Name: Shejal Nilesh Tiwari
D15C Roll no: 57

AIDS Assignment - 1

Q.1] What is AI? Considering the covid-19 pandemic situation, how AI helped to survive and renovated our way of life with different applications?

AI (Artificial intelligence) refers to the simulation of human intelligence in machine enabling them to learn, reason and solve problems. AI encompasses machine learning natural language processing, robotics and expert systems.

AI's role in the COVID-19 pademic

1) Healthcare: AI powered tools helped in diagnosing COVID-19 using chest x-rays, predicting outbreaks and accelerating vaccine development.
2) chatbot → AI driven chatbots provided real time information and assisted in symtom checking.
3) Supply Chain Management → AI optimized supply chain by predicting shortage of medical equipment.
4] Drug discovery → AI assisted in analysing protein structure to identify potential treatments.

Q.2 What are AI agents terminology, explain with example?

An AI agents perceives its environment using sensors and acts upon it using actuators

Key terminology:

1) Agent → An entity that senses and acts.
Ex- a self-driving car.
2) Sensors → Devices used to perceive the environment (ex. camera).
3) Actuators → Components responsible for actions. (eg:- robot arms in automation)

Q.3. How AI techniques is used to solve 8 puzzle problem?

The 8- puzzle consists of a 3X3 grid with numbered tiles and an empty space and the goal is to arrange the tiles in order

AI techniques used:-

1] Breadth - First Search (BFS) → Explore All possible moves at each step.
2] Depth - First search (DFS) → Explores a single path deeply before backtracking.
3] A* Algorithm → Uses heuristics to find the shortest path.
4] Best-First Search → Select moves that bring it closest to the goal state.

Q.4 What is PEAS descriptor? Give PEAS for the following PEAS is used to describe and analyze the task environment of an AI agent.

System	Measure	Environment	Actuator
Taxi driver	Safety time customer satisfaction	Roads, traffic passengers	Steering breaking accelerati-
Medical diagnosis	Accuracy of diagnosis speed	Patient data symptoms	Display recommend ation
Music Composer	Quality of composition popularity	Music industry users	Sound generation editing tools.

Sensors
GPS, speedometer
camera.

Patient records
medical tests

musical notes
users preference

Q.5 Categorize a shopping bot for an offline bookstore according to each of the six dimensions:-

Dimension	Categorization
Observability	Partially observable (it doesn't known all book stocks instantly)
Determinism	stochastic (stock availability danger unpredictably).
Episodic vs sequential	Sequential (it learns from ~~path~~ user interactions).
Static vs Dynamics	Dynamic (book availability and price changes)
Discrete vs Continuous	Discrete (fixed set of book categories & user queries).
Single vs Multi-Agents	Multiagent (interacts with customers and inventory databases).

Q.4 What is PEAS descriptor? Give PEAS for the following PEAS is used to describe and analyze the task environment of an AI agent.

System	Measure	Environment	Actuator
Taxi driver	Safety time customer satisfaction	Roads, traffic paissenger	Steering breaking accelerati
Medical diagnosis	Accuracy of diagnosis speed	Patient data symptoms	Display recommend ation
Musir Composer	Quality of composition popularity	Music industry users	Sound generation editing tools.

Sensors
GPS, speedometer
camera.

Patient records
medical tests

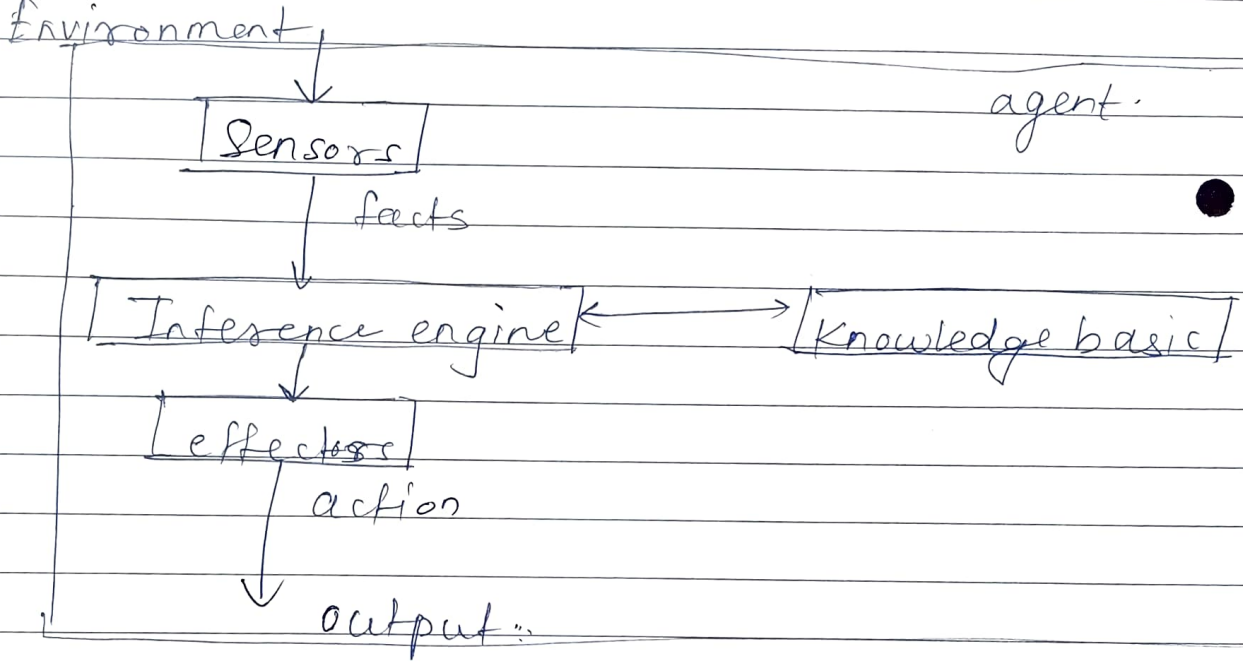
musical notes
users preference

Q.6 Differentiate model based and utility based?

Feature	Model based	Utility based
Definition	Uses an internal representation of the environment to make decisions	Chooses actions based on a utility based functions that maximizes the best possible outcome.
Decision making	Relies on stored knowledge about how the environment behaves.	Selects actions based on numerical values assigned to outcomes.
Complexity	moderate, as it requires a well defined model of the environment	High, as it involves calculating and comparing multiple possible outcomes.
Adaptability	Less adaptive, as it depends on pre-defined rules	more adaptive as it evaluates outcomes dynamically.

Q.7 Explain the architecture of a knowledge based agents & learning Agent

* knowledge based agent



A knowledge based agent includes a knowledge base and an inference engine system. A knowledge base is a set of representations of facts of the world.

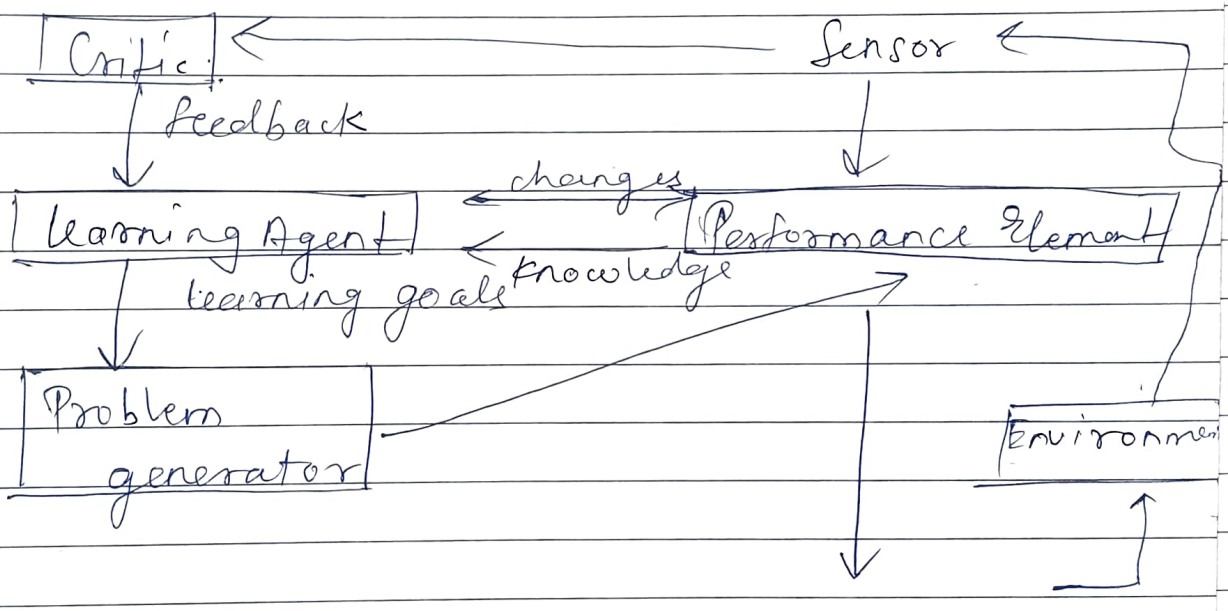
The agent operates as follow :-

1) It tells the knowledge base what it perceives.
2) It ASKs, the knowledgebase what action it should perform.
3) It performs the choosen action.

Learning Agent:

By actively exploring and experimenting with their environment, the most powerful agents are able to learn.

A agent can be further divided into 4 conceptual components.



Q.8 Convert the following to predicates

→ Lets define predicates

- Travels (x, y)
- Available (y)
- Goals-via (y, z)
- Puncture (y) → y has a puncture
- Not (Available (y1)) → y is not available

a] Anita travels by car if available, otherwise by bus

- Available (car) → Travel (Anita, Car)
- Available (car) → Travels (Anita, Bus)

b] Bus goes via Andheri and Goregaon
- Goes – via (Bus, Andheri)
- Goes – via (Bus, Goregaon)

c] Car has a ~~particular~~ puncture, so it is not available

- ~~Puncture~~ (Car)
- ~~Puncture~~ (car) → Available (car)

- ~~Forward Reasoning to Determine Anita's~~ Travel Route

1] Given Puncture(Car), we apply the rule ~~puncture~~ (car) → ¬ Available (car) so:-
Available (car) (Car not available)

2] From – Available (car) → Travels (Anita, Bus), we infer:
- Travels (Anita, Bus) · (Anita travels by Bus).

3] Given does_via (Bus, Goregaon), we know
- The bus passes through goregaon.

Since we found G, we trackback the path using the present mapping.

Step 2:
The shortest path from S to G is
$S \rightarrow B \rightarrow G$

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

→ Depth limited search (DLS)
It is a variant of depth first search that restricts the depth of recursion to fixed limit L. In avoids infinite loop search spaces.

Algorithms:
1] Start from the initial node and explore deeper until depth L is searched.
2] If goal is found, return the solution.
3] If the depth limit is reached without finding the goal, backtracks.
4] If

Since we found G, we trackback the path using the present mapping.

Step 2:
The shortest path from S to G is
$S \rightarrow B \rightarrow G$

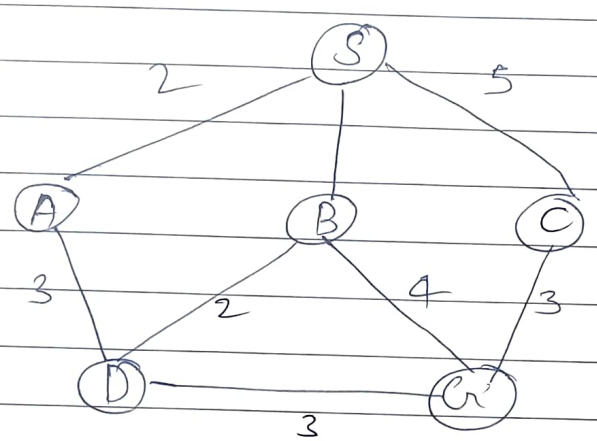
Q.11. What do you mean by depth limited search? Explain iterative deepening search with example.

→ Depth-limited search (DLS)
It is a variant of depth first search that restricts the depth of recursion to fixed limit L. In avoids infinite loop search spaces.

Algorithms:
1] Start from the initial node and explore deeper until depth L is searched.
2] If goal is found, return the solution.
3] If the depth limit is reached without finding the goal, backtrack.
4] If

4) Since Anita is travelling by bus and Bus goes via Goregaon we will conclude:
- Anita will travel via Goregaon.

Q.10 Find the route from S to G



BFS explores the graph level by level, starting from the source node S.

Step 1:

i] Start at S: Add S to the queue
- Queue: [S]

Expand S: Visit its neighbors A, B, C
- Queue: [A, B, C]
- Parents mapping
S → A, S → B, S → C

Expand A: Visit D
- Queue [B, C, D]
- Parents mapping: A → D

Expand B: Visit G (Goal found)
- Queue: [C, D, G]
- Parent Mapping: B → G.

Since we found G, we traceback the path using the present parent mapping. The shortest path from S to G is $S \rightarrow B \rightarrow G$

Q.11] What do you mean by depth limited search? Explain Iterative deepening search with examp

Depth limited search:

1] start from the initial node and explore deeper until depth L is searched.
2] If goal is found, return the solution
3] If the depth limit is reached without finding the goal, backtrack.
4] If no solution is found at all, increase the depth limited or choose another strategy.

Iterative Deepening Search (IDS).

Iterative Deepening search combines the advantages of both BFS and DFS. It seperatedly runs Depth limited search with increasing depth limits until the goal is found.

12] Explain Hill climbing and its drawbacks in detail with example. Also state limitations.

→ Hill climbing is a local search algorith that continuously moves toward higher valued states in search space to find the optimal solution. It is widely used in optimization problem.

Algorithm :-

1] Start from an initial state
2] Evaluate all neighbouring states.
3] Move to the neighbour with highest value.
4] Repeat until no better neighbour is found.

Q.13] Explain simulated annealing and write its algorithm.

→ Simulated Annealing is an optimization technique inspired by the cooling process of metals. Unlike hill climbing it can escape local maxima by allowing some "bad" moves with a probability that decreases over time.

Algorithm :

1] Start with an initial state S and high temperature T.
2] Generate a random neighbour S'.
3] If S' is better than S, move to S'
4] If S' is worse, accept it with probability $e^{-\Delta E/T}$ where ΔE
5] Reduce T gradually
6] Repeat until $T \to 0$ or a solution is found.

Example :- In travelling Salesman Problem (TSP) to optimize the shortest route while avoiding local optimal.

Q.15] Explain Minmax. Explain MinMax Algorithm & draw game tree for TIC TAC TOE.

→ MinMax is an adversarial search algorithm used in zero sum games like TIC-TAC-TOE chess and checkers. It assumes two players.

Maximizer (tries to maximize score)
Minimizer (tries to minimize score).

Algorithm steps.

1] Generate the game tree up a depth where a terminal state

Q.14] Explain A^* algorithm with an example:

→ A^* is a search algorithm used in pathfinding and graph traversal. It finds the shortest path from the start node to the goal node using both.

- $g(n)$ → The actual cost from the start node to node n.
- $h(n)$ → The estimated cost from node n to goal.
- $f(n) = g(n) + h(n)$ → The total estimated cost of the cheapest path through n.

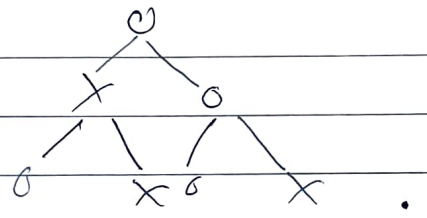
Algorithm:

1] Place the start node in an open list.
2] Move the node with the lowest $f(n)$ from the open list to the closed list.
3] If S' is better than S, move to S'.
4] If S' is worse accept it with probability $e^{-[\Delta E/T)}$ where ΔE is difference in values.
5] Reduce T gradually.
6] Repeat until $T \to 0$ or a solution is found.
3] Expand the node's neighbour, update their $g(n)$ & $h(n)$ values.
4] Repeat until the goal node is reached.

Q.15 Explain MinMax. Explain MinMax Algorithm & draw game tree for TICTACTOE.

→ MinMax is an adversarial search algorithm used in zero games like TICTACTOE, chess and checkers. It assumes two players.

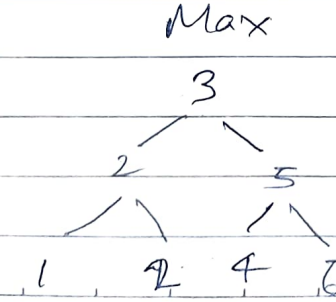
- Maximizer (tries to maximize score)
- Minimizer (tries to minimize score).



Q.16 Explain Alpha beta pruning algorithm for adversarial search with example.

→ Alphabeta pruning improves Minimax by pruning unnecessary branches, reducing computation time without affecting the final result.

- Alpha (α): Best choice for the maximizer
- Beta (β): Best choice for the minimizer.



Q.17] Explain WUMPUS world environment giving its PEAS description. Explain how percept sequence is generated.

Stench		Breeze	Pit
Wumpus	Breeze Stench Gold	Pit	Breeze
Stench		Breeze	Breeze
Start	Breeze	Pit	

Environment Details:

- The grid consist of rooms connected by doors (4×4).
- The wumpus is in one of the room.
- There are pits which the agent must avoid.
- The agent can sense danger using percept..
  - i] Stench → Wumpus nearby
  - ii] Breeze → Pit is nearby
  - iii] Glitter → Gold is present

q.18] Solve the following Crypto-arithmetic problems.

```
  SEND
  MORE
 -----
 MONEY.
```

C_4	C_3	C_2	C_1	
	9	5	6	7
	1	0	8	5
M 1	O 0	N 6	E	Y 2

$S + M = O$

$S + 1 = O$

$\therefore S = 9 \quad O = 0$

$E + 0 = N$

$E + 0 = N$

$E = N.$

$E = 5$

$E + 0 + C_2 = N$

$N = 6.$

Q.21

Q.19] Consider the following axioms

All people who are graduating are happy. All happy people are smiling someone is graduating.

Also explain the following

Step 1 : Convert to Clause form.

1] Graduating (x) → Happy(x) → Graduating (x) ∨ Happy(x).

2] Happy (x) → Smiling (x) → Happy (x) ∨ Smiling(x)

3] Graduating (x) → Introduce constant 'a' → Graduating (a)

Step 2 : Prove "is someone smiling ?"

1] Graduating (a) - - - Given

2] Graduating (x) ∨ Happy (x)

- Substitute x = a : Graduating (a) ∨ Happy (a).
- Since Graduating (a) is true, reso to Happy (a).

3] → Happy (x) V Smiling (x)

- Substituting x = a :→ Happy (a) V Smiling (a)
- Since Happy (a) is true, resolve to smiling
  Thus, someone is smiling (Smiling (a)).

Q.20] Explain Modus Ponen with suitable example.

Modus Ponen is a fundamental rule of inference in logic that states:

If "P → Q" (If P then Q) is true and P is true, then Q must be true.

It follows this logical form.

1] P → Q (If P then Q)
2] P (P is true)
∴ Q (Therefore, Q is true).

Example:

- If it rains, the ground will be wet (P → Q)
- It is raining (P is true).
- Therefore, the ground is wet. (Q is true

Q.21] Explain forward chaining and backward chaining algorithm with the help of example.

→ • Forward Chaining

Starts from known facts and applies inference rule to reach a conclusion.
It is also called as data driven approach

Algorithm

1] Start with given facts is a knowledge base
2] Apply rules to infer new facts.
3] Repeat untill the goal / conclusion is reached.

Example:
Rules :-
1] If a person has a fever and cough, then may have the flu.
2] If a person has a score throat, they may have a cold.

Facts:
1] John has a fever and cough.
2] Apply rule 2 → John may have the flu

Conclusion: The system infers John has the