A* Algorithm.

for Search

→ uses heuristic function of h(n) & cost to reach the node 'n' from start state g(n).

→ finds shortest path through Search space.

↳ ~~Gives~~ provides fast & optimal result.

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

- h(n) → heuristic value (child node)
- g(n) → cost of each node

Algorithm :-

(i) Enter starting node in OPEN list

(ii) If open List is empty return FAIL

iii) Select node from OPEN List which has smallest value (gth)

↳ if node = Goal

return Success

iv) Expand node 'n' and generate all successors

↳ compute (g+h) for each successor node

v) if node 'n' is already in OPEN / CLOSED, attached to back pointer.

vi) go to step (ii)

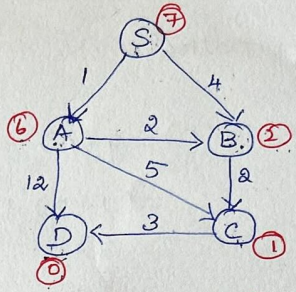
Advantages :-

1. Best searching algorithm
2. It is optimal & complete
3. helps in So solving Complex problems.

Disadvantages :

i) Doesn't always produce shortest path
2) Complexity issues
3) Requires memory.

Ex :-



S → start state
D → Goal state

S → A : f(n) = g(n) + h(n) = 1+6 = 7

S → B : f(n) = g(n) + h(n) = 4+2 = 6

6 < 7.

So choose path S → B.

Explore S → B path :- & keep S → A in hold.

S →(4) B →(2) C (1) : f(n) = g(n) + h(n)
= (4+2) + 1
= 6 + 1 = 7.

S →(4) B →(2) C →(3) D (0) : f(n) = g(n) + h(n)
= (4+2+3) + 0
= 9.

S → B → C → D : 9 → (1)

Explor

S → A paths (4)

$S \xrightarrow{1} A \xrightarrow{2} \overset{②}{B}$ = f(n) = g(n) + h(n)

= (1+2) + 2

= 3 + 2

= 5.

$S \xrightarrow{1} A \xrightarrow{5} \overset{①}{C}$ = f(n) = g(n) + h(n)

= (1+5) + 1

= 6 + 1

= 7.

$S \xrightarrow{1} A \xrightarrow{12} \overset{⑤}{D}$ = f(n) = g(n) + h(n)

= (1+12) + 0

= 13

S → A → D : 13 → (2)

from S → A → B : 5 < S → A → C : 7

So take smallest cost.

Explore :- S → A → B path (5)

$S \xrightarrow{1} A \xrightarrow{2} B \xrightarrow{2} \overset{①}{C}$: f(n) = g(n) + h(n)

= (1+2+2) + 1

= 5 + 1

= 6.

$S \xrightarrow{1} A \xrightarrow{2} B \xrightarrow{2} C \xrightarrow{3} \overset{⑥}{D}$: f(n) = g(n) + h(n)

= [1+2+2+3] + 0

= 8.

S → A → B → C → D : 8 → (3)

Now :- Explore

S → A → C

$S \xrightarrow{1} A \xrightarrow{5} C \xrightarrow{3} \overset{⑥}{D}$ = f(n) = g(n) + h(n)

= [1+5+3] + 0

= 9 + 0

= 9.

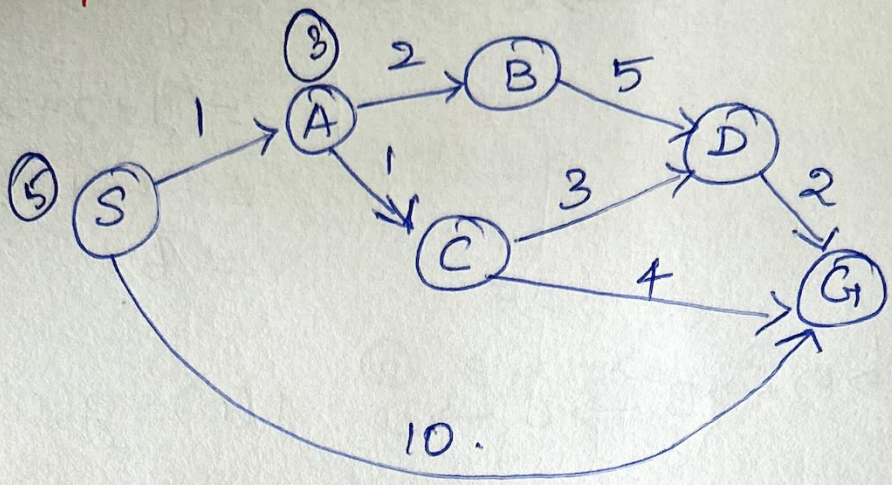
S → A → B → C → D : 9 → (4)

From ①, ⑤, ③ & ④ paths.

Optimal path is

S → A → B → C → D : 8.

Example :- 2.



State	h(n)
S	5
A	3
B	4
C	2
D	6
G	0

Start State :- S
Goal State : G
Find optimal sol. using A^*_v Alg. search.