D15C
01

05/05

ABHINAV SWAMINATHAN
ROLL NO : 01
T.E. (INFT) SEM VI

AIDS- ASSIGNMENT

1. What is AI? Considering the covid19 pandemic situation, how AI helped to survive and renoated our way of life with different applications?

→

Artificial Intelligence (AI) is a branch of computer science that focuses on creating intelligent systems capable of performing tasks that typically require human intelligence. These tasks include problem solving, decision making, learning perception, etc

Role in surviving & renvoating life during covid19:-

i Healthcare & medical diagnosis:

- CTscan analysis & x-ray diagnosis helped in rapid detection of covid19 cases.

ii Virus spread control:

- Social distance monitoring tools helped to monitor & enforce laws in public places.

iii remote work & education:

- work from home optimization & AI in online educations help people continue with their work & education remotely.

iv Fake news detection:

- AI helped to identify rapidly spreading fake news on social media.

2. What are AI Agents terminology? Explain with examples.

- Environment:-

- Everything which surrounds the agent & influences its actions. It can be complete or partially observable

- eg: chessboard environment for chess playing AI.

- Percepts:-
- Percepts are the raw data that an agent gets from its sensors
- eg: self driving car percepts include, GPS signals, etc.
- Actuator:-
- They are the components that allow an agent to take actions in the environment
- eg: A robotic arm uses motors as actuators to pick objects.
- Goal:-
- The final state of an agent to achieve is called as goal
- eg: victory in a chess game.

3. How is AI technique used to solve 8-puzzle problem?

→

- The 8 puzzle problem is a state space search problem in AI where a 3x3 grid contains 8 tiles numbered from 1 to 8. & 1 empty space
- Objective is to rearrange the tiles to reach a predefined goal state
- AI techniques:-
- Uninformed search methods such as:-
  - BFS : Expand the shallowest nodes first.
  - DFS : Explores as deep as possible before backtracking.
  - IDS : combines DFS & BFS to increase depth limit gradually
- Informed search methods:-
  - BFS : best first search based on a heuristic function that appears closest to the goal.
  - A* search : $f(n) = g(n) + h(n)$. Based on both heuristic & cost to node.

Initial state :

1	2	3
5	6	0
4	7	8

Goal state :

1	2	3
4	5	6
7	8	0

i Compute heuristics of each possible move.

ii Expand the state with the lowest f(n). & repeat.

4. what is PEAS description? Give PEAS description for the following?

→ Performance Measure : how success of agent is evaluated

Environment : Surroundings in which agent operates.

Actuators : Component that allows agent to take actions.

Sensors : Component that allows agent to perceive the environment.

i Taxi driver agent.

Perf. measure	Environment	Actuator	Sensor.
- safe driving	- traffic signal	- steering wheel	- cameras
- travel time.	- roads	- accelerator	- GPS
- traffic rules	- weather	- brakes	- fuel gauge

ii Medical dignosis system

Perf. measure	Environment	Actuator	Sensor.
- health of patient	- patient data	- display screen	- heart rate monitors
- accuracy of diagnosis	- symptoms	- alarm system.	
- recommended treatment.	- test reports	- robotic arms	- lab results

iii Music composer agent

Perf. measure	Environment	Actuators	Sensors.
- originality	- music db	- speaker system.	- microphones
- listener engagement	- user preferences.	- digital music interface	- user feedback
- quality.			- recognition s/w

iv. Aircraft autolander.

Perf. measure	Environment	Actuators.	Sensors.
- smooth landing	- runway	- landing gear	- altimeter
- accuracy in	- wind conditions	- flap	- GPS
touchdown	- air traffic	- air brake, rudder	- camera

v. Essay evaluator

Perf. measure	Environment	Actuators	Sensors.
- grading	- plagarism	- display screen	- optical
- grammar	databases	- text to	character
- plagarism check	- rubrice criteria	speech systems	recognition (OCR).

vi. Robotic sentry gun.

Perf. measure	Environment	Actuators	Sensors.
- neutralize threats	- lab area	- gun mechanism	- camera
- target tracking	- potential	- alarm siren	- thermal
- false alarms	intruders	- tracking system	sensors

5. Categorize a shopping bot for an offline shopping bot for an offline bookstore according to the following dimensions

-

- Observability: Partially observable. Relies on limited sensors input
- Deterministic or stochastic: stochastic. Customer pref is unpredictable.
- Episodic vs sequential: Sequential. Decision affects future actions.
- Static vs Dynamic: Dynamic. Customer behaviour is continuously evolving.
- Discrete vs continuous: Finite no of choices such as books, authors, payment options, etc.

- Single vs multiagent: Bot interacts with multiple agents incl. customers, store employees, & other bots.

6. Differentiate between model based & utility based agent

Model based Agent	Utility based Agent
- Agent that maintains an internal model of the env. to understands its current state & predict future states	· Agent that selects actions based on a utility fn. aiming to maximize long term satisfaction or benefit.
- Model updates its knowledge about the environment.	· Measures how desirable different states are.
- Less complex	· More complex
- Doesn't concern long term rewards	· Focus on long term rewards.
- eg: self driven car	· eg: shopping recommendation system.

7. Explain the architecture of knowledge based agent & learning agent.

→.

knowledge based agent: Stores knowledge & reasons c̄ it making decisions based on logical inference.

- knowledge base (kb): Stores facts, rules & heuristics about the environment.
- inference engine: Uses logical reasoning techniques like forward & backward chaining.
- perception: gathers data from the environment
- actions: execute action based on inferred knowledge.
- knowledge update mechanism: Updates itself as new facts are learned.

learning based agent: Agent that improves its perf. overtime by learning from experience, data & feedback.

- learning element: responsible for improving agents perf. by analyzing past experiences using ML techniques.
- critic: provides feedback on agents actions by evaluating success or failure.
- problem generation: supports new experiences for learning & exploration.

8. Convert the following to predicates.

→

Anita travels by car if available otherwise travels by bus

travels (x,y) → Person x travels by y

Available (y) → y (a vehicle) is available

Goes_via (y,z) → Vehicle y goes via z.

Puncture (y) → y (a vehicle)

a. ~Available (car) → Travels (Anita, bus).

b. Bus goes via Andheri & Goregaon

Goes_via (bus, andheri)

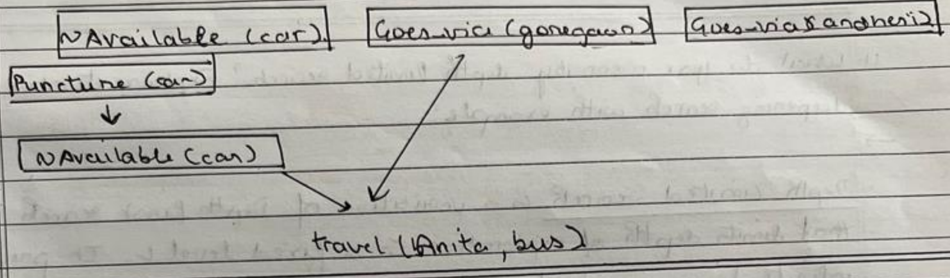
Goes_via (bus, goregaon).

c. Car has a puncture, so its not available

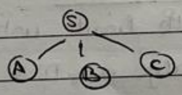
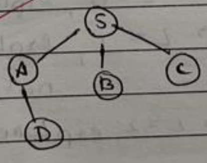
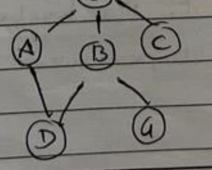
Puncture (car)

~Available (car).

Forward reasoning : will Anita travel via Goregaon



10. Find the route from s to G using BFS. [best first search]

Steps	Representation	~~Priority~~ Sta ~~Queue~~ Stack
i Load s		[S].
ii Pop s. Load A,B,C		[A, B, C].
iii Pop A. Expand D.		[B, C, D].
iv Pop.B. Expand G		[C, D, G].

G is goal node. ∴ Route from S-G is S→B→G

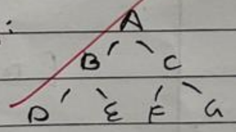
11. What do you mean by depth limited search? Explain iterative deepening search with example.

→

- Depth Limited Search is a variation of Depth first search that limits depth of exploration to a fixed level L. It prevents infinite loops in infinite state spaces & reduces mem. usage.
  - start with initial node until depth limit is reached.
  - if goal is found, return success.
  - backtrack & explore other nodes
  - if limit is reached, without finding goal, return failure.

- Iterative deepening search (IDS) combines both DFS & BFS by gradually inc. depth limit until goal is found
  ~~Eg.~~ It follows same algo as DLS, except if limit is reached, limit +1 & algo repeats

- eg:



→ L=0, explore A, goal not found
→ L=1, explore A-B, A-C, goal not found
→ L=2, explore B→D, B→E, C→F, C→G, goal G found at L=2

12 Explain hill climbing & its drawbacks in detail with example. Also state limitations of steepest ascent hill climbing.

→ Hill climbing is an informed search algorithm used for optimization problems in local search.

- We start with an initial state (random or given).
- Evaluate the neighboring states and choose one with the highest value
- Move to the best neighboring state & repeat.
- eg:- 8 queen problem where we need to add 8 queens on the chessboard such that no two queens attack each other.
  - We start with a random placement of queens.
  - Then move to a neighboring state by modifying the position of 1 queen.
  - repeat until conflicts cannot be reduced further.
- Drawbacks:- local maxima: Get stuck at peak which isn't global optimal.
  - plateau: All neighbouring states have the same value.
  - ridges: complex landscapes.
- Limitation of steepest ascent hill climbing: Only selects neighbor with the highest improvement.

- More computationally expensive: Need to evaluate all possible neighbors.
- Sensitive to initial data: Bad starting leads to poor solutions.
- Existing problems of hill climbing algorithm.

13. Explain simulated annealing & write its algorithm.

→ Its an optimized version of hill climbing designed to escape its limitations

. Its inspired by annealing process in metallurgy where a metal is heated & slowly cooled to remove defects.

- Algorithm:- Start with an initial solution s.
  - Generate a random neighbor s' of state s.
  - If s' is better than s, accept it.
  - If s' is worse, accept it with probability $P = e^{(-\Delta e/T)}$.

- return the best found solution.

14. Explain A^* algorithm with an example.

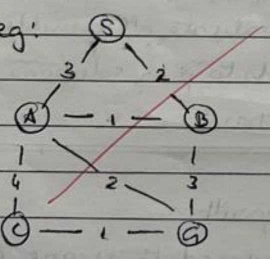
→

· A^* is a path and graph traversal algorithm which combines both - uniform cost search (UCS).
- greedy best first search (BFS).

· A^* evaluates each node using the function.
$f(n) = g(n) + h(n)$.

· Algorithm :- Initialize start node into an open list.
- Compute $f(n) = g(n) + h(n)$. Select node with the smallest $f(n)$.
- Move it to closed list & expand its neighbors.
- If neighbor is goal, return path.
- continue till goal is reached.

· eg:



$h(S) = 4.$, $h(A) = 2$, $h(B) = 1$, $h(C) = 2$
$h(G) = 0.$

i. Initialize S, Expand A, B. $f(A) = 3+2 = 5$
$f(B) = 2+1 = 3.$

ii. Expand B to $f(G) = (2+3) + 0 = 5$

∴ Goal is found

Path is S → B → G

Cost is 5.

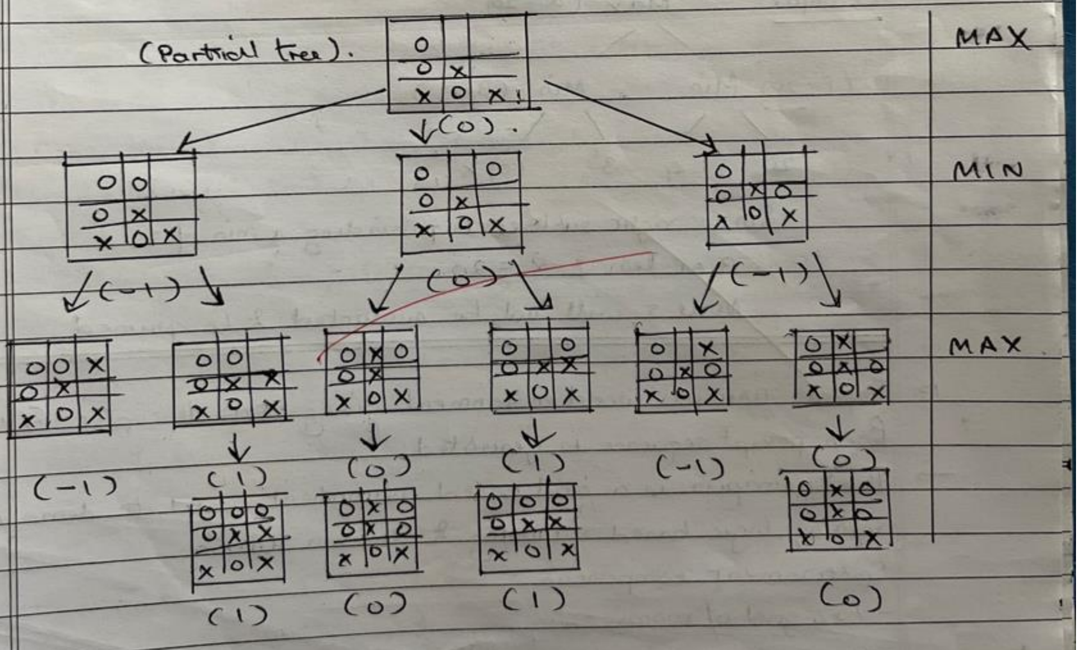
15 Explain minimax algorithm & draw game tree for Tictactoe.

→

minimax is a decision making algorithm used in game theory & AI
It helps in finding the optimal solution. by-

- player1 tries to max their score
- player 2 tries to minimize their score.

Algorithm:- Generate all possible states. [game tree].
- Assign scores to terminal nodes.
- Backtrack → Max picks the highest valued node & viceversa for min.
- Repeat until the root where Max chooses.



∴ Node 2 option has the most optimal choice.

16. Explain Alpha beta pruning algorithms for adversial search with example.

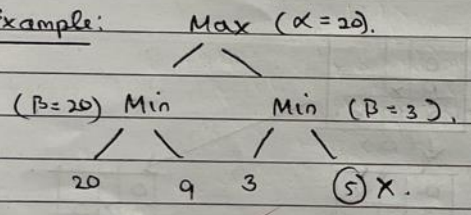
→

Its an optimization technique for minimax algorithm. It reduces no of nodes evaluated in a game tree by eliminating unnecessary branches.

Algorithm :- Set two values α & β for Max & Min respectivly.

- If α ≥ β for any subtree or node, stop evaluating and prune it.
- Prune all nodes which do not influence the final decision.

Example:



- Since right subtree is providing a min of 3.
  ∴ 3 or less & α = 20.
  ∴ Node 5 will not be evaluated & be pruned.

17. Explain wumpus world environment giving PEAS descriptor. Explain how percept sequence is generated.

→ The wumpus is a grid based environment used to demo AI agents, logic based reasoning & decision making.

Environment components -

- 4x4 grid of rooms.
- Rooms contain pits or the wumpus (beast).
- One room contains gold (goal).

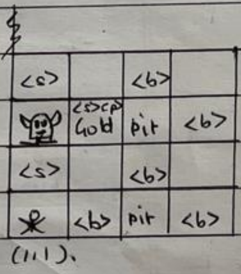
- Agent starts at (1,1). & can move horizontally & vertically.
- One arrow to kill the wumpus.
- Rooms surrounding the wumpus have stench & rooms surrounding the pit have breeze.

PEAS

Perf. measure.	Environment	Actuators	Sensors.
- Find gold in min actions. - Avoid wumpus & pits.	- 4x4 grid - wumpus, pits, gold.	- move front, back, up, down, - grab, shoot - climb.	- detect stench or breeze. - glitter (gold) - scream (kill wumpus)

Percept sequence :-

- Assume agent starts at (1,1) & gold at (3,2)
- Move to (1,2) → Perceive only breeze
- Move back & up (2,1) → Perceive only stench.
- Logically infer that (2,2) contains no pit or wumpus
- Similarly continue till reaching (3,2) goal.



<s>		<b>	
	<s>cp Gold	pit	<b>
<s>		<b>	
	<b>	pit	<b>

(1,1).

18. Solve the following crypto arithmetic problems

→

SEND + MORE = MONEY.

```
          +(1)  +(1)
      S    E     N     D
     (9)  (5)   (6)
 +    M    O     R     E
     (1). (0)   (8)   (5)
 ---------------------------
 M    O    N  .  E     Y
(1)  (0). (6)   (5).
```

Since carry is generated, $M=1$.

$\therefore$ $S+M = O$ generates carry

$\therefore$ $S+1 = O$

$\therefore$ $S=9$.

$\therefore$ $O = 0$

Now $E+O = N$

But $E \neq N$ $\because$ 2 numbers cannot have same value

$\therefore$ $E+O$+generated carry & $E+1 = N$.

$\therefore$ $N+R = E$, generates carry.

- Consider $R=8$ & $N=2$.

Result in $E=0$, not possible

→ - If $N=3$, Result in $E=1$, not possible

- Only case possible is

$N=6$ & $E=5$.

- Now $D+5 = Y$ should generate carry

$\therefore$ $D=7$ & $Y=2$.

0	O
1	M
2	~~N~~ Y.
3	
4	
5	E
6	N.
7	D
8	R
9	S

$\therefore$ $9567 + 1085 = 10652$

19. Consider the following axioms:

→

All people who are graduating are happy.

$\forall x\,(\text{graduating}(x) \rightarrow \text{happy}(x))$.

All happy people are smiling

$\forall x\,(\text{happy}(x) \rightarrow \text{smiling}(x))$.

Someone is graduating $\exists x\,(\text{graduating}(x))$.

- Convert to clause form.

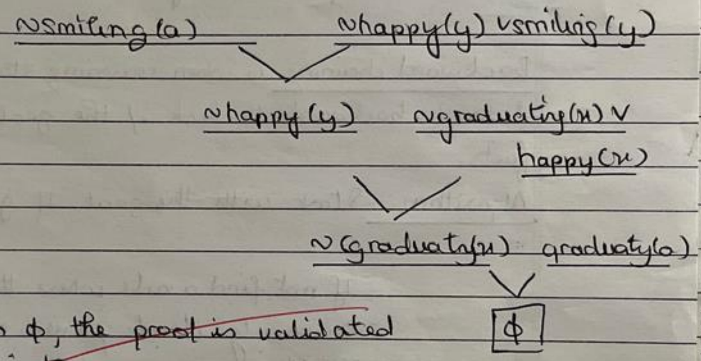
  ~graduating (x) v $\overset{happy}{\cancel{smiling}}$ (x).

  ~happy (y) v smiling (y).

  graduating (a).

- Prove "is someone smiling?" → ~smiling (a).

- Resolution tree:



Since the tree results in φ, the ~~proof is~~ validated

∴ Someone is smiling ~~is true~~.

20. Explain modus ponen with suitable example

→ Its a fundamental rule of logic stating P→Q = P ⊢ Q.

If P implies Q. & P is True, then Q also must be true

example:

- If it rains, the ground will be wet.

  Rains → wetground

- It is raining.

  Rains → True

∴ The conclusion is that the ground is wet.

2. Explain forward & backward chaining with the help of example.

→ Forward chaining is when reasoning starts with known facts & applies inference rules to new facts until the goal is reached.

Algorithm:- Start c̄ a set of known facts.
- Apply inference rules that match current facts.
- Derive new facts & add them to the knowledge base
- Repeat until goal is achieved.

Backward chaining is when reasoning starts with the goal (query) & works backwards to check if the goal can be derived from known facts.

Algorithm:- Start with the goal. If goal is a known fact, return true
- If not find a rule where the goal appears as conclusion
- Recursively check premises of the rule
- If all premises are true, return goal = true.

Example: i If person has fever & cough they have flu.
ii If person has flu, they should take rest.
Facts:- John has fever. John has cough
Prove - John should take rest.

→ Forward chaining: Fever (John) , Cough (John)
Fever(x) ∧ Cough (x) → Flu(x)
Flu (John) → Rest (John)

→ Reverse chaining: Rest (John)
∴ Flu(John)
∴ Fever(John) & Cough(John) both are true, hence statement is proved.