SPRING **MID SEMESTER EXAMINATION-2025**

School of Computer Engineering
Kalinga Institute of Industrial Technology, Deemed to be University
Artificial Intelligence
[(CS30002)]

Answer Scheme

1.

a) The performance measure is very crucial for a rational agent . Justify

Answer:- The performance measure is crucial for a rational agent as it defines success and guides decision-making. It helps the agent choose optimal actions, adapt to changes, and efficiently use resources. Without a proper performance measure, the agent would lack direction and might behave sub optimally. Hence, it is essential for evaluating and improving an agent's effectiveness.

b) Consider the following heuristics for a path finding problem on a 2D grid:

H1(n) = Euclidean distance to the goal

H2(n) = Twice the Euclidean distance to the goal .

Which of these heuristics are admissible and why?

Answer:-

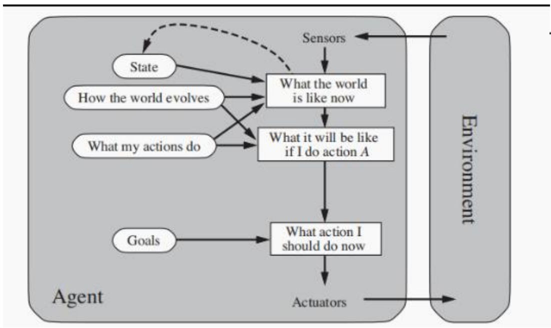
H1(n) is admissible because it never overestimates the true cost.

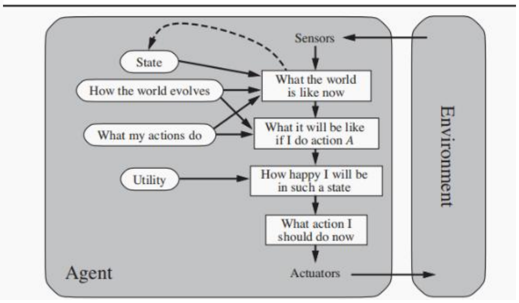
H2(n) is not admissible because it can overestimate the true cost, violating the admissibility condition.

Since H2(n) = 2×(Euclidean Distance), it may overestimate the true cost. In many cases, the actual path cost h(n) will be less than twice the Euclidean distance. If a heuristic overestimates the true cost for even a single node, it is not admissible.

c) Differentiate between goal based agent and utility based agent? With proper diagram.

Answer:-





d) Provide the Performance and Environment components for PEAS of AI Powered Drone for environment monitoring?

Answer:-

PEAS for an AI-Powered Drone for Environmental Monitoring

1. Performance Measures:

Accuracy of environmental data collection (e.g., air quality, temperature, pollution levels), Coverage area efficiency (how well it monitors a given region) , Energy efficiency (battery life and resource consumption), Collision avoidance and safe navigation, Response time to changing environmental conditions and Data transmission speed and reliability.

2. Environment:

Outdoor environment (forests, cities, water bodies, agricultural fields, etc.), Weather conditions (rain, wind, fog, temperature variations), Obstacles (birds, buildings, trees, power lines) and Airborne pollutants, smoke, and fog that may affect visibility.

e) What is the functional difference between Hill Climbing and Greedy Best First Search Algorithm.

Answer:-

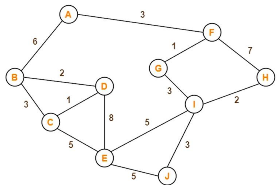
i) Hill climbing algorithm only considers immediate neighbors and moves in the direction of increasing value (or decreasing cost). Where as greedy best first search always picks the node with the lowest heuristic cost (e.g., shortest estimated distance to the goal).

ii) In case of hill climbing optimality is not guaranteed but in case of greedy best first search optimality is guaranteed.

iii) Hill climbing evaluates neighbors only, whereas the greedy best first search uses global heuristic values.

2 (a) Find the path from A to J using Breadth First Search. Show the contents of the data structure after each step. Break ties by visiting lexicographically earlier nodes first.

[3 Marks]



Answer:-

Reached	Frontier
A () 0	#0 A 0
Processing #0 A 0	
A () 0	
B (A) 1	#1 B 1
F (A) 1	#2 F 1
Processing #1 B 1	
A () 0	
B (A) 1	
F (A) 1	#2 F 1
C (B) 2	#3 C 2
D (B) 2	#4 D 2
Processing #2 F 1	
A () 0	
B (A) 1	
F (A) 1	
C (B) 2	#3 C 2
D (B) 2	#4 D 2
G (F) 2	#5 G 2
H (F) 2	#6 H 2
Processing #3 C 2	
A () 0	
B (A) 1	
F (A) 1	
C (B) 2	
D (B) 2	#4 D 2
G (F) 2	#5 G 2
H (F) 2	#6 H 2
E (C) 3	#7 E 3
Processing #4 D 2	
A () 0	
B (A) 1	
F (A) 1	
C (B) 2	
D (B) 2	
G (F) 2	#5 G 2
H (F) 2	#6 H 2
E (C) 3	#7 E 3
Processing #5 G 2	
A () 0	
B (A) 1	
F (A) 1	
C (B) 2	
D (B) 2	
G (F) 2	
H (F) 2	#6 H 2
E (C) 3	#7 E 3
I (G) 3	#8 I 3
Processing #6 H 2	
A () 0	
B (A) 1	
F (A) 1	
C (B) 2	
D (B) 2	
G (F) 2	
H (F) 2	
E (C) 3	#7 E 3
I (G) 3	#8 I 3
Processing #7 E 3	
A () 0	
B (A) 1	
F (A) 1	
C (B) 2	
D (B) 2	
G (F) 2	
H (F) 2	
E (C) 3	
I (G) 3	#8 I 3
J (E) 4	#9 J 4
Processing #8 I 3	
A () 0	
B (A) 1	
F (A) 1	
C (B) 2	
D (B) 2	
G (F) 2	
H (F) 2	
E (C) 3	
I (G) 3	
J (E) 4	#9 J 4
Processing #9 J 4	

Goal reached in 10 iterations: A B C E J

ii) Evaluation scheme:

history of reached states (1 point)

history of frontier (1 point)

correct answer A → B → C → E → J (1 point)

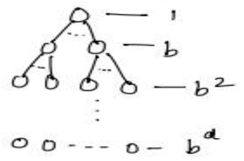
(b) Prove / Disprove the following statement :-

"The time complexity of Iterative deepening depth-first search is asymptotically similar to breadth first search".

[2 Marks]

Answer:-

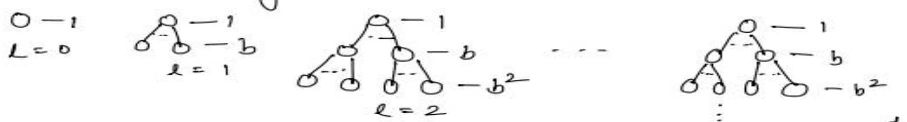
Time complexity of BFS:



so # of nodes generated by BFS = $1 + b + b^2 + \cdots + b^d$

so time complexity of BFS is $O(b^d)$

Time Complexity of IDS:



$0 \; 0 \; \cdots \; 0 - b^d$

$l = d$

so total number of nodes generated by IDS are –

$d \times 1 + (d-1) \times b + (d-2) \times b^2 + \cdots + 1 \times b^d$

so time complexity of IDS is also $O(b^d)$ same as BFS.

3(a) Find the path from the initial state to goal state for the following 8-puzzle problem using the A* algorithm. Use Manhattan distance to calculate the heuristic value.

[3 Marks]

Initial State

1	2	3
8		4
7	6	5

Goal State

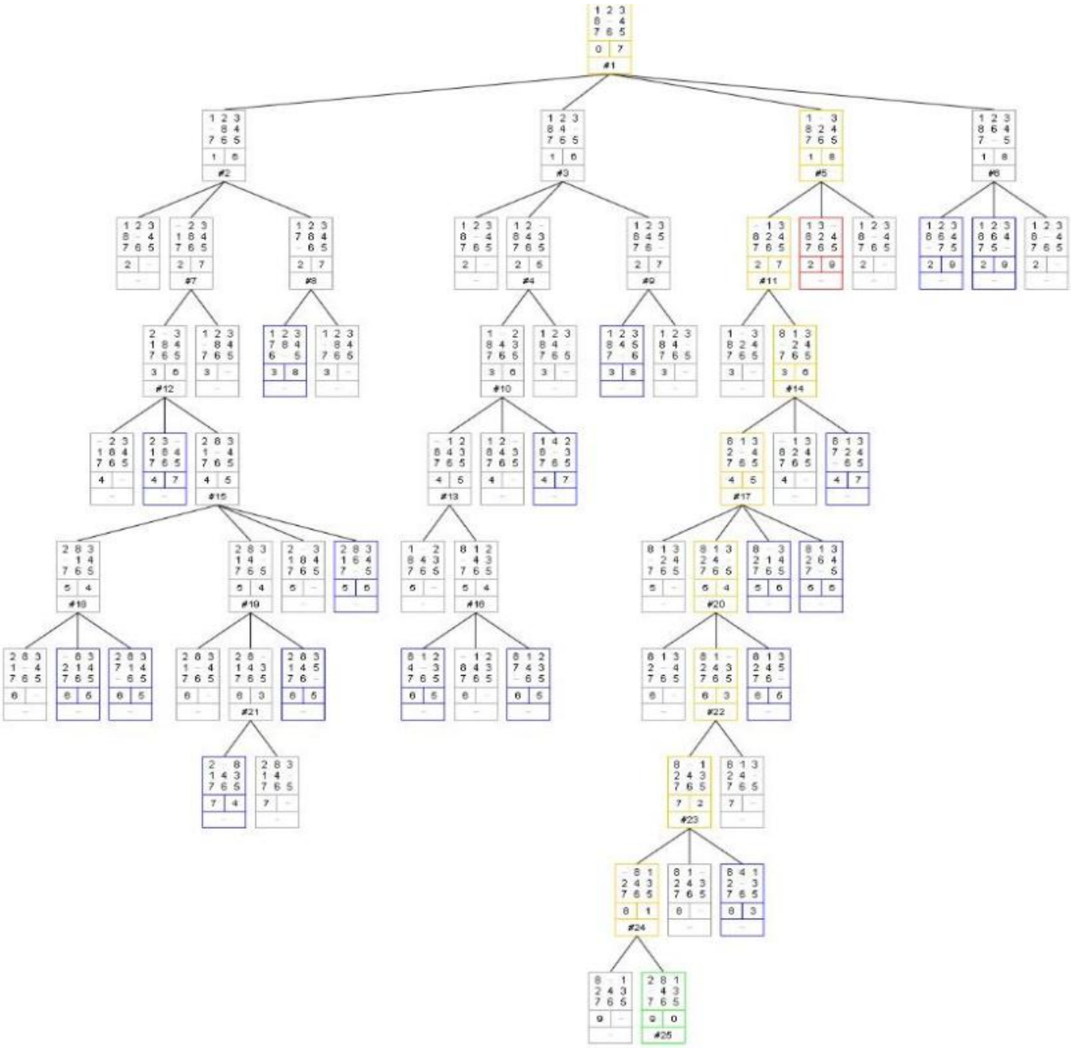
2	8	1
	4	3
7	6	5

Answer:-

Note:- As the search tree is very large , therefore the marks will be given based on the correct approach.

Calculation of heuristic value of the states (1 marks), selection of nodes/states for expansion based on A* algorithm (1 Marks) and generation of new states/successors from the current (1 Marks).

For reference the entire tree is provided as follows.



3(b) Prove/Disprove the following statement with proper example :-

"A heuristic is Admissible if and only if it is Consistent".

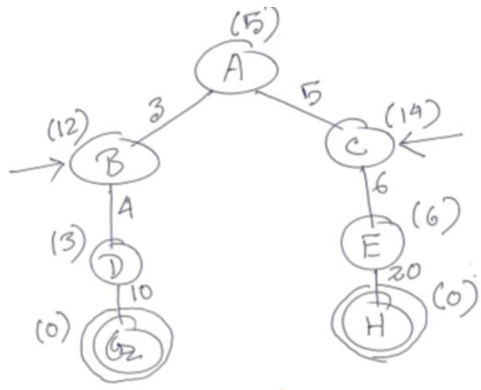
[2 Marks]

Answer:-

This statement is false. [if someone written true then 0]

Proof:- To disprove the above statement, we can consider the following graphs. Values on the edges are cost and values inside the bracket is heuristic value of that node.

Node	h(n)	h*(n)	Admissible?
A	5	17	h(n)<=h*(n). so Admissible
B	12	14	h(n)<=h*(n). so Admissible
C	14	26	h(n)<=h*(n). so Admissible
D	3	10	h(n)<=h*(n). so Admissible
E	6	20	h(n)<=h*(n). so Admissible
G	0	0	h(n)<=h*(n). so Admissible
H	0	0	h(n)<=h*(n). so Admissible



But, for Node B, h(n)= 12

For Node D, h(n)=3

Cost of B to D is 4

Now we can find that,   12> (3+4) =7 i.e. h(B) >= h(D) + cost (B, D). So, at Node B it is not Consistent.

Same for Node C

For Node C, h(n)= 14 . For Node E, h(n)=6   , Cost of C to E is 6.

Now we can find that,

14> (6+6) =12 i.e. . h(C) >= h(E) + cost (C, E)

So, at Node C it is not Consistent. So, we can conclude that the above heuristic is admissible but not consistent. So, the above statement is false.

4(a) Provide the state space representation of the following problems:-

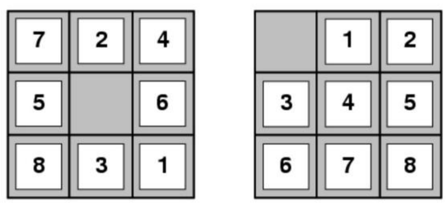
[2.5 Marks]

(i) 8-Puzzle.

(ii) N-Queen .

Answer:-

The 8-puzzle problem…



Start State   Goal State

- states? locations of tiles
- Initial state? Any state can be designated as the initial state.
- actions? move blank left, right, up, down
- Transition model? Maps a state and action to a resulting state; for example, if we apply Left to the start state in Figure, the resulting state has the 5 and the blank switched.
- goal state? = goal state (given)
- Action cost? 1 per move

N Queen problem

In the N-Queens problem, the state space representation includes:
• States:Each state represents a configuration of queens placed on the chessboard, where a queen is positioned in each N row.
• Initial State:Any arrangement of N Queens in N rows.
• Actions:Placing a queen in an available square on the next empty row.
• Transition Model:Given a current state and an action (placing a queen in a specific square), the transition model returns the new state resulting from that action.
• Goal State:A configuration where all N queens are placed on the board without any of them attacking each other (horizontally, vertically or diagonally).
• Action Cost:Usually, each action (placing a queen in a new square) is considered to have a cost of 1, so the path cost is simply the number of steps taken to reach the goal state.

4.(b) From the above problems (8-Puzzle or N-Queen), choose any one of the problems and describe the approach of applying Steepest Hill Climbing for searching the goal state. Choose any admissible heuristic.

[2.5 Marks]

Answer:-

8 puzzle problem using Steepest Hill Climbing

1	**2**	**4**
5		**7**
3	**6**	**8**

Initial State

1	**4**	**7**
2	**5**	**8**
3	**6**	

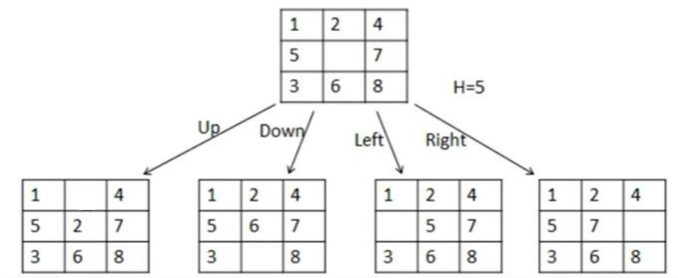
Goal State

Considering h is an admissible heuristic because it is clear that any tile that is out of place must be moved at least once.

H = the number of misplaced tiles

In the above scenario, **H = 5** because the position of 5 tiles (2, 4, 5, 7, 8) are changed.

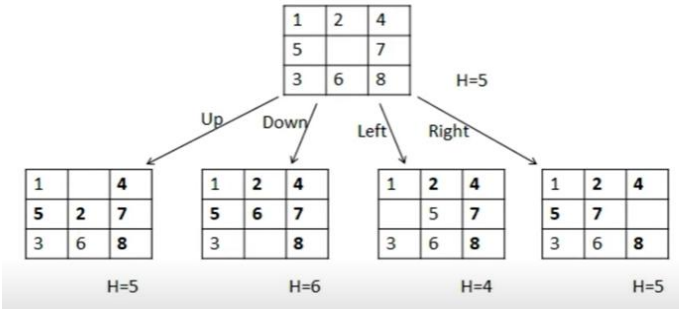
Now, if we expand the initial state, 4 movements (Up, Down, Left, Right) of the blank space are possible. So, the next step looks as follows.



1	**4**	**7**
2	**5**	**8**
3	**6**	

Goal State

After expanding the 4 movements of the blank space, we have to calculate the heuristic values of each of the state with respect to the goal state.



1	**4**	**7**
2	**5**	**8**
3	**6**	

Goal State

We are getting that

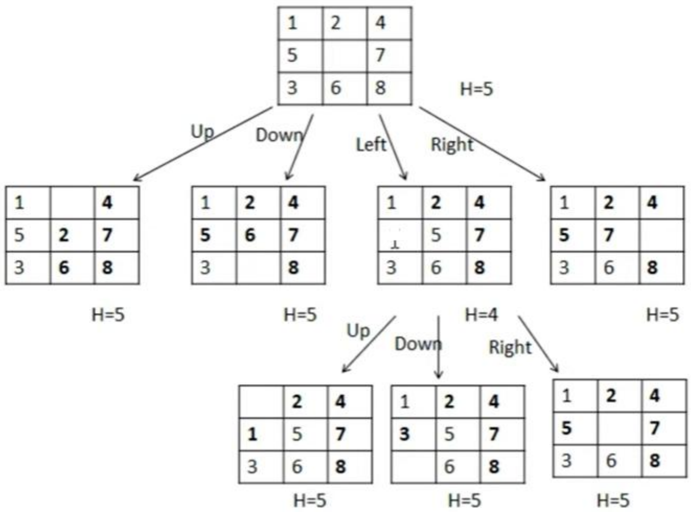
(i) if blank space moves upwards, then H= 5 as five tiles' (4, 5, 2, 7, 8) positions are changed with respect to goal state.

(ii) if blank space moves downwards, then H= 6, as six tiles' (2, 4, 5, 6, 7, 8) positions are changed with respect to goal state.

(iii) if blank space moves left side, then H= 4 as four tiles' (4, 2, 7, 8) positions are changed with respect to goal state.

(iv) if blank space moves right side, then H= 5 as five tiles' (4, 5, 2, 7, 8) positions are changed with respect to goal state.

Now, if we follow the Steepest Hill Climbing algorithm, then we can explore further where the heuristic (H) values are lesser which means it is closer to the goal state.



In the above scenario, if we follow the hill climbing algorithm, then it cannot proceed further because in the third level all state heuristic (H) values are 5 which is more than the previous level heuristic (H=4) value. It will be unable to reach the goal state. But if it follows the Steepest Hill Climbing algorithm, then the expansion of all states expands further until it reaches the goal state.

5(a) How simulated annealing provides opportunity to states that are not promising? [2 Marks]

Answer:-

Simulated Annealing (SA) is one of the most famous algorithms which achieve the incorporation of random action with greedy action. Particularly, it incorporates random walks into greedy approaches. This process is inspired by the annealing process in metallurgy, where a material is heated and then slowly cooled to reduce defects and achieve a stable structure.

Firstly, there is a high probability of choosing the non-promising states when it is heated and it chooses the states judiciously i.e., the probability of accepting worse solutions decreases when it is cooled, thus allowing the algorithm to explore a variety of states, even those that seem to be non-promising initially.

Secondly, unlike greedy algorithms that only accept better solutions, SA takes the non-promising states with some probability p = e-△E/T ; where △E = change in cost (new cost - current cost) and T = temperature (a control parameter).

Thirdly, as the hill-climbing algorithms have the tendency of getting stuck in local optima, by occasionally accepting non-promising states, SA avoids getting trapped in sub-optimal regions and has a higher chance of reaching the global optimum.

5(b) Apply genetic algorithm to maximize the below equation . For creating individuals for the population take the binary representation for the values of variable x. Starting with initial population of 5 individuals , show the selection , crossover and mutation steps. Show the process for one iteration.

[3 Marks]

$$f(x) = -\frac{x^2}{10} + 3x$$

$$1 < x < 31$$

Answer:-

$f(x) = -\frac{x^2}{10} + 3x$ where $1 < x < 31$.

Lets consider 5 diff. parent are ~~given~~ mentioned below

Initial population fitness function. selection duplicated population.

00010 = 2 2/50 = 0.04

00111 = 7 7/50 = 0.14 → 00111

01000 = 8 8/50 = 0.16 → 01000

10000 = 16 16/50 = 0.32 → 10000

10001 = 17 17/50 = 0.34 → 10001

50

Crossover: 00000, 01111, 10001, 11000

Mutation: 10000 — 16, 10111 — 31, 11001 — 25, 11000 — 24 (value of x.)

$f(x) = f(16) = -\frac{(16)^2}{10} + 3(16) = -25.6 + 48 = 22.4$

$f(31) = -\frac{(31)^2}{10} + 3(31) = -96.1 + 93 = -3.1$

$f(25) = -\frac{(25)^2}{10} + 3(25) = -22.5 + 75 = 52.5$ ← 2nd result.

$f(24) = -\frac{(24)^2}{10} + 3(24) = -57.6 + 72 = 14.4$