Artificial Intelligence Question Bank

(Question Bank is for reference only)

Unit I: Introduction to AI & Problem-Solving Agents

Section A: Short

1. Define Artificial Intelligence and distinguish between Human-level AI and Machine-level AI.
2. What are the two main approaches to AI: symbolic and connectionist?
3. Give a timeline of *any three* key milestones in AI history.
4. Define an intelligent agent. Provide two examples from daily life.
5. Explain the term *performance measure* with one illustration.
6. Differentiate between rationality and perfection in AI.
7. List the four essential components of problem formulation.
8. Differentiate between Breadth-First Search (BFS) and Depth-First Search (DFS).
9. What do you mean by “uninformed search”? Why is it also called “blind search”?
10. What are the drawbacks of Depth-Limited Search?

Section B: Medium/Analytical

11. Explain the PEAS (Performance measure, Environment, Actuators, Sensors) framework for agent design with a self-chosen example.
12. Discuss the four types of agents (simple reflex, model-based, goal-based, utility-based) with examples.
13. Explain the concept of *rational agents* with an example in self-driving cars.
14. Compare BFS and Uniform Cost Search in terms of time complexity, space complexity, completeness, and optimality.
15. With the help of diagrams, illustrate how iterative deepening search combines the advantages of BFS and DFS.
16. Explain how environment properties (deterministic, stochastic, episodic, sequential, etc.) affect agent design.
17. Problem formulation: Describe the steps of formulating the “Vacuum Cleaner World” problem.
18. Explain Uniform Cost Search with an example graph and justify why it is optimal.
19. Compare and contrast Artificial Intelligence (AI), Machine Learning (ML) and Deep Learning (DL) based on their definitions, characteristics, applications.
20. Examine the major challenges and ethical issues in Artificial Intelligence (Al) in terns of bias, privacy, job displacement, security, and decision-making transparency. Analyse their potential impact on society
21. What are the three types of AI? Compare their characteristics in terms of their learning capability, adaptability, and decision-making process and provide real-world examples.

22. How do utility-based agents optimize decision-making? Explain with an appropriate example.
23. With the help of a suitable diagram, illustrate the perception-action cycle in intelligent agents.
24. How does a learning agent improve its performance over time? Analyze the role of its components in the learning process

---

Section C: Applied/Numerical

25. Formulate the *8-puzzle problem* as a search problem. Clearly specify:

- Initial state, goal state, operators, path cost.

26. Solve a small graph traversal problem using BFS. Show the step-by-step frontier expansion.
27. Apply DFS to the same graph and show the difference in node exploration order.
28. Show the working of Uniform Cost Search on a weighted graph with nodes A–F.
29. Demonstrate Iterative Deepening Search on a binary tree of depth 3. Show nodes visited at each depth limit.
30. Model the *Missionaries and Cannibals problem* using problem formulation and suggest which uninformed search strategy is suitable.
31. For a given maze representation (grid world), compare BFS, DFS, and UCS in terms of path quality and computational effort.

32. **Self-Driving Car Agents**
    1. Imagine you are designing a self-driving taxi agent.
        1. Define the PEAS description for this system.
        2. What type of agent architecture (simple reflex, model-based, goal-based, utility-based) would be most suitable and why?
33. **Healthcare Diagnosis**
    1. A hospital wants to build an AI system that diagnoses patients based on symptoms.
        1. Formulate this problem as a search problem.
        2. Which uninformed search strategy would be most appropriate if all solutions have the same cost?
34. **Delivery Drone Navigation**
    1. A delivery drone must find the shortest path to deliver a package in a city grid.
        1. Compare BFS and UCS for this task.
        2. Which one would you recommend if streets have different travel times?
35. **Game Playing Agent**
    1. You are building an agent to solve a maze-like game.
        1. Would DFS or BFS be more effective if the maze is very deep but has a single solution? Justify.
36. **Missionaries and Cannibals**
    1. Model this problem as an agent task.

1. State the performance measure, environment type, actuators, and sensors.

Unit II: Informed Search & Heuristics

Section A: Short Conceptual

1. Define heuristic function. Give two practical examples.
2. What is the difference between informed and uninformed search?
3. State the properties of an admissible heuristic.
4. Define heuristic dominance. Why is it important?
5. What is Greedy Best-First Search?
6. Define A* search. Write its evaluation function.
7. What are memory-bounded heuristic searches? Name two types.
8. Differentiate between local and systematic search.
9. What is simulated annealing inspired from in real life?
10. Write one drawback of Hill-Climbing search.

Section B: Medium/Analytical

11. Compare Greedy Best-First Search and A* in terms of optimality and completeness.
12. Prove that A* search is optimal when using an admissible heuristic.
13. What are consistent heuristics? Explain with an example.
14. Explain Iterative Deepening A* (IDA*) search. How does it save memory?
15. Discuss SMA* (Simplified Memory-Bounded A*) and its application.
16. What is the role of heuristic design in the 8-puzzle problem? Compare misplaced-tile heuristic vs Manhattan distance heuristic.
17. Explain Simple Hill-Climbing and Steepest-Ascent Hill-Climbing with diagrams.
18. Discuss the significance of the cooling schedule in Simulated Annealing.

Section C: Applied/Numerical

19. Given a weighted graph and heuristic values, solve using Greedy Best-First Search. Show the path chosen.
20. Apply A* search to the same graph. Show how f(n), g(n), and h(n) values are computed.
21. Prove that Manhattan distance heuristic for the 8-puzzle problem is admissible.
22. Design a heuristic for the Traveling Salesman Problem (TSP). Discuss whether it is admissible.
23. Apply IDA* to solve a simple state-space problem. Show the iterative threshold increases.

24. Solve a function optimization problem using Hill-Climbing ($f(x) = -x^2 + 6x$). Show iterations.
25. Demonstrate Simulated Annealing for the same function. Show how random choices avoid local maxima.

26. **Route Planning App (like Google Maps)**
    1. Your app must suggest the shortest driving path between two cities.
        1. Would you use Greedy Best-First Search or A*? Why?
        2. Suggest a suitable heuristic for this problem.
27. **Robot Vacuum Cleaner**
    1. A robot vacuum cleaner must clean a dirty room efficiently.
        1. How can local search (Hill-Climbing) cause the robot to get stuck?
        2. How can simulated annealing help overcome this issue?
28. **Puzzle Game (8-Puzzle)**
    1. You are designing a mobile puzzle-solving app.
        1. Propose two admissible heuristics for the 8-puzzle.
        2. Which one would make A* search more efficient? Why?
29. **Job Scheduling**
    1. A company wants to schedule tasks with deadlines and penalties for delays.
        1. How can heuristic search be applied?
        2. Suggest one heuristic function that might help optimize the schedule.
30. **Pathfinding for a Drone**
    1. A drone must deliver aid supplies in disaster areas with obstacles.
31. How would A* with a Euclidean distance heuristic perform compared to Greedy Best-First Search?

Unit III: Constraint Satisfaction & Adversarial Search

Section A: Short Conceptual

1. Define CSP with two examples from real life.
2. What is variable ordering in CSP?
3. Define MRV heuristic.
4. Explain the Least Constraining Value (LCV).
5. What is constraint propagation?
6. Differentiate between Forward Checking and Arc Consistency.
7. What is intelligent backtracking?
8. What is the Minimax principle in game theory?
9. What is alpha-beta pruning?
10. Define imperfect real-time decision making in games.

Section B: Medium/Analytical

11. Explain the Map Coloring problem as a CSP.
12. Discuss the Backtracking Search algorithm for CSPs.
13. Compare MRV heuristic and Degree heuristic in variable selection.
14. Explain Forward Checking with an example.
15. Discuss the AC-3 algorithm for arc consistency.
16. What is the Min-Conflicts heuristic? Why is it suitable for large CSPs?
17. Write the Minimax algorithm for two-player zero-sum games.
18. Discuss how alpha-beta pruning reduces the branching factor in minimax.

Section C: Applied/Numerical

19. Solve the *4-Queens problem* using backtracking search. Show each step of placement and backtracking.
20. Apply MRV and LCV heuristics to solve a CSP with 3 variables (domains given by teacher).
21. Demonstrate Forward Checking on a small CSP. Show how domains reduce after each assignment.
22. Apply AC-3 algorithm on a binary CSP and demonstrate arc consistency step-by-step.
23. Solve a small Sudoku (4×4) puzzle using CSP techniques.
24. Construct a minimax tree for Tic-Tac-Toe up to depth 2. Identify the best move.
25. Apply alpha-beta pruning on the same tree. Show pruned nodes clearly.

26. **Sudoku Solver**
    1. A gaming company is designing an AI Sudoku solver.
        1. Which CSP techniques (backtracking, forward checking, arc consistency) would be most effective? Why?
27. **University Timetabling**
    1. A university needs to schedule exams without overlapping.
        1. How can MRV and LCV heuristics help in solving this CSP efficiently?
28. **4-Queens Chess Puzzle**
    1. In solving the 4-Queens problem:
        1. How does backtracking work?
        2. What improvements do min-conflicts heuristics bring?
29. **Tic-Tac-Toe Game AI**
    1. You are building an AI to play Tic-Tac-Toe.
        1. Construct a minimax tree for two moves ahead.
        2. Apply alpha-beta pruning to optimize the decision.
30. **Real-Time Game Decision**
    1. In a real-time strategy game, the AI cannot compute the full minimax tree due to time constraints.
        1. How would you modify the decision-making process to still make competitive moves?