IAE ACADEMY INSTITUTE FOR ENGINEERS

CORE JOBS · INTERVIEW QUESTIONS

IAE Academy free study material · Updated 30 October 2026

Artificial Intelligence Interview Questions

INSIDE

- 22 questions with short answers
- 6 numericals with the working
- Grouped by topic

Companies do not publish their interview questions; these are ours.

Online, with links to the notes: **iaeacademy.in/study/artificial-intelligence/interview-questions**

FREE COUNSELLING

Preparing for GATE or a core job? Talk to IAE Academy.

Or call **+91 84204 36524**

Every day, 10:30 am to 7:30 pm

Chat on WhatsApp

Free. Not for sale. Pass it on.

Search

1. How does breadth-first search work, and when is it complete and optimal?

Breadth-first search expands the shallowest unexpanded node first, using a FIFO queue. It is complete when the branching factor is finite, and optimal when every step costs the same. Its time and memory both grow as $O(b^d)$ for a goal at depth d.

2. What is the difference between depth-first and breadth-first search in memory use and completeness?

Depth-first search keeps only the current path and the unexpanded siblings along it, about $O(bm)$ memory for branching factor b and maximum depth m, while breadth-first search stores a whole level, $O(b^d)$ for a shallowest goal at depth d. Breadth-first search is complete when b is finite. Depth-first search is not: it can descend an infinite path forever, and in a finite space with cycles it loops unless repeated states are checked.

3. Why does iterative deepening search combine the strengths of depth-first and breadth-first search?

It runs depth-limited depth-first search with limits 0, 1, 2 and so on. It needs only $O(bd)$ memory like depth-first search, yet it finds the shallowest goal like breadth-first search when step costs are equal. The repeated work is small because most nodes lie at the deepest level.

4. Why is A* search optimal when its heuristic is admissible?

An admissible heuristic never overestimates the true remaining cost, so $f = g + h$ never exceeds the cost of the best solution through a node. When a goal is chosen for expansion, no frontier node can lead to a cheaper goal. This argument is for tree search. Graph search needs consistency as well.

5. What is the difference between greedy best-first search and A* search?

Greedy best-first search orders the frontier by the heuristic $h(n)$ alone, so it chases the node that looks closest to the goal. A* orders by $g(n) + h(n)$, adding the cost already paid. Greedy search is fast but neither optimal nor always complete. A* is optimal with an admissible heuristic.

6. What is a consistent heuristic, and how does it relate to an admissible one?

A heuristic is consistent if $h(n) \le c(n, n') + h(n')$ for every step from n to a successor n', a triangle inequality, and $h = 0$ at goals. Every consistent heuristic is admissible, but not the reverse. With consistency, f never decreases along a path, so A* graph search expands each node once at its best cost.

7. A MAX root has three MIN children. Their leaves, read left to right, are (3, 12, 8), (2, 4, 6) and (14, 5, 2). Alpha-beta pruning visits children and leaves left to right. How many leaves are evaluated, and what is the root value?

Alpha-beta evaluates 7 of the 9 leaves, and the root value is 3, the same as plain minimax gives. Once the second MIN node reads 2, which is no better than the 3 MAX already has, its other two leaves cannot change the result.

WORKING

1. First MIN node reads 3, 12, 8 and returns 3, so $\alpha = 3$ at the root.
2. Second MIN node reads 2. Since $2 \leq \alpha = 3$, MAX will never choose it, so 4 and 6 are pruned: 1 leaf read.
3. Third MIN node reads 14, 5, 2. Its value only drops to 2 at the last leaf, so all 3 are read.
4. Root value $= \max(3, 2, 2) = 3$; leaves read $= 3 + 1 + 3 = 7$.

Propositional and predicate logic

8. What is the difference between a tautology, a contradiction and a satisfiable formula?

A tautology is true under every assignment, and a contradiction is false under every assignment. A satisfiable formula is true under at least one assignment, so every tautology is satisfiable and no contradiction is. A formula is a tautology exactly when its negation is a contradiction.

9. What does it mean for a knowledge base to entail a sentence?

A knowledge base KB entails a sentence α, written $KB \models \alpha$, when every assignment that makes all of KB true also makes α true. Equivalently, $KB \wedge \neg\alpha$ is unsatisfiable. This is a statement about meaning, which is separate from whether a proof procedure finds a derivation.

10. Why does proof by resolution add the negation of the goal to the clauses?

Resolution is a refutation method. It shows $KB \models \alpha$ by showing that $KB \wedge \neg\alpha$ is unsatisfiable, which is reached when the empty clause is derived. All sentences must first be in clause (CNF) form. Resolution is sound, and refutation-complete for propositional logic.

11. Over the three propositional variables p, q and r, in how many of the 8 truth assignments is $(p \vee q) \rightarrow r$ true?

The formula is true in 5 of the 8 assignments, because only 3 assignments make it false: those where $p \vee q$ is true and r is false. An implication is false only in that case.

WORKING

1. An implication is false only when its premise is true and its conclusion is false.
2. Premise $p \vee q$ is true for 3 of the 4 settings of (p, q).
3. Conclusion r is false in 1 way, so the formula is false in $3 \times 1 = 3$ rows.
4. True rows $= 8 - 3 = 5$.

12. Why is "every student likes some course" written with an implication after the universal quantifier and a conjunction after the existential one?

The sentence is $\forall x\,(Student(x) \rightarrow \exists y\,(Course(y) \wedge Likes(x, y)))$. The implication limits the claim to students, since a conjunction after $\forall$ would say everything is a student. The conjunction after $\exists$ asserts that such a course exists, while an implication there would be satisfied by anything that is not a course.

13. What is unification in first-order logic, and why is the occurs check needed?

Unification finds a substitution that makes two expressions identical, ideally the most general one. For $P(x, f(y))$ and $P(a, f(z))$ it gives $\{x/a, y/z\}$. The occurs check rejects binding a variable to a term that

contains it, such as x with $f(x)$, which would create an infinite term.

14. Four propositional variables a, b, c and d give 16 truth assignments. How many of them satisfy the clause $a \vee \neg b \vee c$?

The clause is satisfied by 14 of the 16 assignments, since only 2 assignments make it false. A disjunction fails only when every literal is false, which fixes a false, b true and c false, and leaves d free.

WORKING

1. A disjunction is false only when every literal is false.
2. That needs a false, b true and c false: 1 setting of (a, b, c).
3. Variable d is free, so there are 2 falsifying assignments.
4. Satisfying assignments $= 16 - 2 = 14$.

Reasoning under uncertainty

15. What is a Bayesian network, and how does it give the joint distribution?

A Bayesian network is a directed acyclic graph with one node per random variable and a conditional table $P(X_i \mid Parents(X_i))$ at each node. The joint distribution is the product of these tables: $P(x_1, \ldots, x_n) = \prod_i P(x_i \mid parents(X_i))$. When each node has few parents, this stores far fewer numbers than the full joint table.

16. Boolean variables A and B have no parents, C has parents A and B, and D has parent C. How many independent probabilities does this Bayesian network need, and how many does the full joint table need?

The network needs 8 independent probabilities, against 15 for the full joint table over four Boolean variables. Each Boolean node needs one number per setting of its parents, 2^k for k Boolean parents, so the network stores $1 + 1 + 4 + 2$.

WORKING

1. A Boolean node with k Boolean parents needs 2^k probabilities.
2. A: 1, B: 1, C: $2^2 = 4$, D: $2^1 = 2$.
3. Network total $= 1 + 1 + 4 + 2 = 8$.
4. Full joint $= 2^4 - 1 = 15$.

17. How do chain, fork and collider structures differ in d-separation?

In a chain $A \to B \to C$ and a fork $A \leftarrow B \to C$, the path is blocked when B is observed. A collider $A \to B \leftarrow C$ is the reverse: the path is blocked while neither B nor any descendant of B is observed, and observing either opens it.

18. Why is variable elimination cheaper than inference by enumeration?

Enumeration sums the full joint product for every combination of the hidden variables, repeating the same sub-products many times. Variable elimination sums out one variable at a time and stores the result as a factor that is reused. Its cost depends on the elimination order, and a poor order can still be expensive.

19. What is the difference between rejection sampling and likelihood weighting?

Rejection sampling draws full samples from the prior and throws away those that disagree with the evidence, which wastes many samples when the evidence is unlikely. Likelihood weighting fixes the evidence variables, samples only the others, and weights each sample by the probability of the evidence given its parents.

20. How does Gibbs sampling estimate a posterior probability?

Gibbs sampling fixes the evidence variables and starts from an arbitrary state. It repeatedly resamples one non-evidence variable at a time from its distribution given its Markov blanket. In the long run the states visited follow the posterior, so the fraction of states with the query value estimates it. Successive samples are correlated.

21. Rejection sampling draws 1000 samples from a network. The evidence E is true in 200 of them, and the query variable X is true in 150 of those 200. What is the estimate of $P(X \mid E)$?

The estimate of $P(X \mid E)$ is 0.75, based on the 200 samples that agree with the evidence. The other 800 are rejected, and the estimate is the fraction of kept samples with X true, $150/200$.

WORKING

1. Samples that disagree with E are rejected: $1000 - 200 = 800$ discarded.
2. Only the 200 consistent samples are counted.
3. $P(X \mid E) \approx 150/200 = 0.75$.

22. For events A and B, $P(A) = 0.3$, $P(B \mid A) = 0.8$ and $P(B \mid \neg A) = 0.1$. What is $P(A \mid B)$?

By Bayes' rule, $P(A \mid B)$ is about 0.774. The denominator $P(B) = 0.31$ comes from summing $P(A \wedge B) = 0.24$ and $P(\neg A \wedge B) = 0.07$, so the probability of A rises from 0.3 to about 0.77 once B is known.

WORKING

1. $P(A \wedge B) = 0.8 \times 0.3 = 0.24$.
2. $P(\neg A \wedge B) = 0.1 \times 0.7 = 0.07$.
3. $P(B) = 0.24 + 0.07 = 0.31$.
4. $P(A \mid B) = 0.24/0.31 \approx 0.774$.

FROM IAE ACADEMY

Preparing for GATE or a core job?

IAE Academy teaches GATE in seven papers in Ballygunge, Kolkata, and online. Counselling is free: ask us about batches, fees or your preparation.

FREE COUNSELLING

Chat on WhatsApp

Call +91 84204 36524

Open every day, 10:30 am to 7:30 pm.

Scan to chat on WhatsApp

GATE COURSES

Classroom batches in Ballygunge and online batches. Fees and details are on the course pages.

Scan: courses

See the courses

FREE ON IAEACADEMY.IN

Notes and solved previous-year questions for Artificial Intelligence, topic by topic.

Scan: the notes

Open the notes

Found this useful? Send it to a friend preparing for core-job interviews.

Send to a friend on WhatsApp

IAE Academy free study material. Companies do not publish their interview questions; these are ours. Prepared and published by IAE Academy, Kolkata. Free to read and free to share, unaltered and with attribution.

IAE ACADEMY INSTITUTE FOR ENGINEERS

GATE coaching in Kolkata since 2012 · EC, EE, IN, CS and IT, DA, BT and CE · in the classroom and online

227A Rash Behari Ave, Ballygunge, Kolkata 700019

+91 84204 36524 · iaeacademy.in · Updated 30 October 2026

Get directions