Danish

AI 2022:

Q. What do you mean by completeness of a search? Why DFS is not always complete?

Ans. Completeness in the context of search algorithms refers to the ability of an algorithm to find a solution if one exists. A search algorithm is considered complete if it guarantees to find a solution, given enough time and resources, whenever a solution exists in the search space.

Depth-First Search (DFS) is not always complete because it can get stuck in infinite loops or fail to find a solution even if one exists due to the following reasons:

1. **Infinite Branching Factor:** If the search space has an infinite branching factor (meaning that each state has an infinite number of successors), DFS may go down an infinite path without finding a solution.
2. **Limited Memory:** DFS does not keep track of visited states by default. In cases where there is a cycle in the state space, DFS may enter into an infinite loop, revisiting states and never reaching the goal.
3. **Non-Optimal Paths:** DFS does not guarantee finding the shortest path to a solution. If the search space has multiple paths to a solution, DFS may find a suboptimal solution before exploring more promising paths.
4. **Depth Limit:** In some cases, a depth limit may be imposed on DFS to prevent it from going too deep into the search space. If the solution lies beyond the depth limit, DFS may fail to find it.

Q. "If heuristic is consistent, then the heuristic is admissible, but reverse is not true" – Justify your answer.

Ans.

1. **If Heuristic is Consistent, Then It is Admissible**:

A consistent heuristic, satisfying the triangle inequality, ensures that the estimated cost to the goal from any state is never greater than the cost of reaching a successor plus the estimated cost from the successor. This guarantees admissibility by preventing overestimation.

2. **The Reverse is Not True:**

Admissibility only requires that a heuristic never overestimates the true cost. Consistency is a stronger condition, ensuring a more specific relationship between heuristic estimates and actual costs. Some admissible heuristics may lack this consistent relationship, making the reverse statement untrue.

Q. What is an agent? What do you mean by rational agent? Explain.

Ans. Agent:

- An agent is an entity that perceives its environment through sensors and acts upon that environment through effectors. In the context of artificial intelligence, an agent can be a computer program or system designed to perform tasks autonomously or semi-autonomously.

Rational Agent:

- A rational agent is an agent that acts to achieve the best outcome or maximize its performance measure, given its perceptual history and built-in knowledge.
- The concept of rationality is closely tied to decision-making. A rational agent makes decisions that lead to the most favorable outcome based on its understanding of the world and its goals.
- Rationality doesn't necessarily mean perfection; rather, it implies making the best possible decision with the available information. A rational agent may not always achieve the optimal outcome due to incomplete information or computational limitations, but it strives to make decisions that are reasonable given its knowledge.

Q. Explain the PEAS description of the task environment for a Medical diagnostic system.

Ans. PEAS Description for a Medical Diagnostic System :

1. **Performance Measure:**
   - Goal: Make sure the system gives correct and quick diagnoses.
   - Criteria: System should make as few mistakes as possible, find diseases early, and use resources wisely.
2. **Environment:**
   - Dynamic: Patient conditions change.
   - Semi-Observable: Some things happening inside the patient are not easy to see.
   - Multi-Agent: The system works with different healthcare professionals.
3. **Actuators:**
   - Treatment Recommendations: Suggest what medicines or therapies can help.
   - Communication: Share test results and work together with healthcare professionals.
4. **Sensors:**
   - Patient Data: Collect information from tests and patient history.
   - Medical Records: Look at electronic health records for a full picture of the patient's health.

Q. What do you mean by a heuristic?

Ans. A heuristic is a practical rule or method that aids problem-solving by providing a simplified, efficient strategy or shortcut to reach a solution, particularly in situations where exhaustive search or precise computation is challenging.

Q. Differentiate between Goal-based agent and Utility-based agent.

Criteria	Goal-based Agent	Utility-based Agent
Objective	Achieve specific goals or states.	Maximize overall utility or satisfaction.
Representation	Uses explicit goal representation.	Represents preferences through utility functions.
Decision Making	Decisions based on achieving predefined goals.	Decisions based on maximizing expected utility.
Flexibility	May lack flexibility if goals are rigidly defined.	Offers more flexibility as it considers preferences.
Multiple Goals	Handles multiple goals individually.	Balances trade-offs among multiple goals.
Uncertainty Handling	Focuses less on handling uncertainty explicitly.	Can incorporate uncertainty through utility values.
Example Scenario	Chess-playing agent aiming to checkmate the opponent.	Shopping agent maximizing user satisfaction and cost.

Q. Compare Hill-Climbing and Best First-Search

Criteria	Hill-Climbing	Best-First Search
Objective	Move towards the goal state by improving current state.	Select paths based on an evaluation function, aiming for the most promising path to the goal.
Memory	Minimal memory requirement. Only the current state is retained.	Keeps a priority queue or list of visited states based on evaluation scores. Requires more memory.
Completeness	Not always complete; can get stuck in local optima.	Completeness depends on the evaluation function. Can handle some local optima through proper heuristics.
Optimality	May not find the optimal solution due to local maxima.	Can find an optimal solution based on the evaluation function and heuristic information.
Backtracking	No backtracking; only moves forward.	Backtracks if a more promising path is found.

Q. Short Notes:

a) Dempster-Shafer Theory :

The **Dempster-Shafer theory** is a mathematical framework used in artificial intelligence to handle uncertainty and make decisions based on incomplete or conflicting information. It deals with belief functions and evidence combination. In simple terms, imagine you have different pieces of evidence from various sources, but you're not sure which ones to trust. Dempster-Shafer theory provides a way to combine these pieces of evidence and come up with a more informed decision, considering the uncertainty or conflict in the information.

b) NLP - Natural Language Processing :

Natural Language Processing (NLP) is a field of artificial intelligence focused on making computers understand, interpret, and generate human language in a way that is both meaningful and useful. It involves the interaction between computers and human language. NLP applications include machine translation, sentiment analysis, speech recognition, and chatbots. In simpler terms, NLP allows computers to read, understand, and respond to human language, making it easier for us to communicate with machines.

c) Reasoning with Fuzzy Logic :

Fuzzy Logic is a mathematical framework that deals with uncertainty and imprecision in decision-making. Unlike traditional logic, which is based on crisp, binary values (true or false), fuzzy logic allows for degrees of truth. It is particularly useful when describing concepts that are not precisely defined. So, when we talk about something being "slightly hot" or "very cold," fuzzy logic helps computers reason about these vague terms. It's like a middle ground between true and false, making it more applicable to real-world situations where things are not always black and white.

d) Knowledge Acquisition :

Knowledge Acquisition in artificial intelligence is the process of gathering, organizing, and storing information to build a knowledge base for a computer system. It involves extracting knowledge from various sources, including human experts, documents, and databases. Think of it as teaching a computer what it needs to know to perform specific tasks. This process is crucial for developing intelligent systems that can make informed decisions or solve problems based on a broad range of information.

e) Artificial Neural Network :

Artificial Neural Network (ANN) is a computational model inspired by the human brain's structure and functioning. It consists of interconnected nodes, or "neurons," organized in layers. Neural networks are used in machine learning to recognize patterns, classify data, and make predictions. In simpler terms, an ANN learns from examples and can generalize its learning to make predictions or decisions on new, unseen data. It's like training a computer to recognize patterns by showing it a bunch of examples, allowing it to make smart decisions on its own.

AI 2021:

Q. Discuss on the disadvantageous issues of Hill Climbing Technique in finding solution to state-space search problem and also suggest strategies to overcome the issues.

Ans. Disadvantageous Issues of Hill Climbing :

1. **Local Optima**: Hill climbing can get stuck in local optima, which are points in the search space where it thinks it's found the best solution but hasn't explored enough to know for sure.
2. **Plateau Problem**: In cases where many neighboring states have the same value, hill climbing might struggle to make progress, leading to slow convergence.
3. **No Backtracking**: Hill climbing doesn't backtrack; if it makes a wrong move, it might miss better solutions.

Strategies to Overcome Issues :

1. **Random Restarts**: Restart the search from different random initial states to escape local optima and explore alternative paths.
2. **Simulated Annealing**: Introduce randomness in the decision-making process, allowing the algorithm to occasionally accept worse solutions. This helps in exploring a broader search space.
3. **Adaptive Step Sizes**: Adjust the size of the steps taken during each iteration based on the local terrain. This helps navigate steep regions more effectively.
4. **Backtracking Mechanisms**: Implement mechanisms to backtrack and reconsider previous choices, allowing the algorithm to explore alternative paths.
5. **Multi-Start Hill Climbing**: Run multiple instances of hill climbing from different starting points concurrently to increase the chances of finding a better solution.

Q. What do we mean by admissibility and consistency of a heuristic function?

Admissibility : Admissibility means the heuristic never thinks a task will be harder than it actually is. It gives an estimate that's fair or even a bit optimistic, ensuring we don't miss the best solution.

Consistency : Consistency is about being fair along the way. If the heuristic says going from A to B and then to the goal costs less than going directly from A to the goal, that makes sense. It keeps the estimates in check and helps in finding the goal efficiently.

Q. Short Notes:

a) Means-End-Analysis:

Means-End-Analysis is like a problem-solving approach for computers. Imagine you have a goal, but you're not there yet. Means-End-Analysis breaks the big goal into smaller parts. It figures out the difference between where you are and where you want to be. Then, it tries to find actions or steps to reduce that difference. It keeps doing this until the goal is reached. It's like saying, "I want to reach the mountaintop, and right now, I'm here. What can I do to get closer, step by step?"

b) Basic Steps of Natural Language Processing :

Natural Language Processing (NLP) helps computers understand and talk like humans. Here are the basic steps:

1. **Tokenization:** Break sentences into words (tokens).
2. **Part-of-Speech Tagging**: Figure out if each word is a noun, verb, etc.
3. **Parsing**: Understand the structure and relationships between words.
4. **Named Entity Recognition**: Identify names of people, places, etc.
5. **Sentiment Analysis**: Decide if the text expresses positive, negative, or neutral feelings. These steps help computers read, understand, and respond to human language.

c) Simulated Annealing :

Simulated Annealing is a bit like cooking and cooling metal, but for problem-solving. Imagine you have a problem, and you want the best solution. Simulated Annealing starts with an initial guess. It then explores nearby solutions, like trying different recipes. If it finds a better one, it keeps it. But sometimes, it might accept a worse solution to avoid getting stuck in a bad spot. It's like saying, "I'm cooking a dish, and if it's getting better, I'll keep it. But sometimes, I'll try a different recipe to see if it's even better." This way, it avoids getting stuck in a not-so-great solution and finds the best one.

AI 2020:

Q. Differentiate between Inheritable knowledge and Inferential knowledge

Criteria	Inheritable Knowledge	Inferential Knowledge
Definition	Passed down from others, like traditions or learning.	Derived through reasoning or logical thinking.
Source	Comes from external sources or teachings.	Emerges from thinking and drawing conclusions.
Nature of Knowledge	Tangible facts, skills, or explicit information.	Implicit understanding, drawing connections.
Transferability	Can be easily taught or shared.	Might need more effort to convey reasoning processes.
Examples	Cultural practices, learned skills, historical facts.	Logical reasoning, problem-solving, insights from thinking.

Q. Differentiate between procedural and declarative knowledge.

Criteria	Procedural Knowledge	Declarative Knowledge
Nature	Concerned with "how-to" or the process of doing.	Focuses on "what is" or the statement of facts.
Representation	Knowledge about the steps or procedures to perform a task.	Involves statements of information or descriptions.
Examples	Skills, techniques, and methods for accomplishing tasks.	Facts, information, and descriptions of objects or concepts.
Application	Applied knowledge used for performing actions or tasks.	Descriptive knowledge used for understanding concepts or situations.
Transferability	Often requires hands-on experience and practice.	Can be easily communicated and transferred through explanation.
Learning Approach	Emphasizes practice, repetition, and hands-on experience.	Emphasizes understanding, memorization, and comprehension.

Q. How do you evaluate any search technique?

Ans. Evaluating a search technique is like checking how well it's doing the job. Here's how you can do it:

1. **Completeness :**
    - Ask if the search technique can always find a solution if one exists. If it can, it's complete. If not, it might miss solutions.
2. **Optimality :**
    - Check if the search technique finds the best solution. If it consistently gives the best answer, it's optimal. If it sometimes settles for okay solutions, it might not be.
3. **Time Complexity:**
    - Look at how much time the search technique takes. If it's quick, that's good. But if it takes too long, it might not be practical for some situations.
4. **Space Complexity:**
    - See how much memory or space the search technique needs. If it uses a lot, that could be a problem, especially for big problems.
5. **Simplicity :**
    - Consider how easy it is to understand and implement the search technique. If it's straightforward, it's user-friendly. If it's too complex, it might be hard to work with.

Checking these things helps decide if a search technique is reliable, quick, not too memory-hungry, easy to understand, and if it consistently finds the best solutions. It's like testing to see if a tool is good for the job!

Q. Under what condition is breadth-first search optimal?

Ans. BFS in like exploring a map step by step. It is optimal when:

1. **Uniform Edge Costs**: All paths between nodes have the same cost.
2. **Shortest Path**: BFS naturally finds the shortest path between the starting and goal nodes.
3. **FIFO Queue:** BFS uses a First-In-First-Out (FIFO) queue for exploration.
4. **Complete Graph**: If all nodes are connected, BFS will find the optimal path.
5. **Non-negative Costs**: BFS works optimally when all edge costs are non-negative. Negative costs can lead to suboptimal results.

Q. Write down the disadvantages of hill climbing search procedure

Ans. Disadvantages of Hill Climbing Search Procedure:

1. **Local Optima**: Gets stuck in local best solutions, potentially missing the overall best.
2. **Plateau Problem**: Struggles in flat regions, causing slow progress.
3. **No Backtracking**: Doesn't reconsider past choices, missing better solutions.
4. **Sensitivity to Initial State**: Outcome heavily depends on the starting point.
5. **Illusion of Optimality**: May think it found the best solution, but it could be local rather than global.

Q. Explain the PEAS description of the task environment for an automated taxi

Ans. PEAS Description for Automated Taxi:

1. **Performance Measure** : The performance measure for the automated taxi would be the efficiency and safety of transportation. It includes factors such as minimizing travel time, adhering to traffic rules, and ensuring passenger safety.
2. **Environment** : The environment for the automated taxi includes the road network, traffic conditions, weather, and interactions with other vehicles and pedestrians. It encompasses both urban and suburban settings.
3. **Actuators** : The actuators for the automated taxi involve the physical components responsible for actions. This includes the vehicle's steering system, acceleration and braking mechanisms, turn signals, and communication devices for interaction with the environment.
4. **Sensors** : Sensors provide the automated taxi with information about its surroundings. This includes cameras, lidar, radar, and other sensors to detect obstacles, traffic signals, pedestrians, and monitor road conditions.

Task Environment:

The task environment involves transporting passengers from one location to another while navigating through traffic, obeying traffic rules, and ensuring passenger comfort and safety. It requires effective route planning and decision-making in response to dynamic road conditions.

Q. What is expert system? Describe its architecture and characteristics.

Ans. **Expert System**: An expert system is a computer program designed to mimic the decision-making abilities of a human expert in a specific domain. It uses knowledge and reasoning techniques to provide solutions or make decisions within that domain.

Architecture:

1. **Knowledge Base**:
    - The knowledge base stores information, facts, and rules relevant to the specific domain. It is the brain of the expert system and is created by domain experts.
2. **Inference Engine**:
    - The inference engine processes information from the knowledge base to make decisions or draw conclusions. It uses reasoning mechanisms like rule-based reasoning, fuzzy logic, or machine learning algorithms.
3. **User Interface**:
    - The user interface allows interaction between the expert system and the end user. It can be text-based or graphical, providing a platform for users to input queries and receive system outputs.

Characteristics:

1. **Domain Specific**:
    - Expert systems are designed for a specific domain, such as medical diagnosis, financial analysis, or troubleshooting technical issues.
2. **Knowledge-Driven**:
    - They rely on a knowledge base created by human experts, consisting of facts, rules, and heuristics related to the problem domain.
3. **Decision-Making**:
    - Expert systems excel in decision-making tasks, providing recommendations or solutions based on the input data and the knowledge stored in the system.

Q. Compare Hill-Climbing and Best First-Search.

Criteria	Hill-Climbing	Best First-Search
Objective	Find a solution, may not be optimal.	Find the optimal solution.
Memory Usage	Low memory usage.	Can have high memory requirements.
Completeness	Not guaranteed to find a solution.	Complete if the branching factor is finite and the cost function is bounded.
Heuristic Use	Uses heuristics for local decisions.	Utilizes heuristics for global decisions, prioritizing nodes based on an evaluation function.
Backtracking	Does not backtrack; may get stuck in local optima.	Can backtrack and explore alternative paths, potentially finding a better solution.

Q. What problems may arise in A* algorithm, if value of h is: i) Underestimated and ii) Overestimated? ***Explain with proper diagram***

Ans. A Algorithm with Underestimated and Overestimated h Values:*

i) Underestimated h Value:

Problem: If the heuristic value (h) is underestimated, A* may become too optimistic, leading to suboptimal solutions. The algorithm might explore paths that seem promising based on the low heuristic, but they may not be the best.

ii) Overestimated h Value:

Problem: If the heuristic value (h) is overestimated, A* may become too pessimistic. It might excessively explore alternative paths, slowing down the search process and potentially missing efficient solutions.

Q. What is FOPL?

Ans. **FOPL (First-Order Predicate Logic):** FOPL is a logical system that expands on propositional logic, introducing variables, quantifiers, and predicates. It's used in AI for representing complex relationships and properties between objects in a domain.

AI 2019:

Q. Compare uninformed search and informed search.

Criteria	Uninformed Search	Informed Search
Objective	Find a solution with no additional information.	Find the optimal solution using domain-specific knowledge.
Heuristic Use	No heuristics involved; decisions based on the current state.	Utilizes heuristics to guide search decisions, focusing on more promising paths.
Memory Usage	Typically lower memory requirements.	May have higher memory requirements, especially with extensive heuristic information.
Completeness	Complete if a solution exists.	Complete if the heuristic is admissible and the branching factor is finite.
Optimality	Not guaranteed to find the optimal solution.	Aims to find the optimal solution if an admissible heuristic is used.

Q. What is Prolog? Explain the concept of fact and rule in Prolog.

Prolog: Prolog (Programming in Logic) is a programming language commonly used for artificial intelligence and symbolic reasoning tasks. It is particularly well-suited for tasks involving logical relationships and rule-based decision-making.

Concept of Fact and Rule in Prolog:

Fact:

In Prolog, a fact is a statement that is assumed to be true. It represents a piece of information about relationships or properties in the domain. Facts are typically simple and can be used to state concrete and specific knowledge. Eg: `is_mammal(cat).`

In this example, the fact "cat is a mammal" is represented.

Rule:

A rule in Prolog defines a relationship or condition based on other facts or rules. It consists of a head and a body. The head specifies the relationship or condition being defined, and the body contains the conditions that must be satisfied for the rule to be true.

Eg: `has_feathers(X) :- is_bird(X), can_fly(X).`

In this example, the rule states that if something is a bird and can fly, then it has feathers.

Q. What is Natural Language processing (NLP)? Give some common applications of Natural Language processing.

Natural Language Processing (NLP): NLP, or Natural Language Processing, is a field of artificial intelligence that focuses on the interaction between computers and human language. It involves the development of algorithms and models that enable machines to understand, interpret, and generate human language in a way that is both meaningful and contextually relevant.

Common Applications:

- **Chatbots**: Use NLP to simulate conversation, understand user queries, and provide assistance.
- **Sentiment Analysis:** Applies NLP to analyze and determine sentiment in text, useful in social media monitoring and feedback analysis.
- **Machine Translation:** Leverages NLP for accurate translation in applications like Google Translate.

Q. What are the different steps involved in NLP?

Steps in Natural Language Processing (NLP):

1. **Tokenization:** Breaks down text into smaller units (tokens) such as words or phrases for further analysis.
2. **Part-of-Speech Tagging:** Assigns grammatical categories (e.g., noun, verb) to each token, aiding in understanding sentence structure.
3. **Named Entity Recognition (NER):** Identifies and classifies entities (e.g., names, locations) in the text.
4. **Syntax and Parsing:** Analyzes the grammatical structure of sentences to understand relationships between words.
5. **Sentiment Analysis:** Determines the emotional tone of the text, classifying it as positive, negative, or neutral.

Q. Write down the advantages and disadvantages of Genetic algorithm

Advantages of Genetic Algorithms:

- **Parallel Search:** Simultaneously explores multiple potential solutions.
- **Global Optimization:** Well-suited for finding the best solution across a broad space.
- **No Derivative Requirements:** Applicable to problems with non-continuous or noisy functions.
- **Versatility**: Adaptable to a wide range of optimization problems.
- **Adaptability:** Handles changes in the problem space over time.

Disadvantages of Genetic Algorithms:

- **Computational Intensity:** Requires substantial computational resources.
- **No Guaranteed Solution**: Doesn't ensure finding the optimal solution.
- **Difficulty in Parameter Tuning:** Challenging parameter tuning requires domain expertise.
- **Not Always Efficient for Simple Problems:** May not be the most efficient for straightforward tasks.
- **Limited Handling of Constraints:** Challenges in handling constraints in optimization problems.

Short Notes:

1. **Expert System:** An expert system is a computer program designed to emulate the decision-making abilities of a human expert in a specific domain. It comprises a knowledge base, a set of rules, and an inference engine. The knowledge base contains information and facts about the domain, while the rules define logical relationships and decision criteria. The inference engine processes user queries, applying the rules to draw conclusions or provide recommendations. Expert systems are utilized in various fields, including medicine, finance, and troubleshooting, where they offer expertise and decision support.

2. **Intelligent Agent:** An intelligent agent is a software entity capable of perceiving its environment, making decisions, and taking actions to achieve specific goals. These agents operate autonomously and may exhibit learning and adaptation. They can be categorized into simple reflex agents, model-based reflex agents, goal-based agents, and utility-based agents, depending on their design and capabilities. Intelligent agents are employed in diverse applications, from game playing and autonomous vehicles to virtual personal assistants, contributing to the advancement of artificial intelligence.