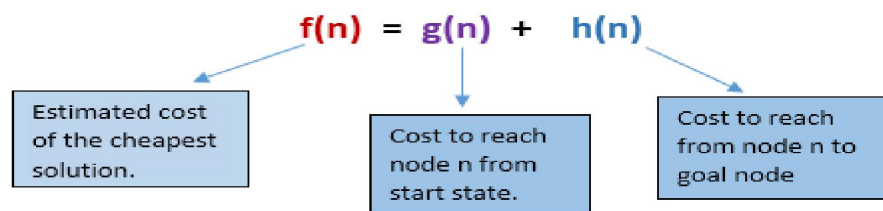## A* Algorithm:

- A* Algorithm is one of the best and popular techniques used for path finding and graph traversals.
- A lot of games and web-based maps use this algorithm for finding the shortest path efficiently.
- It is essentially a best first search algorithm.

Working Procedure:

A* Algorithm works as follows:

- It maintains a tree of paths originating at the start node.
- It extends those paths one edge at a time.
- It continues until its termination criterion is satisfied.

A* Algorithm extends the path that minimizes the following function-

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

Estimated cost of the cheapest solution.	Cost to reach node n from start state.	Cost to reach from node n to goal node

Here,

- 'n' is the last node on the path
- g(n) is the cost of the path from start node to node 'n'
- h(n) is a heuristic function that estimates cost of the cheapest path from node 'n' to the goal node

Algorithm:

- The implementation of A* Algorithm involves maintaining two lists- OPEN and CLOSED.
- OPEN contains those nodes that have been evaluated by the heuristic function but have not been expanded into successors yet.
- CLOSED contains those nodes that have already been visited.

Problem-01(For Odd ID Students): Solve 8-Puzzle Game using A* Algorithm.

Given an initial state of a 8-puzzle problem and final state to be reached-

2	8	3
1	6	4
7		5

Initial State

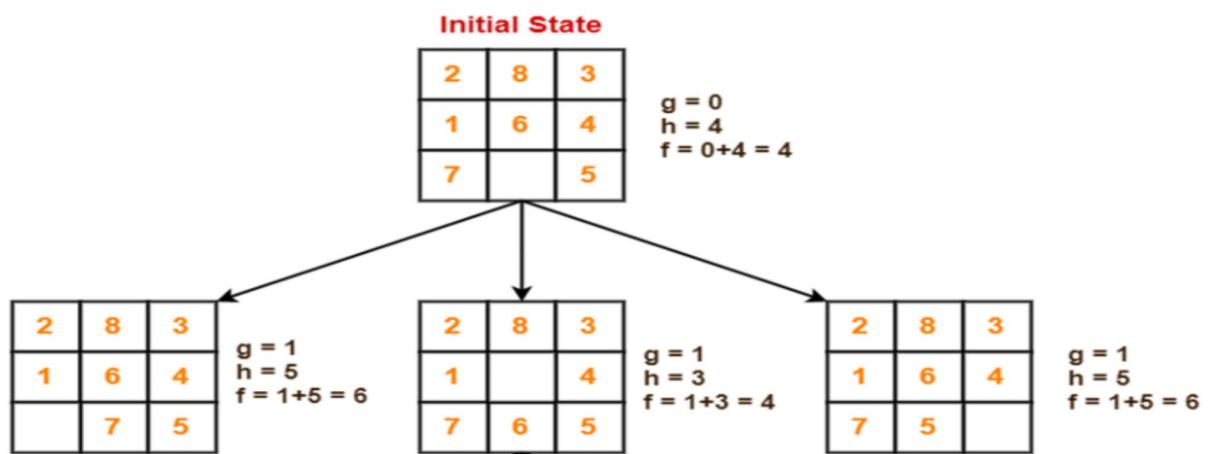| 1	2	3
8		4
7	6	5

Final State

Find the most cost-effective path to reach the final state from initial state using A* Algorithm.
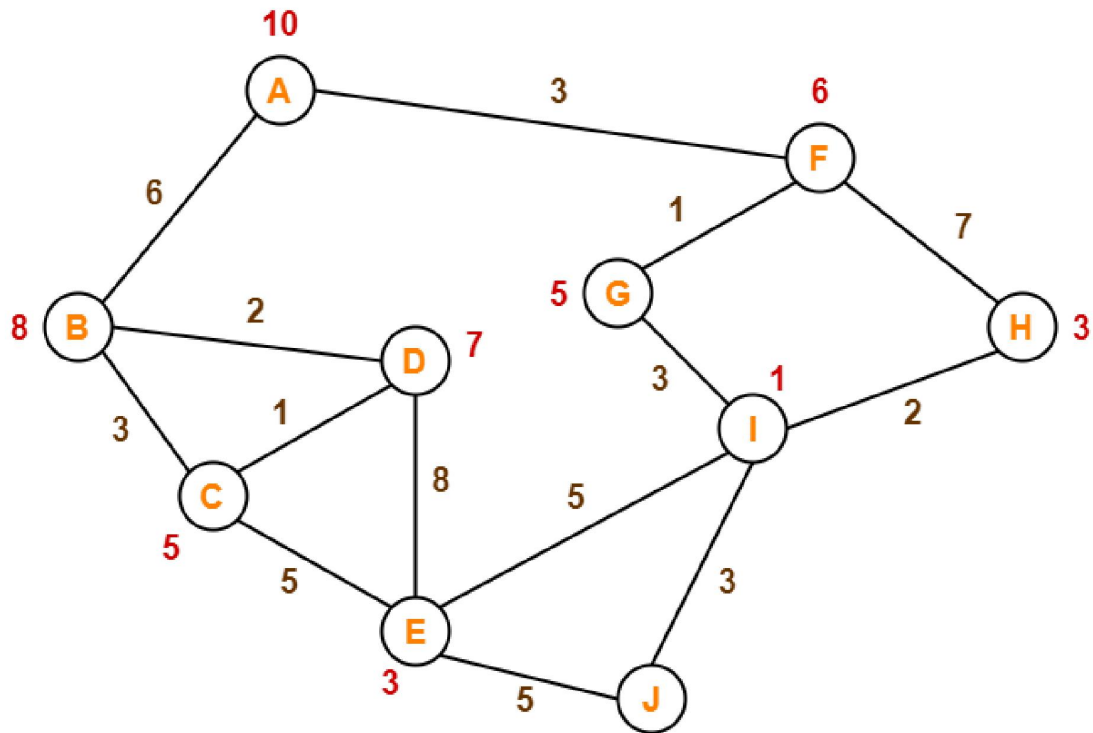
Solution Hints:

Consider $g(n)$ = Depth of node and $h(n)$ = Number of misplaced tiles.

- A* Algorithm maintains a tree of paths originating at the initial state.
- It extends those paths one edge at a time.
- It continues until final state is reached.

Initial State

2	8	3
1	6	4
7		5

$g = 0$
$h = 4$
$f = 0+4 = 4$

2	8	3
1	6	4
	7	5

$g = 1$
$h = 5$
$f = 1+5 = 6$

2	8	3
1		4
7	6	5

$g = 1$
$h = 3$
$f = 1+3 = 4$

2	8	3
1	6	4
7	5	

$g = 1$
$h = 5$
$f = 1+5 = 6$

Consider the following graph. Find the most cost-effective path to reach from start state A to final state J using A* Algorithm.

Solution Hints:

- The numbers written on edges represent the distance between the nodes.
- The numbers written on nodes represent the heuristic value.
- Here, node A is the start state and node J is the goal state.
- For efficiency, you can skip backward ancestor if needed.