DS & AI ENGINEERING

Artificial Intelligence

Informed search

Lecture No.- 6

By- Aditya sir

Recap of Previous Lecture

Topic IDA*

Topic Questions

Topic

Topic

Topics to be Covered

Topic

Topic

Practice

Topic

Concepts

About Aditya Jain sir

1. Appeared for GATE during BTech and secured AIR 60 in GATE in very first attempt - City topper
2. Represented college as the first Google DSC Ambassador.
3. The only student from the batch to secure an internship at Amazon. (9+ CGPA)
4. Had offer from IIT Bombay and IISc Bangalore to join the Masters program
5. Joined IIT Bombay for my 2 year Masters program, specialization in Data Science
6. Published multiple research papers in well known conferences along with the team
7. Received the prestigious excellence in Research award from IIT Bombay for my Masters thesis in ML
8. Completed my Masters with an overall GPA of 9.36/10
9. Joined Dream11 as a Data Scientist
10. Have mentored 15,000+ students & working professions in field of Data Science and Analytics
11. Have been mentoring & teaching GATE aspirants to secure a great rank in limited time
12. Have got around 27.5K followers on Linkedin where I share my insights and guide students and professionals.

@ADITYASIR_PW

Telegram Link for Aditya Jain sir:

https://t.me/AdityaSir_PW

[MSQ]

#Q. The heuristic path algorithm is a best-first search in which $f(n) = (2-w)\, g(n) + w\, h(n)$.

Select the correct statement(s).

$f = (2 \cdot w) g + w \cdot h$

A. For w = 1, f(n) represents the A* algorithm. $\rightarrow f = g + h$

B. For w = 2, f(n) is complete. $\rightarrow f = 2 \cdot h$ ✗

C. For w = 2, f(n) is optimal. $\rightarrow f = 2 \cdot h$ ✗

D. For w = 0, f(n) represents UCS. $\rightarrow F = 2 \cdot g$ ✗

1) UCS: $F = g$

2) GBFS $\Rightarrow F = h$

3) $A^* \Rightarrow f = g + h$

$a + a = 2a$ ✓

$a + a = 10a$? ✗

$F = g$

$F = 2g$ ✗

$F \propto g$

[NAT]

#Q. Consider the below graph with heuristic values.

State	Heuristic: h(n)
A	366
B	374
C	329
D	244
E	253
F	178
G	193
H	98
I	0

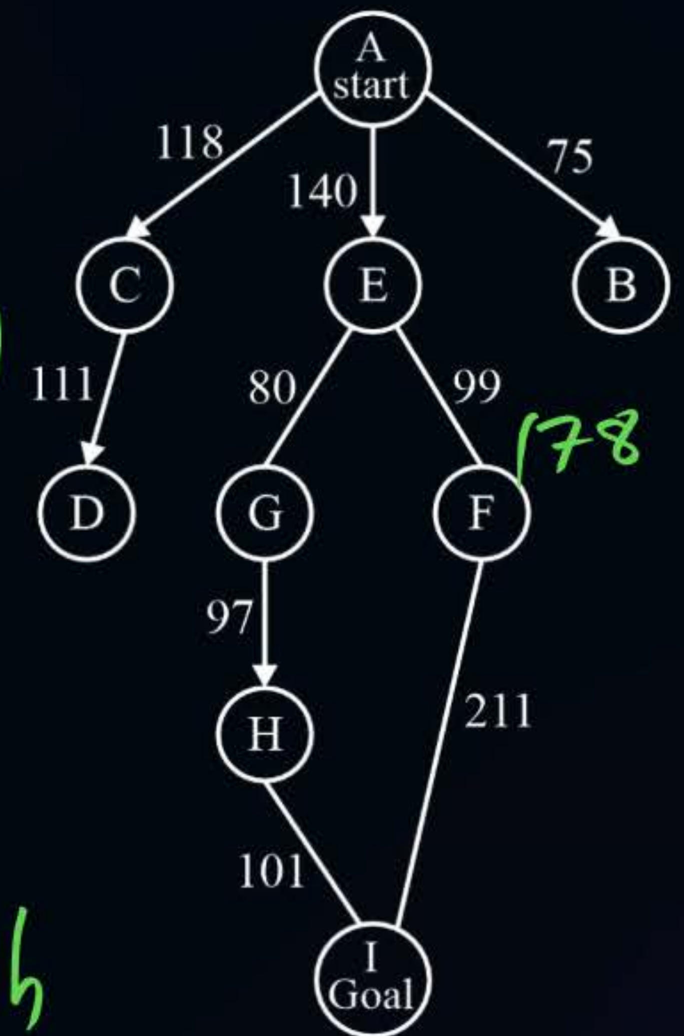
A) 178+211

B) 253+99+211

C) 178

D) 253

GBFS ⇒ $f = h$

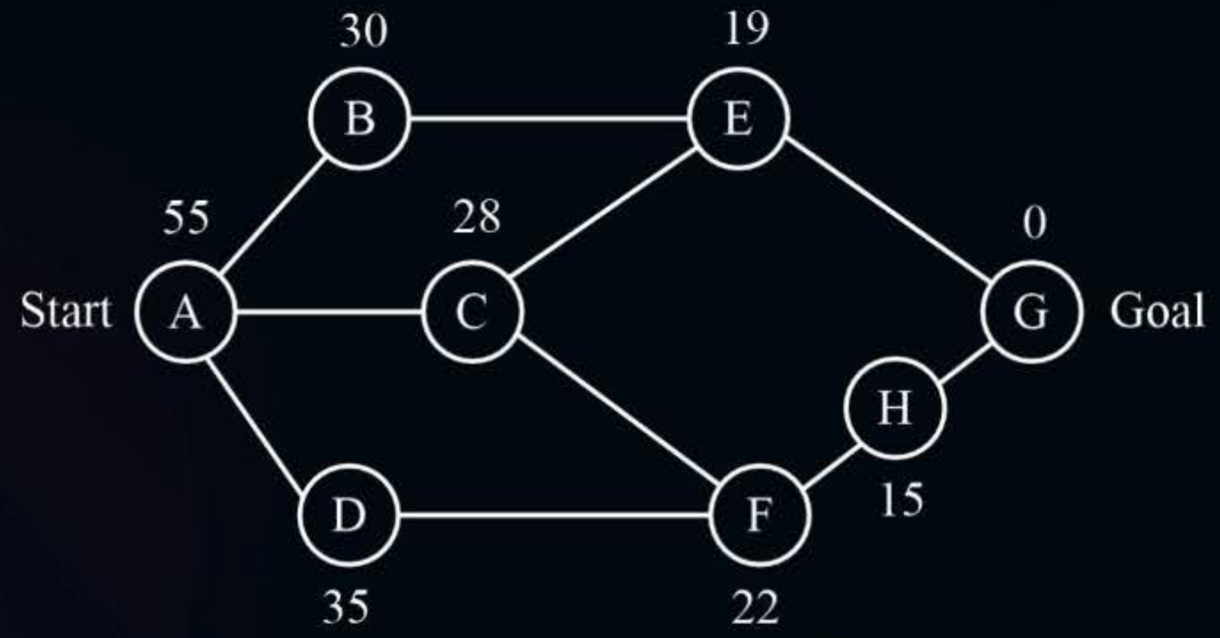


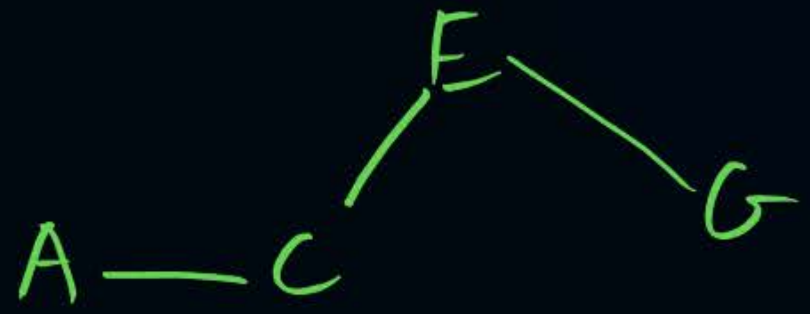
What will be the value of evaluation function for the node F, if we apply Greedy Search algorithm?

[MCQ]

GBFS

#Q. Consider the following graph use the best first search algorithm to find the path from node A to node G (Assume node A is start node and node G is goal node). Which of the following path sequence is correct?





A $A \rightarrow D \rightarrow C \rightarrow G$

B $A \rightarrow C \rightarrow E \rightarrow G$

C $A \rightarrow C \rightarrow F \rightarrow H \rightarrow G$

D $A \rightarrow D \rightarrow F \rightarrow G$

[~~MCQ~~] MSQ

#Q. Which of the following is/are true about A* Search?

A, B, C

A. A* search is both complete and optimal under certain conditions for the heuristic function h(n)

B. A* search is complete but not optimal under certain conditions for the heuristic function h(n)

C. A* search is similar to uniform-cost search, it uses the sum of g and h values instead of g

D. A search is similar to uniform-cost search, as it uses the cumulative cost of node ‘n’ from a start node

[MCQ]

#Q. In Greedy Best First Search, what is the evaluation function used to select the next node for expansion?

$f = h$

85%

A $f(n) = g(n) + h(n) \longrightarrow A^*$

B $f(n) = g(n) - h(n)$

C $f(n) = h(n)$

D $f(n) = g(n) \longrightarrow$ UCS

[MCQ]

UCS

50%

#Q. Consider the graph given below:

If we apply Uniform Cost Search Algorithm, then what will be the path to reach node G from node S?

Dj

S → G

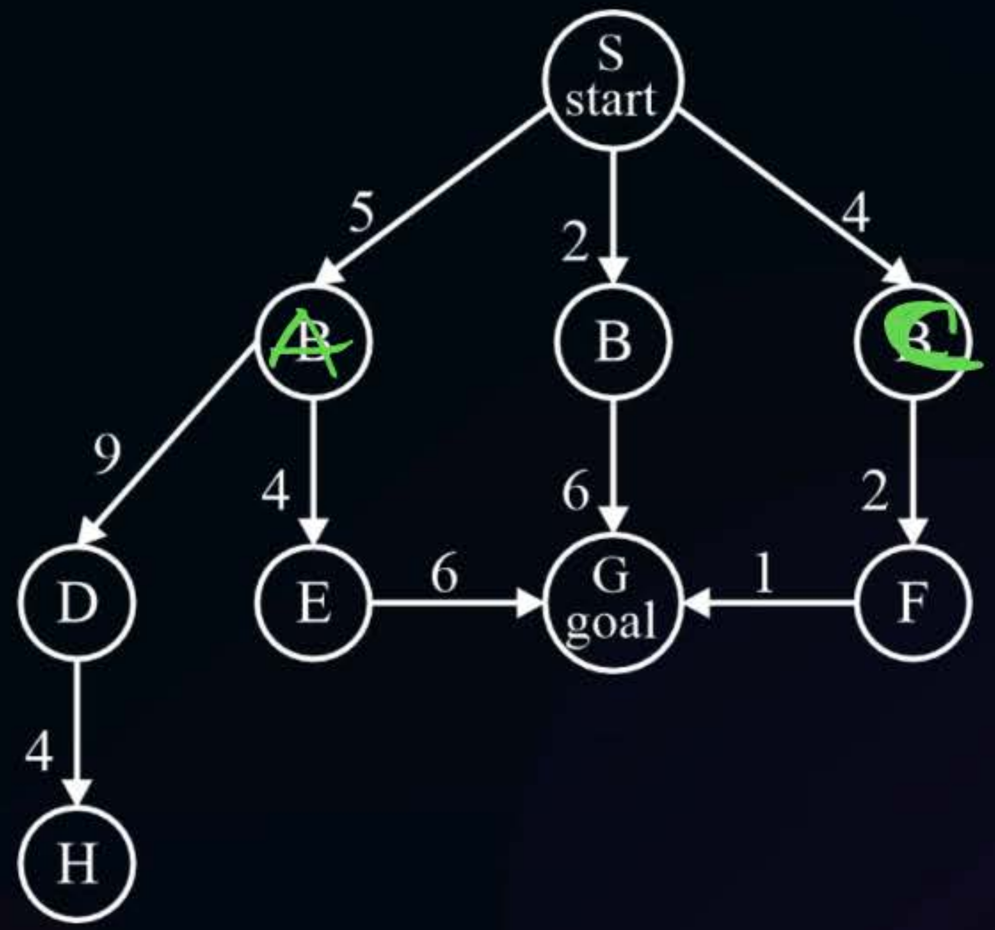
A. S → C → F → G ⇒ 4+2+1=7

B. S → B → G ⇒ 2+6=8

C. S → A → E → G ⇒ 5+4+6=15

D. S → A → B → C → G ⇒ ✗

Shortcut



[MCQ]

UCS

50%

#Q. Consider the graph given below:

If we apply Uniform Cost Search Algorithm, then what will be the path to reach node G from node S?

SCFG

S → G

A. S → C → F → G

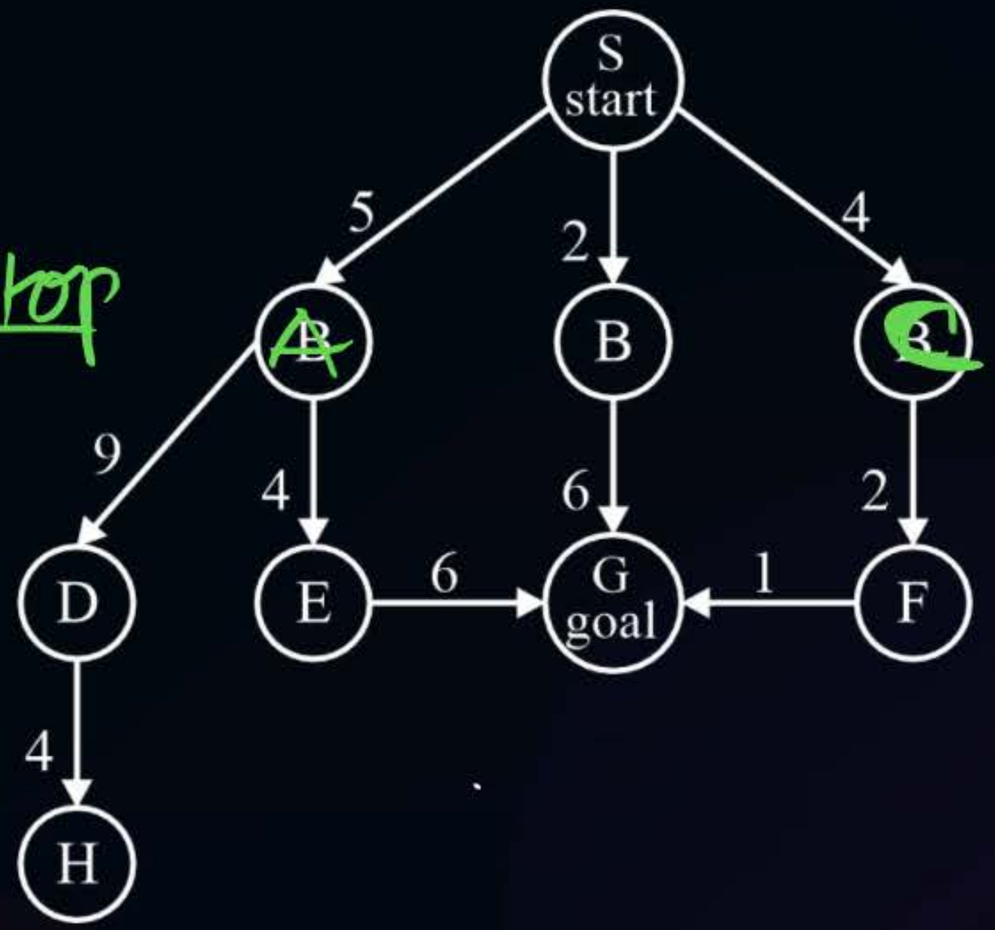
B. S → B → G

C. S → A → E → G

D. S → A → B → C → G

Open	Closed: S	B	C	A	F	G
S						
A	5	5	5	5	5	
B	2	2	2	2	2	
C	4	4	4	4	4	
G	×	8	8	8 → 7		
F	×	×	6	6	6	
D	×	×	×	14	14	
E	×	×	×	9	9	

stop



[NAT]

#Q. In the problem the start state is S, and the goal state is G. The transition costs are next to the edges, and the heuristic estimate, h, of the distance from the state to the goal is in the state's node. Assume ties are always broken by choosing the state which comes first alphabetically.

- What is the order of states expanded using A* search?

(Graph Search)

SABDCEFG

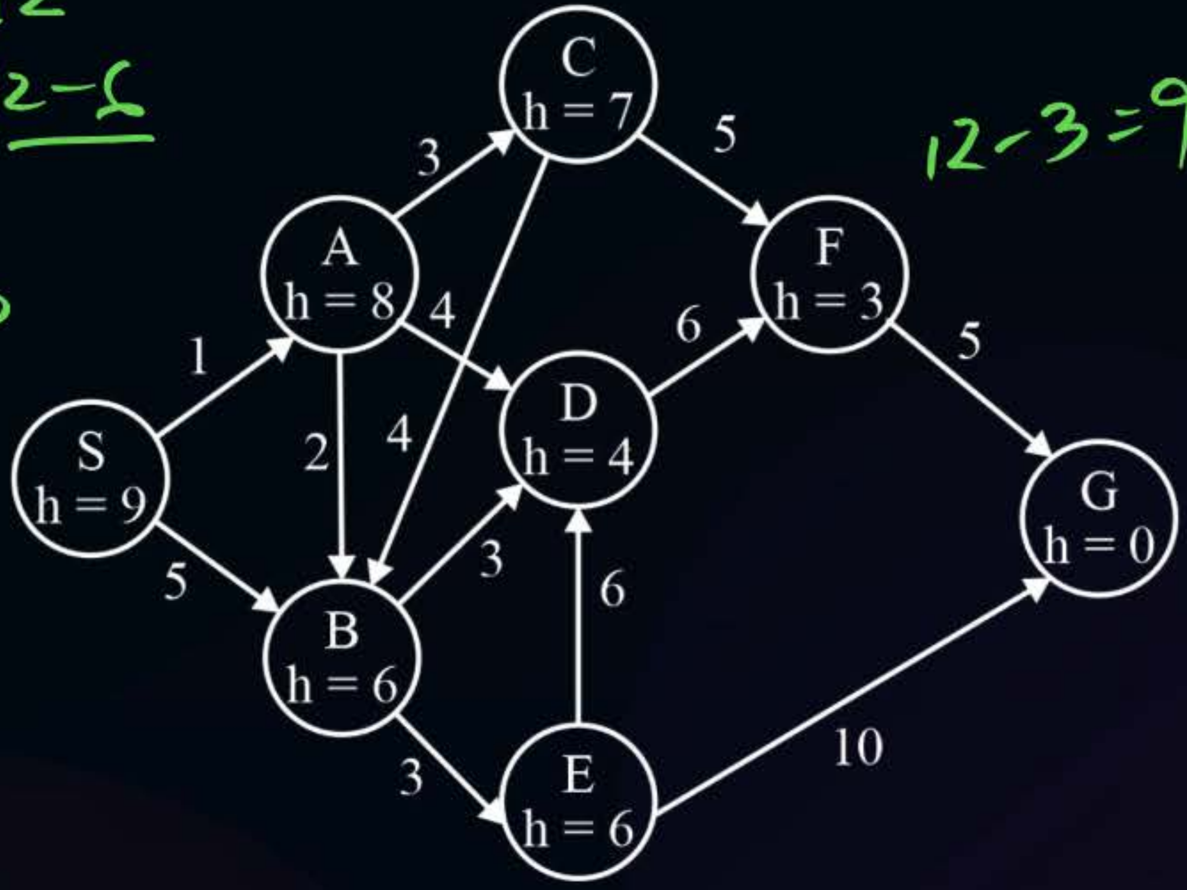
Open	Closed: S	A	B	D	C	E	F	G
S×								
A×	9	9	9	9	9	9	9	
B×	11	9	9	9	9	9	9	
C×	×	11	11	11	11	11	11	
D×	×	9	9	9	9	9	9	
E×	×	×	12	12	12	12	12	
F×	×	×	×	14→12		12	12	
G	×	×	×	×	×	16	14	

L stop

$x+6=12$

$x=12-6$

$12-3=9$



[MCQ]

#Q. Consider a graph where each edge has a different cost. If you are using the Uniform Cost Search algorithm to find the shortest path from a start node A to a goal node G, which of the following statements is TRUE regarding the characteristics of UCS?

UCS → Complete ✓
optimal ✓

80%

A. UCS is complete, but not optimal. ✗

B. UCS is both complete and optimal ✓

C. UCS can only be used in graphs with uniform edge costs. ✗

D. UCS uses a depth-first strategy ✗

[MCQ]

#Q. Let h_1 and h_2 be two admissible heuristics used in A* search.

Which ONE of the following expressions is always an admissible heuristic?

A $h_1 + h_2$ ✗ $\quad 9+10 \le 10$ ✗

B $h_1 \times h_2$ $\quad 9 \times 10 \le 10$ ✗

C $|h_1 - h_2|$ ✓

D $h_1 / h_2 \ (h_1 \neq h_2)$ ✗

$\frac{10}{0.1} = 100 \le 10$ ✗

$A \xrightarrow{10} B$

$[0.1, 1, 2, \ldots 10]$

h_1, h_2

100%

$h \le h^*$

[MCQ]

#Q. If h is an admissible heuristic (non-negative), which of the following can never be an admissible heuristic?

$A \xrightarrow{10} B$

$[0.1, 1, 2, 3 \ldots 10]$

$0 \le h \le 10$

47%

A $h + 1$ $\Rightarrow h = 1,\ 1+1 \le 10$

B $2h$ $\Rightarrow h = 1,\ 2 \times 1 \le 10$

C $\sqrt{h}$ $\Rightarrow h = 1,\ \sqrt{1} = 1 \le 10$

D They all can be admissible under some situation.

A)

B

X

77.33/100

-0.66

[MSQ]

#Q. If h_1 and h_2 are admissible heuristics, which of the following are guaranteed to be admissible?

A. $h_1 + h_2$ ✗ $10+9=19$

B. $\min(h_1, h_2)$ ✓

C. $\max(h_1, h_2)$ ✓ $\max(9,10)=10$

D. $\alpha h_1 + (1 - \alpha)\, h_2$ for $\alpha \in [0, 1]$ ✓

$A \xrightarrow{10} B$

$\downarrow$

$[0.1, 0.5, 1, 2 \ldots 10]$

Tricky

$\alpha = 0,$

$h = 0 \cdot h_1 + 1 h_2$

$h = h_2$

$[h_1, h_2]$

$\alpha = 1$

$h = h_1$

$\alpha = 0.5$

$= 0.5 h_1 + 0.5 h_2$

$= \frac{9+10}{2} \le 10$

[NAT]

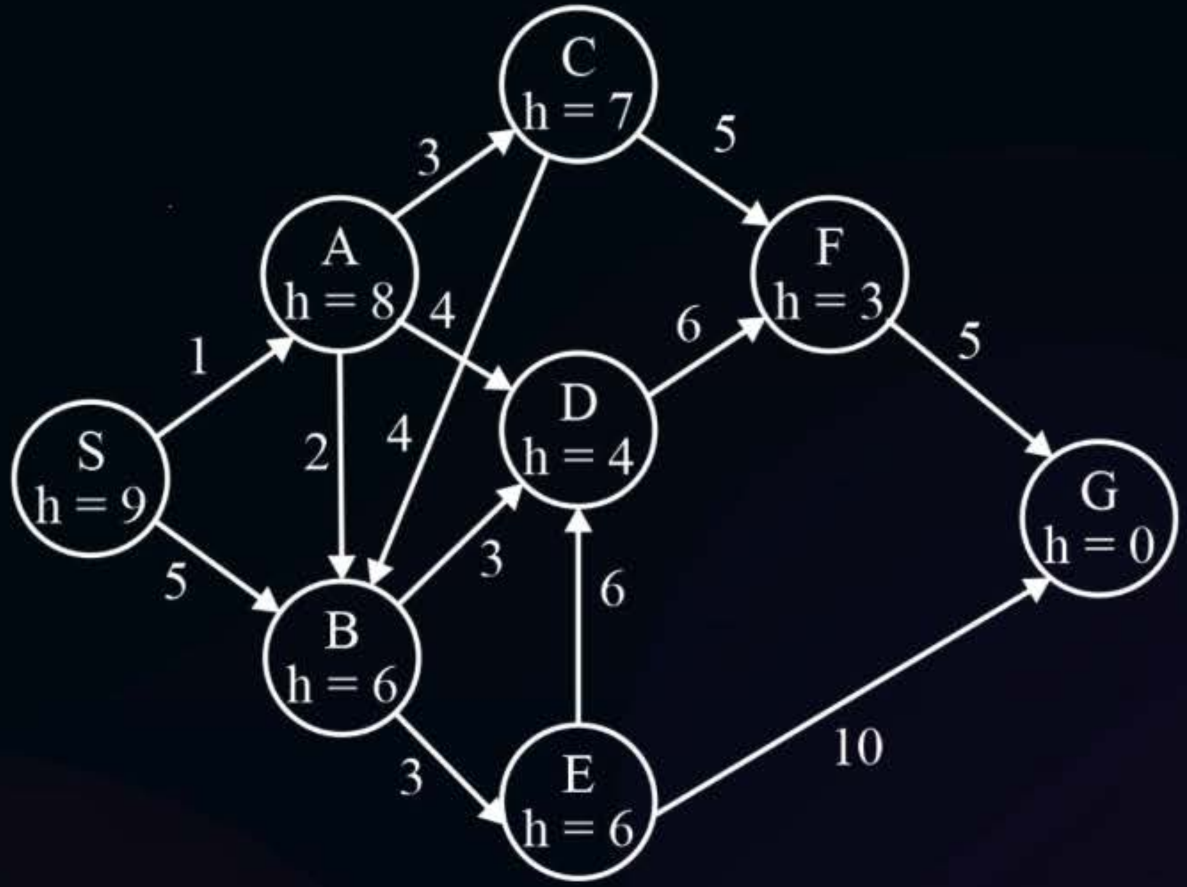
#Q. In the problem the start state is S, and the goal state is G. The transition costs are next to the edges, and the heuristic estimate, h, of the distance from the state to the goal is in the state's node. Assume ties are always broken by choosing the state which comes first alphabetically.

- What is the order of states expanded using Depth First Search? Assume DFS terminates as soon as it reaches the goal state.

$\boxed{SABDFG}$

DFS

$S \to A \to B \to D \to F \to G$



[MSQ]

Adv

#Q. Which of the following evaluation functions will result in identical behaviour to greedy best-first search (assume all edge costs are positive)?

A) $f(n) = 100*h(n)$ ✓

B) $f(n) = g(n) * h(n)$ ✗

C) $f(n) = h(n^2)$ $(h(n))^2$ ✓

D) $f(n) = 1/h(n)$ ✗

GBFS

$f = h$

A, C

Idea: take min h

only depend on h

node	$h(n)$	$100*h$	$1/h$	h^2	$h \times g$ g
P	1	100 ✓	$1/1$	1	1000
Q	5	500	$1/5$	25	2
R	2	200	$1/2$	4	5
S	7	700	$1/7$	49	1

BFHS vs GBFS* → GBFS (not Complete)

BFHS (Complete)

- BFHS is a hybrid algorithm that combines aspects of Breadth-First Search (BFS) and heuristic guidance. It systematically explores the search space level by level, but with a preference for nodes with better heuristic values.
- GBFS is an algorithm that always expands the node that appears to be closest to the goal according to a heuristic. It focuses purely on the heuristic value (h(n)), without considering the cost to reach the node (g(n)).
- BFHS is generally less efficient than GBFS because it explores all nodes at a particular depth before moving on, potentially expanding unnecessary nodes.
- BFHS need more memory.
- GBFS need less memory.
- BFSH expand more nodes than GBFS.

THANK - YOU