**Multiple Choice Questions for Lecture 3: Solving Problems by Searching Algorithm**

**Types of Search Algorithms**

1. What are the two main types of search algorithms in AI?
   * A) Fast search and slow search
   * B) Uninformed search and informed search
   * C) Easy search and hard search
   * D) Online search and offline search

Answer: B) Uninformed search and informed search

1. What is another name for uninformed search?
   * A) Smart search
   * B) Blind search
   * C) Quick search
   * D) Easy search

Answer: B) Blind search

1. In uninformed search, what does the agent know?
   * A) The number of steps needed to reach the goal
   * B) The path cost to reach the goal
   * C) Only when it reaches a goal
   * D) How far away the goal is

Answer: C) Only when it reaches a goal.

1. What information does informed search use?
   * A) Only the path cost
   * B) The exact steps to reach the goal
   * C) Background information about the problem
   * D) Only the branching factor

Answer: C) Background information about the problem

1. What is the main advantage of informed search over uninformed search?
   * A) It's always faster
   * B) It always finds a solution
   * C) It uses background information about the problem
   * D) It never gets stuck

Answer: C) It uses background information about the problem

**Uninformed Search Strategies**

1. What is Breadth-First Search (BFS)?
   * A) A search method that explores the deepest nodes first
   * B) A search method that explores all nodes at the current depth before moving to the next level
   * C) A search method that only looks at the first few nodes
   * D) A search method that starts at the goal

Answer: B) A search method that explores all nodes at the current depth before moving to the next level

1. What data structure is used in Breadth-First Search?
   * A) FIFO queue
   * B) LIFO stack
   * C) Tree
   * D) Array

Answer: A) FIFO queue

1. Breadth-First Search (BFS) explores:
   * A) The deepest nodes first
   * B) The lowest cost nodes first
   * C) All nodes at the current depth before moving to the next level
   * D) Nodes randomly

Answer: C) All nodes at the current depth before moving to the next level

1. What happens in Uniform-Cost Search (UCS)?
   * A) It always chooses random paths
   * B) It prioritizes the lowest-cost path when multiple paths exist
   * C) It always explores the deepest paths first
   * D) It avoids paths with any cost

Answer: B) It prioritizes the lowest-cost path when multiple paths exist

1. What does Uniform-Cost Search (UCS) prioritize?
   * A) The closest node to the goal
   * B) The lowest-cost path when multiple paths exist
   * C) The deepest node in the tree
   * D) Random nodes

Answer: B) The lowest-cost path when multiple paths exist

1. When does Uniform-Cost Search (UCS) behave like Breadth-First Search (BFS)?
   * A) When the goal is very far away
   * B) When there are multiple goals
   * C) When all transitions have equal costs
   * D) When there are no solutions

Answer: C) When all transitions have equal costs

1. What is Depth-First Search (DFS)?
   * A) A search method that explores all nodes at the current depth before moving deeper
   * B) A search method that starts at the goal and works backward
   * C) A search method that explores branch nodes deeply before backtracking
   * D) A search method that only uses costs

Answer: C) A search method that explores branch nodes deeply before backtracking

1. What data structure is used in Depth-First Search?
   * A) Queue
   * B) FIFO
   * C) LIFO stack
   * D) Tree

Answer: C) LIFO stack

1. What is Bidirectional Search?
   * A) A search method that only works in two dimensions
   * B) A search that runs two searches simultaneously—one from start and one from goal
   * C) A search that always goes left and right
   * D) A search that uses two different algorithms

Answer: B) A search that runs two searches simultaneously—one from start and one from goal

1. What is the main advantage of Bidirectional Search?
   * A) It always finds the optimal solution
   * B) It only needs to explore half the path compared to traditional methods
   * C) It never gets stuck in loops
   * D) It doesn't need any memory

Answer: B) It only needs to explore half the path compared to traditional methods

**Informed Search Strategies**

1. What does Greedy Best-First Search use to decide which node to explore next?
   * A) Random selection
   * B) The lowest cost from the start
   * C) A heuristic function that estimates distance to goal
   * D) The shortest path

Answer: C) A heuristic function that estimates distance to goal

1. What function does Greedy Best-First Search use to evaluate nodes?
   * A) f(n) = g(n) + h(n)
   * B) f(n) = g(n)
   * C) f(n) = h(n)
   * D) f(n) = 1/g(n)

Answer: C) f(n) = h(n)

1. What does h(n) represent in search algorithms?
   * A) The height of the node
   * B) The estimated cost from node n to the goal
   * C) The path cost from the start to node n
   * D) The number of children of node n

Answer: B) The estimated cost from node n to the goal

1. What is a limitation of Greedy Best-First Search?
   * A) It is too slow
   * B) It can get stuck in loops or dead ends
   * C) It never works
   * D) It uses too much memory

Answer: B) It can get stuck in loops or dead ends

1. In A\* search, what does the evaluation function f(n) = g(n) + h(n) represent?
   * A) f(n) is the total estimated cost, g(n) is the path cost from start, h(n) is the estimated cost to goal
   * B) f(n) is the total time, g(n) is the speed, h(n) is the distance
   * C) f(n) is the number of steps, g(n) is steps taken, h(n) is steps remaining
   * D) f(n) is the solution, g(n) is the problem, h(n) is the method

Answer: A) f(n) is the total estimated cost, g(n) is the path cost from start, h(n) is the estimated cost to goal

1. How does A\* search combine UCS and Greedy Best-First Search?
   * A) It runs them one after another
   * B) UCS keeps solution cost low, Best-first helps find solution quickly
   * C) It chooses the better of the two
   * D) It alternates between them

Answer: B) UCS keeps solution cost low, Best-first helps find solution quickly

**Important Concepts**

1. What is the difference between traversal and search?
   * A) Traversal visits each node once, search finds a path between two nodes
   * B) Traversal is faster, search is more accurate
   * C) Traversal is for trees, search is for graphs
   * D) Traversal goes forward, search goes backward

Answer: A) Traversal visits each node once, search finds a path between two nodes

1. What does it mean when an algorithm is "complete"?
   * A) The algorithm is finished
   * B) The algorithm is perfect
   * C) The algorithm guarantees finding a solution if one exists
   * D) The algorithm has no errors

Answer: C) The algorithm guarantees finding a solution if one exists

1. What is a heuristic function in search algorithms?
   * A) A function that counts the number of steps
   * B) A function that estimates the cost to reach the goal
   * C) A function that measures algorithm speed
   * D) A function that creates the search tree

Answer: B) A function that estimates the cost to reach the goal