AI
ASSIGNMENT-1

PART-A → Conceptual Questions

Solⁿ 1 → Admissible Heuristic → A heuristic function $h(n)$ is admissible if for every node n, $0 \le h(n) \le h^*(n)$, where $h^*(n)$ is the true cost of the least-cost path from n to a goal.
> An admissible heuristic never overestimates the true remaining cost.

Ex:- for a grid-world (4 neighbour moves, unit cost):
Manhattan distance :- $h_{man}(r,c) = |r - r_{goal}| + |c - c_{goal}|$
This equals the exact minimum number of orthogonal moves required on an unobstructed grid, so it never overestimates when move cost = 1.

Ex (Non-admissible) for a grid-world:-

$h_{non}(r,c) = 1.5 \times \text{Manhattan}(r,c)$. Multiplying by 1.5 can cause the heuristic to exceed the actual remaining cost on some nodes, i.e it can overestimates → not admissible.

Solution 2 → Definition (consistent / monotone):
A heuristic h is consistent if for every edge from node n to successor n' with cost (n, n'):
$$h(n) \le c(n, n') + h(n')$$
and typically $h(goal) = 0$.

P.T.O

Relation to triangle inequality :-
This condition is equivalent to a triangle-inequality-like property :- The estimated cost from n to the goal is no greater than the cost of stepping to a neighbor n' plus the estimated cost from that neighbor to the goal. It enforces that following an edge cannot make the estimated remaining cost suddenly drop by more than the edge cost — just like how a triangle inequality bounds distances.

solution 3:- YES. Consistency implies admissibility (a consistent heuristic must be ≤ true cost to goal), but the converse is not always true. One can design a heuristic that never overestimates the true remaining cost (admissible) but violates $h(n) \leq c(n,n') + h(n')$ for some edge (i.e., there exists a local dip/jump that breaks monotonicity). That local violation means A* may need to re-open nodes (in graph-search) even though the heuristic remains optimistic overall.

solution 4:- Admissibility necessary (for optimality in tree-search)
Admissibility ensures that $f(n) = g(n) + h(n)$ is a lower bound on the true cost of any solution path that goes through n. If h can overestimate, A* might prune or prefer a path whose f looks better but actually cannot reach the real optimum — so A* can return a suboptimal goal.

PART – B (Worked Examples (Paper/Pencil))

Given graph:

A --2--> B --2--> C (goal)

A --5--> C

costs shown on edges. $h(C) = 0$ always.

Soln: To show with $h(A) = 3$, $h(B) = 1$, $h(C) = 0$ → A* finds the optimal path..

Start: only A in OPEN with $g(A) = 0$

* $f(A) = g(A) + h(A) = 0 + 3 = 3$. Pop A (expand)

Expanding A:

* Successor B: $g(B) = g(A) + cost(A,B) = 0 + 2 = 2$.
  $f(B) = g(B) + h(B) = 2 + 1 = 3$.

* Successor C (direct): $g(C) = g(A) + cost(A,C) = 0 + 5 = 5$. $f(C) = 5 + 0 = 5$.

OPEN now: B ($f=3$), C ($f=5$). pop the smallest f: B ($f=3$).

Expand B:

* Successor C: tentative $g = g(B) + cost(B,C) = 2 + 2 = 4$. $f(C) = 4 + 0 = 4$. This improves C (from $g = 5$ to $g = 4$).

OPEN now: C ($f=4$). Pop C which is goal. Path recovered: A → B → C with total cost 4 – optimal (since $4 < 5$). A* succeeded.

Solution 2) with $h(A) = 5$, $h(B) = 5$, $h(C) = 0$,
to show step-by-step that A^* fails to find the optimal path.

Start:

* $f(A) = 0+5 = 5$. Pop A

Expand A:

- To B: $g(B) = 2 \rightarrow f(B) = 2+5 = 7$
- To C: $g(C) = 5 \rightarrow f(C) = 5+0 = 5$.

OPEN now: $C(f=5)$, $B(f=7)$. Pop the smallest f: C (the goal!) and return $A \rightarrow C$ with cost 5. But the true optimal path is $A \rightarrow B \rightarrow C$ with cost 4. Because h overestimated and biased f-values, A^* selected the suboptimal direct path to the goal and terminated — so A^* failed here due to non-admissible heuristic.

Solution 3) Consistency tests (same graph):
we check $h(n) \leq c(n, n') + h(n')$ for each edge.

Case 1) $h(A) = 3$, $h(B) = 1$, $h(C) = 0$

* $A \rightarrow B$: $3 \leq 2+1 = 3$ ✓
* $B \rightarrow C$: $1 \leq 2+0 = 2$ ✓
* $A \rightarrow C$: $3 \leq 5+0 = 5$ ✓

All Satisfied $\longrightarrow$ Consistent case2 → $h(A) = 4$, $h(B) = 5$, $h(C) = 0$

* $A \to B$: $4 \leq 2+5 = 7$ ✓
* $B \to C$: $5 \leq 2+0 = 2$ ✗ (violated)
* $A \to C$: $4 \leq 5+0 = 5$

So case 2 violates consistency because $h(B) > 2 + h(C)$.

PART-C

PART-D → Critical thinking (Write-up / Examples).

Soln 1 ⇒ Why A* is guaranteed optimal if both admissibility and consistency hold

- Admissibility ensures that every node's $f(n) = g(n) + h(n)$ is a lower bound on the true ~~of~~ cost of any solution going through that node. So A* never abandons a path that may lead to a cheaper solution than the first goal it finds (in tree-search).

- Consistency ensures monotonicity (f never decreases along a path). Therefore when a node is expanded (popped from OPEN), we already have the minimal g-value for that node → it will not be improved later. This guarantees the first time a goal is popped, its g is the lowest possible → optimal. In graph-search, consistency avoids needing to re-open nodes, preserving both correctness and efficiency.

Que 2

M

9	8	7	6	5	4	3	2	1	G
10	#	#	#	#	#	#	#	2	#
11	#	#	Q	#	#	#	#	3	#
12	#	#	9	8	7	R #	#	4	#
13	#	#	10	#	8	#	#	5	#
14	#	#	11	#	9	#	#	6	#
15	#	#	12	#	10	9	8	7	#
16	#	#	13	#	X	#	#	Y Z	#
S	16	15	14	P	#	#	#	#	#

We have two paths S, M, G and S, P, Q, R, X, Y, Z we make first one admissible

In path SMG Heuristic distance is 16

now we made non-admissible = 3×16 → 48

So, Here we select the second path SPQRXYZ whose actual step is 22 that is longer the actual distance of the first one SMG (16) steps.

Here we have to select Non-optimal path because of non-admissible of the S,M,G

PRAGATI
Page:
Date:

Que 2

→D

	9	8	7	6	5	4	3	2	1	G
C ←	20	#	#	#	#	#	4	#	#	
	11	#	→N	#	#	#	5	#	#	
	12	#	10	9	8 →	O #	6	#	#	
	13	#	11	#	9	#	7	#	#	
	14	#	12	#	10	#	8	#	#	
	15	#	13	#	11	10	9 →	Q	#	
	16	#	14	#	P	#	#	#	#	
	17	#	15	#	#	#	#	#	#	
	S	17	16	→M	#	#	#	#	#	

⇒ We have two paths SCDG and SMNOPQG
we selected the first one randomly, it calls C
we checked consistency

Condition: $h(C) \leq c(C, D) + h(D)$

$\Rightarrow 20 \leq 1 + 9$

$20 \leq 10$ That is false

So, it is inconsistent
we choose the second part the actual path is 24 steps that is longer than the actual path (17) steps of first one SCDG.

So, we have to choose the non optimal path because of the inconsistency of the first path that is actually shortest one.

Ques 4

#	#	#	#	#	#	3	2	1	G
#	#	#	#	#	#	4	#	#	#
#	#	#	#	#	#	5	#	#	#
#	#	#	#	#	#	6 →	R	#	#
#	#	#	#	#	#	4 →	Q	#	#
#	#	A		#	#	8 →	P	#	#
#	#	13	12	11	10	9	#	#	#
#	#	14	#	#	#	#	#	#	#
#	#	15	#	#	#	#	#	#	#
S	17	16	#	#	#	#	#	#	#

Here $h(Q) = 4$ and actual path is 7

$\Rightarrow h(Q^*) = 7$

$h(Q) \leq h(Q^*)$ that is admissible

now for consistency

$h(Q) \leq c(Q,R) + h(R)$

$4 \leq 1 + 6$

$4 \leq 7$ that is false means it is inconsistent.

So, ~~He~~ Here we got the one path that is admissible but inconsistent.

classmate
Date
Page

Short conclusion paragraph :-

Admissibility keeps heuristics optimistic so f-values remain lower bounds on true costs; consistency enforces a triangle-inequality-like monotonicity that prevents f from decreasing along paths. Together they guarantee that A* expands nodes in a an order that yields optimally and that nodes need not be re-opened. Violating admissibility risks wrong answers (suboptimal goals). Violating consistency typically preserves admissibility but may force re-expansions and inefficient searches (and can break graph-search guarantees)