has no outgoing edges.
⇒ Visited = {S, A, B, C, D, F}
Frontier = [(H, 19)]

(8) $f(I) = g(I) + h(I)$
$= (4+5+1+1) + 11 = 22$
$f(J) = (4+5+1+2) + 13 = 25$ → > 23 so disregarded
Visited = {S, A, B, C, D, F, H}
Frontier = [(I, 22)]

(9) $f(G) = (4+5+1+1+3) + 0 = 14$
New f^*-value = 14
$f(J) = (4+5+1+1+5) + 13 = 29 > 14$ → disregarded
$f(K) = (4+5+1+1+13) + 4 = 28 > 14$ → disregarded
⇒ Visited = {S, A, B, C, D, F, H, I}
Frontier = []

⇒ Cost-optimal path : S → A → D → H → I → G
Cost = 14

(ii) **Best-First Search**

Evaluation funcⁿ → h(n) for node n
(estimated cost from n to G)

① Frontier = [(S,1)]

$h(A) = 8$

$h(B) = 6$

$h(C) = 2$

② Frontier [(C,2), (B,6), (A,8)]

expand

$h(E) = 3$, $h(F) = 2$, $h(D) = 7$

③ Frontier = [(F,2), (E,3), (B,6), (D,7), (A,8)]

expand

$h(G) = 0$

↳ found goal so it terminates

⇒ Cost-optimal path by BFS : $S \rightarrow C \rightarrow F \rightarrow G$

(iii) Djikstra Algorithm

$d(n)$ → Distance of node n from S along current path

Frontier → Priority queue ordered by b in (a,b)

Distance → Array depicting lowest distance from S

[S:0, A:∞, B:∞, C:∞, D:∞, E:∞, ..., K:∞]

(nodes with distance ∞ are not written as part of Distance for ease of representation)

① Frontier = [(S,0)]

$d(A) = 4$, $d(B) = 18$, $d(C) = 11$

Distance = [S:0, A:4, B:18, C:11]

② Frontier = [(A,4), (C,11), (B,18)]

expand

$d(B) = 4+8 = 12$, $d(D) = 4+5 = 9$,

Dist. = [S:0, A:4, B:12, C:11, D:9]

(3) Frontier = [(D,9), (C,11), (B,12)]

expand

$d(H) = 4+5+1 = 10$, $d(F) = 10$, $d(I) = 4+5+20 = 29$

Dist. = [S:0, A:4, B:12, C:11, D:9, F:10, H:10, I:29]

(4) Frontier = [(H,10), (F,10), (C,11), (B,12), (I,29)]

expand

$d(I) = 4+5+1+1 = 11$, $d(J) = 4+5+1+2 = 12$,

Dist. is updated with (I:11), (J:12)

(5) Frontier = [(F,10), (I,11), (C,11), (J,12), (B,12), (I,29)]

expand

$d(G) = 13+4+5+1 = 23$

Dist. is updated with (G:23)

(6) Frontier = [(I,11), (C,11), (J,12), (B,12), (I,29)]

expand

$d(J) = 4+5+1+1+5 = 16$ ⟶ > current dist. of J ⇒ ignored

$d(K) = 4+5+1+1+13 = 24$, $d(G) = 4+5+1+1+3 = 14$

Dist. is updated with (G:14), (K:24)

(7) Frontier = [(C,11), (J,12), (B,12), (K,24), (I,29)]

expand

$d(F) = 11+2 = 13$ ⟶ > current dist. of F ⇒ ignored

$d(E) = 11+20 = 31$ ⟶ updated to Dist.

$d(D) = 11+13 = 24$ ⟶ > current dist. of D ⇒ ignored

(8) Frontier = [(J,12), (B,12), (K,24), (I,29), (E,31)]
expand

$d(K) = 12+7 = 19$ ⟶ updated to Dist.

(9) Frontier = [(B,12), (K,19), (I,29), (E,31)]
no neighbours ⇒ popped

(10) Frontier = [(K,19), (I,29), (E,31)]
expand

$d(G) = 19+6 = 25$ ⟶ > current dist. of G ⇒ ignored

(11) Frontier = [(I,29), (E,31)]
expand

$d(G) = 29+3 = 32$, $d(J) = 29+5 = 34$, $d(K) = 29+13 = 42$

⟶ all > current dist-s of respective nodes ⇒ ignored

(12) Frontier = [(E,31)]
expand

$d(G) = 31+19 = 50$ ⟶ > current dist. of G ⇒ ignored

⇒ Cost-optimal path by Djikstra's algo. : S → A → D → H → I → G
with cost = 14