A.I Assignment

Q1) Define Artificial Intelligence. Explain different types of AI agents

→ "It is a branch of computer science by which we can create intelligent machines which can behave like a human, think like humans, and able to make decisions."

Artificial Intelligence exists when a machine can have human based skills such as learning, reasoning and solving problems.

Agents can be grouped into five classes based on their degree of perceived intelligence and capability.

1) Simple-Reflex agent
   i) The simple reflex agents are the simplest agents. These agents take decisions on the basis of the current percepts and ignore the rest of the percept history.
   ii) These agents only succeed in the fully observable environment.
   iii) It works on condition-action rule, which means it maps the current state to action.

2) Model based agent
   i) It work in a partially observable environment, and tracks the situation.
   ii) It has two important factors:
      a) Model - It is knowledge about "how things happen in the world."
      b) Internal state - It is a representation of the current state based on percept history.

3) Goal based agents
   i) The agent needs to know its goal which describes desirable situations.

ii) Goal based agents expand the capabilities of the model-based agent by having the 'goal' information.
iii) They choose an action, so that they can achieve the goal.

4) Utility based agents
:- i) These agents are similar to the goal-based agents but provide an extra component of utility measurement which makes them different by providing a measure of success at a given state.
ii) Utility-based agent act based not only goals but also the best way to achieve the goal.

5) Learning Agents
:- i) It can learn from its past experiences, or it has learning capabilities.
ii) It starts to act with basic knowledge and then able to act and adapt automatically through learning.

Q2) Differentiate informed search and uninformed search.

→ Informed search
:- i) It uses knowledge for the searching process.
ii) It finds solution more quickly.
iii) It may or may not be complete.
iv) cost is low.
v) It consumes less time
vi) Greedy search, A* search, Graph search.

Uninformed Search
:- i) It doesn't use knowledge for searching process
ii) It finds solution slow as compared to

informed search.
iii) It is always complete.
iv) cost is high
v) It consumes moderate time.
vi) Depth first search, Breadth first search.

Q3) Write a Hill Climbing algorithm. Explain the problems associated with it.

→ Algorithm -

1) Evaluate initial state -
   if goal state, stop and return success. Else, make initial state current.
2) Loop until the solution reached or until no new operators left to apply to current state:
   a) Select new operator to apply to the current producing new state.
   b) Evaluate new state:
      i) if goal state, stop and return success.
      ii) if better than the current state, make it current state, proceed.
      iii) Even if not better than the current state, continue until the solution reached.
3) Exit.

Problems with Hill Climbing Search

i) Local Maximum -
   All neighbouring states have values worse than the current. The greedy approach means we won't be moving to a worse state. This terminates the process even though there may have been a better solution. As a workaround, we use backtracking.

ii) Plateau -
   All neighbors to it have the same value. This makes it impossible to choose a direction. To

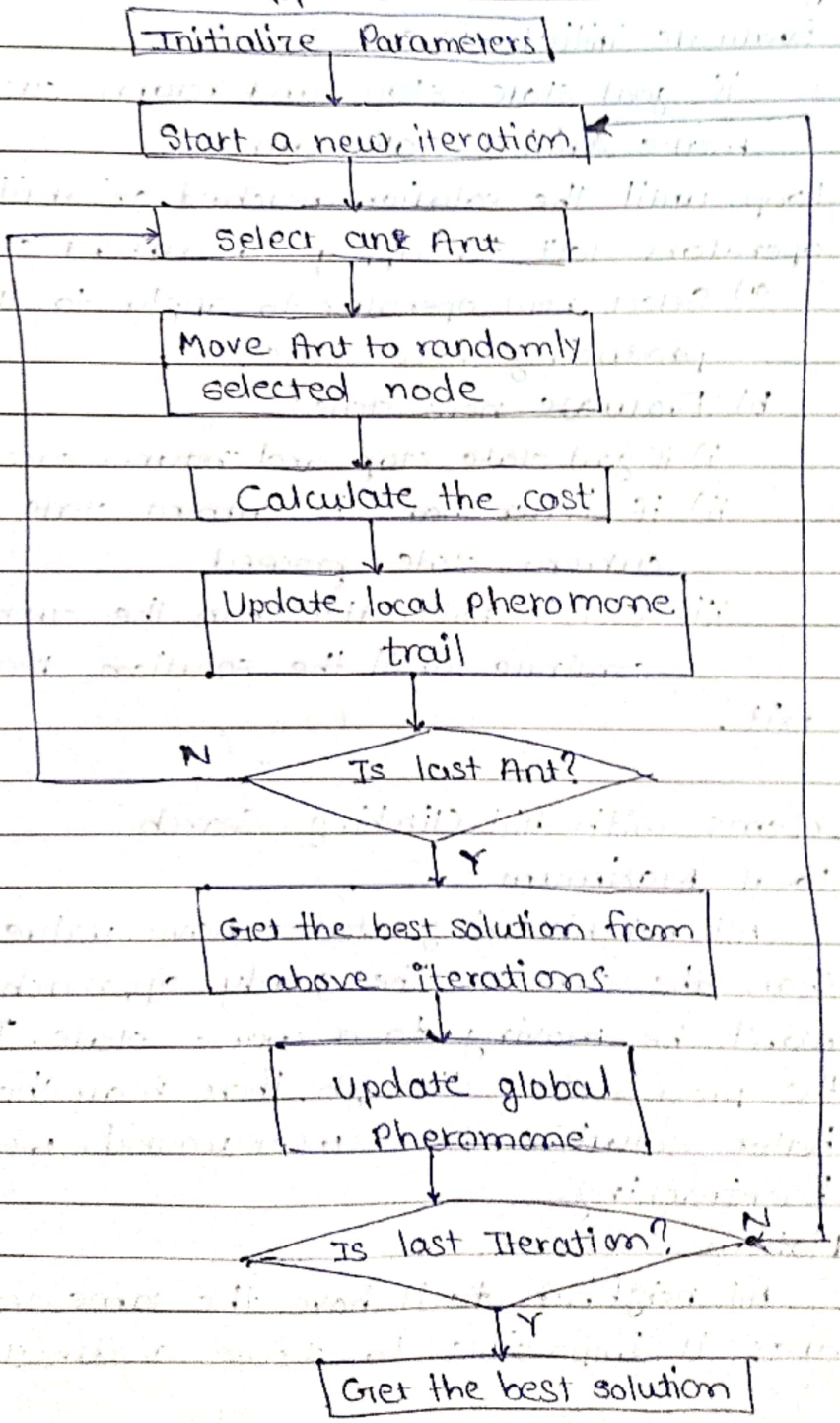
avoid this, we randomly make a big jump.

iii) Ridge

∴ At a ridge, movement in all possible directions is downward. This makes it look like a peak and terminates the process. To avoid this, we may use two or more rules before testing.

Q4) Design an Ant colony optimization algorithm and write down its application.

→



Applications of ACO

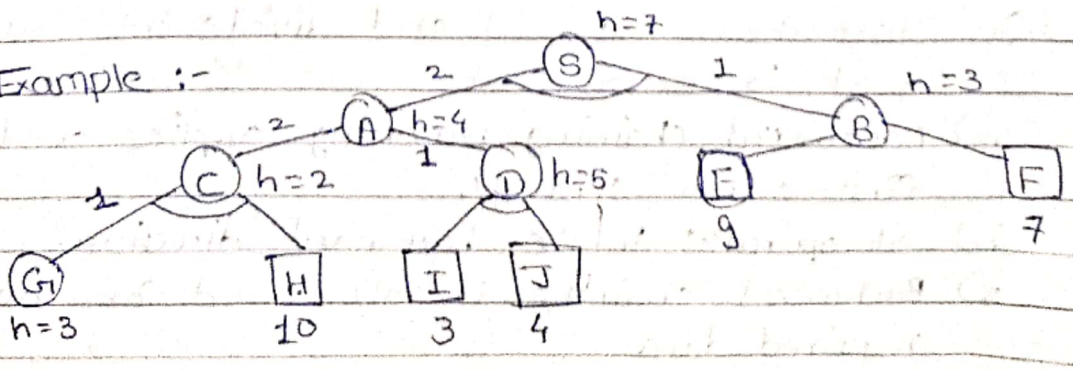
i) Travelling Salesman Problem (TSP)
ii) Quadratic Assignment Problem
iii) Vehicle Routing Problem.
iv) Graph Coloring Problem.
v) Sequential Ordering Problem
vi) Job scheduling Problem
vii) Routing in Telecommunications Networks.

Q5) Construct AO* Algorithm & give one example.

→ Algorithm

1) Create an initial graph with a single node (start node)
2) Transverse the graph following the current path, accumulating node that has not yet been expanded or solved.
3) Select any of these nodes and explore it. If it has no successors then call this value = Futility else calculate f(n) for each of the successors.
4) If f'(n) = 0, then mark the node as solved.
5) Change the value of f'(n) for the newly created node to reflect its successors by backpropagation.
6) Whenever possible use the most promising routes, if a node is marked as solved then mark the parent node as solved.
7) If the starting node is solved or value is greater than Futility then stop else repeat from step-2.

Example :-



AO* Analysis :-
AO* always find a minimum cost solution tree ($h^*(n) <= h(n)$ and all arc costs are positive)

Q6) Demonstrate the use of Forward and Backward chaining.

→ 1) Forward chaining :-
i) It is a bottom-up approach
ii) It known as data-driven inference technique as we reach to the goal using the available data.
iii) Forward chaining reasoning applies a breadth-first search strategy.
iv) It operates in the forward direction.
v) Forward chaining is aimed for any conclusion.

eg :- Goal :- a = 1
initial state ;
x = 1 and y = 2
Algorithm :-
if (x == 1 & y == 2)
then,
z = 3
if (z == 3).
a = 1

2) Backward chaining :-
i) It is a top-down approach
ii) It is known as goal-driven technique as we start from the goal and divide into sub-goal to extract the facts.
iii) Backward chaining reasoning applies a depth-first-search strategy.
iv) It operates in the backward direction.
v) Backward chaining is only aimed for the required data.

eg:- Goal : B = ?

Initial state;

A = It is raining

Given statement

It's raining and roads are wet

A B = Goal

Q7) Explain Goal stack planning in detail.

→ i) This is one of the most important planning algorithms, which is specially used by STRIPS. The stack is used in an algorithm to hold the goals and actions that satisfy the goal. A knowledge base is used to hold the current state, actions.

ii) Goal stack is similar to a node in a search tree, where the branches are created if there is a choice of an action.

iii) In this method, the problem solver makes use of a single stack that contains both goals & operators that have proposed to satisfy those goals.

iv) The problem solver also relies on a database that describes the current situation and a set of operators described as PRECONDITION, ADD and DELETE lists.

eg:- ON(B,A)

clear(B)

ArmEmpty

ON(B,A)^clear(B)^armEmpty

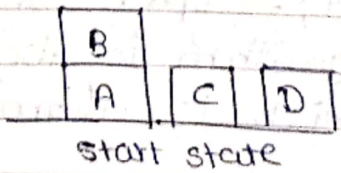
UNSTACK(B,A)

Holding(C)

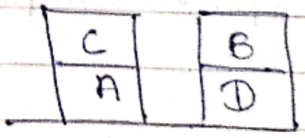
clear(A)^Holding(C)

Stack(C,A)

on(B,D)

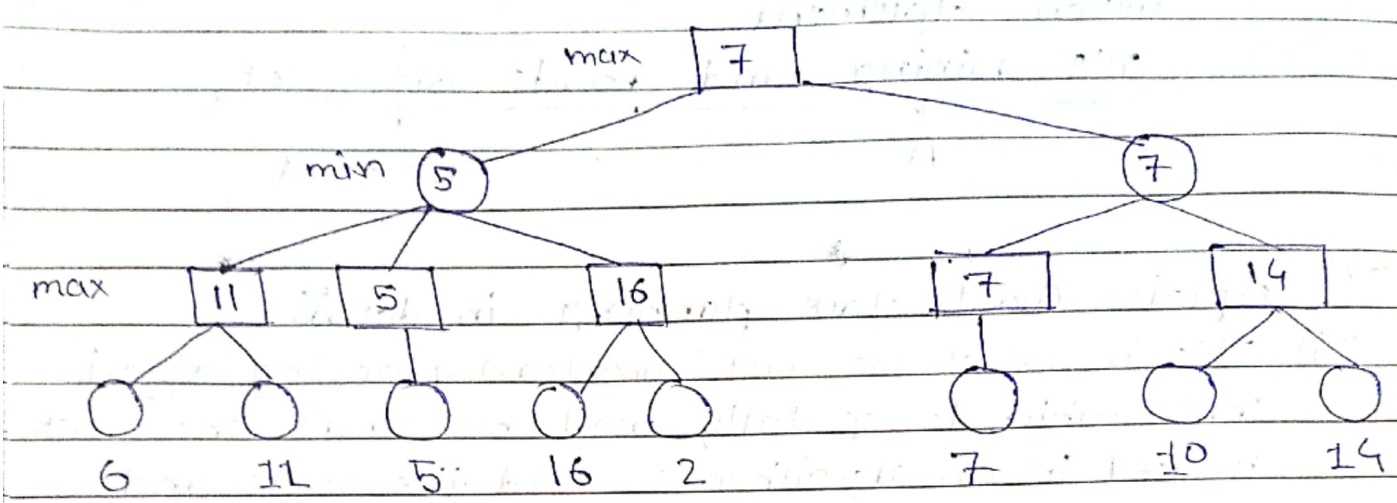


Start state



Goal state

Q8) Solve the following game tree with the help of Min-Max algorithm & write down the analysis of Min-Max algorithm.



Analysis of Min-Max algorithm:-

i) Complete - Min-Max algorithm is complete. It will definitely find a solution (if exist), in the finite search tree.

ii) Optimal - Min-Max algorithm is optimal if both opponents are playing optimally.

iii) Time-Complexity :- As it performs DFS for the game-tree, so the time complexity of Min-Max algorithm is $O(b^m)$, where b is branching factor of the game-tree, and m is the maximum depth of the tree.

iv) Space Complexity :- Space complexity of Min-Max algorithm is also similar to DFS which is $O(bm)$.