COMP90054 AI Planning for Autonomy

THE UNIVERSITY OF MELBOURNE · CLOSED-BOOK MEMORY SHEET

Full guide + ask Sia about COMP90054

SIDE 1 · RECALL — **Priority:** definitions → equations → choice conditions → limits — **NOTHING PRINTED ENTERS THE EXAM**

01 · Exam map

COURSE · BRIEF

IPA 25 | Q 10 | IMA 15 | FINAL 50

ASSESSMENT	WT	FORMAT
Individual Programming Assignment	25%	Week 6; ≈30 h
Online Quiz	10%	30 min; Week 7
Individual Modelling Assignment	15%	Weeks 11–12; ≈15 h
Closed-book Final Examination	50%	2 h; **must be passed**

"One closed book end-of-semester examination." *COMP90054 2026 Handbook assessment table (FETCHED_UNIMELB_2026_COMP90054_ASSESSMENT_DOM.txt)*

02 · Model before solving

C1

Autonomous system: active system that perceives, reasons and acts towards goals or rewards. A ***state*** contains information treated as relevant to future action choice. A ***planning model*** specifies actions, preconditions, effects, initial conditions and goals.

DETERMINISTIC TRANSITION

```
s' = γ(s,a)
Valid current state + action → one successor state.
```

TRANSLATION CHECKLIST

1. Declare state variables and observability.
2. List each action's preconditions and effects.
3. Write the initial state and goal.
4. Represent safety constraints explicitly.
5. Trace ordinary, boundary and invalid inputs.

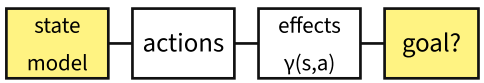
EXAM CUE *A planning bug may be a missing state variable. Omitted variables, wrong observability and abstraction choices can make a formally solved model unsafe.*

STATE/ACTION SANITY AUDIT

QUESTION	EVIDENCE IN THE ANSWER
What can change?	State variables have domains; every action effect changes named variables
When is a move legal?	Preconditions exclude invalid state–action pairs
What counts as success?	Goal condition is testable against a state
What can the system know?	Observability is stated rather than assumed
What must never occur?	Safety is a constraint or safely handled transition, not an afterthought

Minimal complete explanation: "From state s, action a is applicable because its preconditions hold; γ produces s' by its effects; s' is then tested against the goal and safety conditions." If the real task contains uncertainty, separate the known nominal plan from uncertain execution and say whether replanning or an uncertainty-aware policy is needed.

Model quality comes before algorithm choice. A faster solver does not recover variables, consequences or failure states omitted from the representation.



Source-derived C1 route: representation controls every later guarantee.

03 · Uninformed search

C2

A *search node* stores a state plus its path, cost and frontier information. The *frontier* is generated nodes awaiting expansion. Match its ordering to the objective.

PATH COST

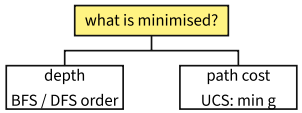
$g(n) = \Sigma_{i=1}^{k} c(a_i)$

METHOD	FRONTIER RULE	USE / QUALIFICATION
BFS	Shallowest first	Minimum depth; minimum cost only when step costs equal
DFS	Deepest first	Depth order does not optimise total cost
UCS	Lowest g(n)	Minimum total cost under non-negative costs; keep best-known repeated-state cost

GUARANTEE AUDIT

- Name the objective: depth or cost.
- State frontier priority and tie handling.
- For graph search, say how duplicates are handled.
- Qualify completeness/optimality by branching and cost assumptions.

IF YOU SEE *varying edge costs + minimum total cost → UCS, not BFS.*



C2 decision rule: frontier priority must express the promised objective.

05 · A-star

C3

EVALUATION

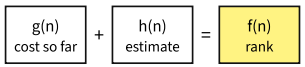
```
f(n) = g(n) + h(n)
g = incurred path cost; h =
estimated remaining cost.
```

Admissible: h(n) never overestimates true optimal remaining cost. ***Consistent***: heuristic change across an edge does not exceed that edge cost.

CONSISTENCY TEST

$h(n) \le c(n,a,n') + h(n')$

CLAIM	WHAT TO WRITE
Next expansion	Compute every frontier f; choose minimum f, not minimum h
Optimality	Name heuristic property and graph-search handling conditions



C3 score anatomy; audit h before attaching an optimality claim.

07 · Delete relaxation

C4

Delete relaxation ignores action delete effects: achieved facts persist. A ***relaxed plan*** solves the easier relaxed problem; a ***helpful action*** comes from an early relaxed-plan layer as promising guidance.

FF-STYLE ESTIMATE

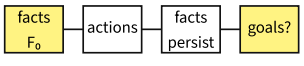
$hFF(s) = |RP(s,G)|$

RELAXED PLANNING GRAPH

1. Start from facts true in state s.
2. Add applicable actions at the next action layer.
3. Add their positive effects; ignored deletes do not remove facts.
4. Continue until goals appear or no new facts arise.
5. Extract a relaxed plan backwards; count its actions.

USEFUL BECAUSE	BLIND BECAUSE
Easier reachability gives search guidance	Resource conflict and negative interaction can disappear
Extracted actions suggest promising successors	Low hFF is **not** an executable real plan

TRAP *Restore deletes before claiming feasibility, ordering or resource reuse.*



C4 flow: ignored deletes make fact layers monotone, not the real world.

08 · Planning answer boundary

C1 · C4

Always separate **model validity**, **heuristic estimate** and **plan executability**. An estimate can rank states well while its extracted relaxed sequence fails after real delete effects return.

PROMPT VERB	REQUIRED OUTPUT
Construct	Show alternating fact/action layers and applicability
Extract	Select actions supporting goals backwards through layers
Estimate	Count extracted relaxed actions: hFF
Guide	Use relaxed information to order real search
Validate	Restore deletes and check ordering/resources
Helpful	Use early relaxed-plan actions as guidance, not proof
Qualify	Ignored deletes can hide resource conflict and subgoal interaction; a low estimate is not executable

09 · MDP + Bellman

C5

An *MDP* has states, actions, transition probabilities, rewards and discounting. A *policy* maps states or observations to actions. Bellman reasoning combines immediate reward with discounted continuation value.

BELLMAN OPTIMALITY

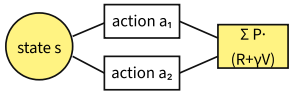
$V^*(s) = \max_a \Sigma s' P(s'|s,a) [R(s,a,s') + \gamma V^*(s')]$

COMPUTE IN THIS ORDER

1. For each action, enumerate successors s'.
2. Inside each successor term add reward + γ continuation.
3. Weight by P(s'|s,a) and sum.
4. Only then maximise over actions.

SYMBOL	ROLE	COMMON ERROR
P	Transition uncertainty	Taking max over successors instead of probability-weighted sum
R	Immediate encoded objective	Assuming it contains unmodelled safety costs
γ	Discount	Applying it to the immediate reward
$\max_a$	Optimal action choice	Maximising before the expectation

LIMIT *The Markov property, reward design, γ and transition model are assumptions. Correct arithmetic cannot repair a bad reward.*



C5 Bellman order: expectation within each action, then maximum.

10 · Value vocabulary

C5

Return is realised discounted reward; **value** is an expectation under a model or policy; **optimal value** uses the best available continuation. Keep sample, estimate and expectation distinct.

QUANTITY	QUESTION ANSWERED
$V^\pi(s)$	Expected return from s while following π
$Q^\pi(s,a)$	Expected return after taking a in s, then following π
$V^*(s)$	Best expected return achievable from s
$\pi(s)$	Action selected at state or observation s

BELLMAN ANSWER AUDIT

11 · MCTS / UCT

C6

MCTS repeats **selection → expansion → simulation → backup**. Each cycle converts a rollout into updated visit counts and return estimates.

UCT SELECTION

$UCT_i = \bar{X}_i + c\sqrt{(\ln N / n_i)}$

$\bar{X}_i$ = exploitation; bonus = exploration.

TERM CHANGES	EFFECT
Higher $\bar{X}_j$	Raises evidence-based preference
Lower child visits n_j	Raises exploration bonus
Higher parent visits N	Raises pressure to revisit underexplored children
Higher c	Weights exploration more strongly

FOUR-STAGE ANSWER

1. Select recursively by UCT.
2. Expand an available child.
3. Simulate with the stated rollout policy.
4. Back up sampled return through visited nodes.

QUALIFICATION *Finite rollouts, weak simulation policies and variance can make the current leader unreliable.*



C6 cycle: the answer must connect selection score to backed-up evidence.

12 · PUCT guidance

C10

PUCT-STYLE SCORE

$Score(s,a) = Q(s,a) + cP(s,a)\sqrt{N(s)/(1+N(s,a))}$

A strong prior can lift a little-visited action; additional child visits shrink its bonus. Learned policy and value guidance save search only if model errors remain discoverable under the compute budget.

EVIDENCE SOURCE	ROLE	FAILURE CHECK
Q(s,a)	Accumulated search evidence	Few or noisy rollouts
P(s,a)	Learned policy prior	Distribution shift or blind spot
Value network	Leaf outcome estimate	Calibration and objective mismatch
Visit counts	Allocate future search	Compute concentration
Self-play	Generate learning experience	Narrow coverage or changing opponents

13 · MC, TD and control

C7 · C8

MONTE CARLO RETURN

$G_t = \Sigma_{k=0}^{T-t-1} \gamma^k R_{t+k+1}$

MC waits for an episodic sampled return. ***TD learning*** updates before episode end from reward plus a bootstrapped next estimate.

SARSA · ON-POLICY

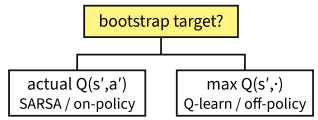
$Q(s,a) \leftarrow Q(s,a) + \alpha[r + \gamma Q(s',a') - Q(s,a)]$

a' is the action actually selected.

Q-LEARNING · OFF-POLICY

$Q(s,a) \leftarrow Q(s,a) + \alpha[r + \gamma \max_{a'} Q(s',a') - Q(s,a)]$

SEE	DO
Full episode rewards	Discount by powers of γ; distinguish one G_t sample from expected value
Behaviour next action given	SARSA target uses that a'
Greedy bootstrap requested	Q-learning target uses max next Q
Update arithmetic	Write **target → error → α step → new Q**



C8 split: the next-state bootstrap, not the outer update, distinguishes them.

14 · Policy gradient + control

C9 · C11

POLICY-GRADIENT IDENTITY

$\nabla\theta J(\theta) = E[\nabla\theta \log \pi\theta(a|s) A\pi(s,a)]$

Actor selects actions; **critic** estimates value or advantage. Positive advantage drives ascent to increase the sampled action's log probability locally. Variance, critic bias and unsafe exploration remain.

BEST-RESPONSE CONDITION

$u_i(a_i^*, a_{-i}^*) \ge u_i(a_i, a_{-i}^*)$ for every a_i

Equilibrium or reward optimality does not imply fairness, safety, legality or social desirability. Add stakeholder costs, hard constraints, monitoring, appeal and human intervention.

BANDIT → PREDICTION → CONTROL

A ***multi-armed bandit*** balances uncertain rewards without state transitions. An ***ε-greedy*** policy chooses a best action most of the time and a random action with probability ε. Monte Carlo prediction uses complete sampled returns; TD control bootstraps the unknown remainder.

CHOICE BOUNDARY	QUESTION TO ASK
Bandit vs sequential	Do actions change a state that affects future rewards?

STABILITY AND ETHICS AUDIT

COMP90054

Self-test drills · answers must show the path

AI PLANNING FOR AUTONOMY · THE UNIVERSITY OF MELBOURNE

Full guide + ask Sia about COMP90054

SIDE 2 · DRILL — **Protocol:** cover the answer → compute aloud → reveal → explain the limit — **4 WORKED PATHS + JUDGEMENT CARDS**

15 · Worked: A-star order — C3 WORKED

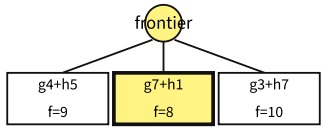
Q. Frontier nodes have (g,h): `(4,5)`, `(7,1)`, `(3,7)`. Which expands first?

NODE	G+H	F
1	4+5	9
2	7+1	8
3	3+7	10

ANSWER PATH

`Compute all f → compare 9, 8, 10 → expand (7,1).`
Then qualify: `ordering is not by h alone; final optimality still depends on heuristic and graph-search conditions.`

SELF-CHECK *Can you state admissibility and consistency without looking?*



C3 worked figure: minimum f expands; yellow marks the chosen node.

16 · Card: search choice — C2 WORKED

Q. A graph has unit-depth distractors, widely varying edge costs, and the objective is minimum total cost.

1. Reject depth as the objective.
2. Use a priority frontier ordered by g.
3. For repeats, keep the best-known cost.
4. State non-negative-cost and finite-branching conditions.

Answer: uniform-cost search. BFS agrees only when step costs are equal.

CHANGED-CASE PROBES

CHANGE	WHAT CHANGES IN THE ANSWER?
All step costs equal	BFS and UCS can return the same minimum-cost result; their frontier keys remain different
Objective becomes fewest actions	Depth, not total cost, controls the choice
Repeated state found more cheaply	Update best-known cost and frontier treatment
Goal unreachable	Termination/completeness needs finite branching and duplicate handling

Oral trace: say which node leaves the frontier, which successors enter, their new path costs, how repeats are treated, and the resulting order. If you cannot name the next frontier, you have not yet demonstrated the algorithm.

HEURISTIC PROPERTY SELF-TEST

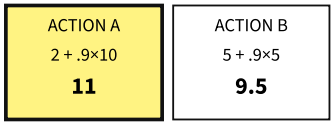
17 · Worked: Bellman choice — C5 WORKED

Q. At state s, A gives reward 2 then value 10; B gives reward 5 then value 5. γ=0.9; transitions are certain.

A
`2 + 0.9×10 =` **`11`**

B
`5 + 0.9×5 =` **`9.5`**

Choose A under this model. The conclusion is conditional on state, reward and transition representation; inspect whether rewards encode all safety costs.



C5 worked figure: compare complete action values, then audit the model.

18 · Worked: MC return — C7 WORKED

Q. From t, rewards are 3, 2, 4 and γ=0.5.

ANSWER PATH

$G_t = 3 + 0.5\times2 + 0.25\times4$
$= 3 + 1 + 1 =$ **5**

This is one observed episodic return under a policy, **not the value itself**. More samples are needed to estimate expected value; later rewards carry increasing powers of γ.

19 · Return vs value — C7

OBJECT	MEANING
G_t	One realised discounted sum
$V^\pi(s)$	Expected return from s under π
$Q^\pi(s,a)$	Expected return after a, then π
Sample mean	Estimate; sensitive to policy, coverage, rare events and non-stationarity

CHANGE ONE ASSUMPTION

If γ changes, recompute each reward's power; do not reuse the old return. If transitions become uncertain, replace the certain Bellman continuation with a probability-weighted sum. If a reward is terminal, follow the task's terminal handling rather than bootstrapping blindly. If a safety cost is absent, the larger numeric value is only the winner of the incomplete model.

Policy boundary: every MC sample is generated under a policy. State whose behaviour produced the episode before generalising from its return. Adequate coverage matters because rare but costly outcomes can be missed.

BELLMAN SCRATCHPAD DISCIPLINE

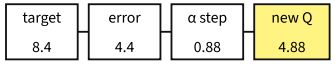
LINE	WRITE BEFORE CALCULATING
1	One expression per action

20 · Worked: Q-learning — C8 WORKED

Q. Q(s,a)=4, r=3, γ=0.9, max next Q=6, α=0.2.

STAGE	CALCULATION	RESULT
Target	3+0.9×6	8.4
Error	8.4−4	4.4
Step	0.2×4.4	0.88
Update	4+0.88	**4.88**

RECITE *target → error → step → new value. This arithmetic alone does not prove convergence.*



C8 worked update; separating stages prevents sign and target errors.

21 · Card: UCT judgement — C6 WORKED

Q. A has mean 0.7 from 40 visits; B has mean 0.5 from 5 visits, under the same parent. No c or parent visit count is supplied.

1. Write exploitation + exploration terms.
2. B has the larger exploration bonus because nB is smaller.
3. Do **not** invent a numeric winner: it depends on c and N.
4. Report sensitivity to visit counts and rollout variance.

UCT *can* select B despite its lower mean. Limited visits make its uncertainty bonus larger.

22 · Card: advantage sign — C9 WORKED

Positive estimated advantage → weight the sampled log-policy gradient positively → gradient ascent increases that action's probability locally. Retain learning-rate, variance and safety controls; critic evidence is not ethical approval.

TARGET SUBSTITUTION TEST

Take the Q-learning numbers above but suppose the behaviour policy actually selects a next action valued below the maximum. The Q-learning target remains greedy; the SARSA target substitutes the selected action's Q. Do not compute a numeric SARSA update unless that selected value is supplied.

PROMPT WORDING	TARGET EVIDENCE
"next action taken"	Use Q(s′,a′): SARSA
"best/max next action"	Use max Q(s′,·): Q-learning
"positive advantage"	Increase sampled log probability locally
"strong prior, few visits"	PUCT exploration may select it

UCT / PUCT RETRIEVAL STRIP

23 · Worked: relaxed plan — C4 WORKED

Q. A relaxed solution uses four actions, but two require a reusable tool that the real model consumes.

1. Report relaxed-plan length **4** as guidance.
2. Restore the tool's delete effect.
3. Recheck ordering and re-acquisition.
4. Do not execute the relaxed sequence without validation.

Answer: hFF may be 4 while the real plan needs extra actions or is infeasible. The ignored delete creates resource interaction.



C4 counter-case: delete effects can reverse the feasibility conclusion.

24 · Card: parcel robot — C1 WORKED

Scenario. Robot moves among three rooms, carries one parcel and must deliver it without entering a blocked room.

1 MARK EACH	REQUIRED MOVE
Representation	Robot location, parcel location/carry state, blocked status
Actions	Move, pick, drop with explicit preconditions/effects
Boundary	Initial state and goal
Safety	Blocked-room entry prohibited or safely handled

TRAP *Do not bolt safety on after finding the nominal route.*

25 · Method chooser — C1–C10

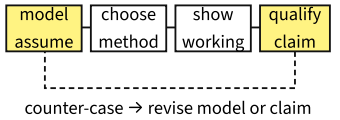
IF THE TASK GIVES…	START WITH…
Known deterministic actions/goals	Classical planning/search
Costs + heuristic	A-star; audit h
Transition probabilities/rewards	MDP/Bellman
Simulator + planning budget	MCTS/UCT
Episodic samples	MC prediction
Bootstrapped control samples	SARSA or Q-learning
Differentiable parameterised policy	Policy gradient/actor-critic

INTEGRATED DELIVERY-SYSTEM FRAME

Answer route: Translate the task into states, actions, effects and goals. Under uncertainty, calculate values and select a policy; with a simulator, apply UCT. Finish by checking model and stakeholder controls.

26 · Judgement cards — C2–C10

IF YOU SEE X	DO Y
Minimum depth	Use a depth-ordered frontier; do not claim minimum weighted cost
Minimum cost, unequal edges	Use UCS / lowest g
A-star frontier	Compute every g+h; expose ties
Heuristic guarantee	Test admissibility/consistency, not usefulness by intuition
Relaxed plan	Use to guide; restore deletes before execution claims
Stochastic successors	Probability-weight then sum before max
Low child visit count	Expect larger UCT/PUCT exploration term
Sampled episode	Compute G; do not call one sample the value
Actual next action	SARSA target
Greedy next estimate	Q-learning target
Positive advantage	Increase sampled action's probability locally
Stable strategic profile	Test external stakeholder harm anyway



Integrated answer route: model, method, working, boundary, then counter-case.

27 · Classic traps — C1–C11

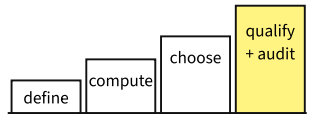
- **State ≠ node:** a node also carries path/search information.
- **h ≠ f:** A-star ranks g+h.
- **Relaxed ≠ executable:** ignored deletes matter.
- **Sum before max:** stochastic Bellman backup.
- **Return ≠ value:** sample versus expectation.
- **SARSA ≠ Q-learning:** actual action versus maximum.
- **Convergence ≠ safety:** performance claims stay inside model assumptions.
- **Equilibrium ≠ acceptability:** incentives do not certify objectives.

FIVE-SENTENCE ANSWER SKELETON

1. **Model:** name state, actions, objective and uncertainty.
2. **Method:** select the rule matching that model.
3. **Working:** show frontier order, expectation, return or update.
4. **Decision:** compare candidates and answer the actual question.
5. **Boundary:** state the assumption or failure mode that could change it.

28 · Fast-recall matrix — ALL CHAPTERS

METHOD	CORE OBJECT	MUST QUALIFY
Planning	s′=γ(s,a)	State completeness; observability
UCS	min g	Costs; duplicates; branching
A-star	min g+h	Admissibility/consistency
hFF	|relaxed plan|	Deletes/resources/interactions
Bellman	ΣP(R+γV), max a	Markov/reward/model
UCT	mean + visit bonus	Budget/policy/variance
MC	discounted episode return	Coverage/non-stationarity
SARSA	r+γQ(s′,a′)	On-policy behaviour
Q-learning	r+γmax Q(s′,·)	Exploration/stability
Policy gradient	log-policy gradient × A	Variance/bias/locality
PUCT	Q + prior-weighted bonus	Prior error/compute
Best response	utility under others' actions	Stakeholders and constraints



Recall ladder: a complete response ends with the claim boundary.

29 · Exam-morning plan — COURSE STRATEGY

1. **Retrieve:** write all core equations from memory.
2. **Discriminate:** explain BFS/UCS/A-star; MC/TD; SARSA/Q-learning; UCT/PUCT.
3. **Calculate:** redo the four numeric drills without prompts.
4. **Transfer:** change one assumption in each and revise the conclusion.
5. **Audit:** state one modelling, compute, safety or stakeholder limit per method.

FINAL ORAL TEST *For any scenario: formalise → select → compute → compare → qualify. If one verb is missing, the answer is incomplete.*

30 · Ethical-control close — C11 WORKED

Routing-game drill: define parties/actions/utilities → test unilateral deviations → add external stakeholder costs and hard constraints → assign monitoring, appeal and intervention responsibility. A stable fleet equilibrium may still externalise unacceptable harm.

TECHNICAL RESULT	QUESTION STILL OPEN
Search found a plan	Were relevant hazards represented?