Name : Sandeshkumar M. Yadav   Div : D15C
Roll : 61 04/05
Assignment - 1   Subject : AIDS - 1

Q1. What is AI? Considering the Covid-19 Pandemic Solution, how AI helped to Survive and renovated our way of life with different applications?

→ Artificial Intelligence (AI) enables machines to think, learn, and make decisions. It includes machine learning, robotics, and natural language processing, widely used in various industries.
AI's Role in Covid-19:
Healthcare : AI-assisted diagnosis, drug discovery, and chatbot symptom checkers.
Contact Tracing : AI-powered apps tracked virus spread and predicted outbreaks.
Remote work : AI enhanced virtual meetings and automated customer support.
Supply chain : AI optimized logistics and enabled contactless deliveries.
Mental well-being : AI chatbots offered mental heath support; streaming platforms personalized entertainment.

Q2. What are AI Agents terminology, explain with examples?

→ An AI agent percieves its environment, processes data, and takes action to achieve goals. It operates autonomously or Semi-autonomously based on rules or learned pattern.
Key Terminologies of AI Agents :
1. Agent : An AI system tat percieves and acts in an environment
e.g. A self-driving car that observes traffic and adjust speed accordingly
2. Environment : The external world in which an agent operates.
e.g. for chess playing AI, the chessboard is its environment.
3. Perception : The process of gathering data from the environment using sensors. E.g. A facial recognition camera.

4. Sensors : Devices that allow an agent to receive input from the environment e.g. A robot vacuum's infrared sensors detect obstacles.

5. Actuators : Component that allow an agent to take actions in it environment e.g. The robotic arms of an assembly line move objects.

Q3. How AI technique is used to solve 8 puzzle problem?

→ The 8-puzzle problem is ~~[illegible]~~ sliding puzzle that requires arranging tiles in a specific order by moving them into an empty space. AI techniques solve this problem using Search Algorithms :

1. Uninformed search :
   - Breadth first Search : Explores all possible moves level by level
   - Depth first search : Explores move deep into the search tree befor backtracking.

2. Informed Search :
   - Best first Search : uses heuristic to prioritize moves.
   - A* : uses a Cost function $f(n) = h(n) + g(n)$.

Q4. What is PEAS descriptor? ~~[illegible]~~ Give PEAS descriptor for following : Taxi Driver, Medical diagnosis system, A music Compresser, An aircraft autolander, An essay evaluator, A robotic sentry gun for the Keck lab

→ The PEAS descriptor is used to define the components of an AI agent in a structured manner. It helps in understanding how an AI System operates in its environment.

PEAS Descriptors for Different AI Systems:

1. Taxi Driver
P → Safety, passenger satisfaction, fuel efficiency.
E → Roads, traffic, passengers.
A → steering wheel, accelerator, brakes.
S → GPS, Cameras, Speed Sensors.

2. Medical Diagonasis System
P → Accuracy of diagnosis, treatment efficiency, patient satisfaction
E → medical databases, patient Symptoms, test results.
A → Display Screen, prescriptions, treatment recommendations.
S → patient data inputs, lab test result, medical images.

3. AI music Composer
P → Musical harmony, originality, audience engagement.
E → music genres, user preferences, existing compositions.
A → Digital instrument, speakers,
S → user-feedback, music databases, genre analysis.

4. Aircraft Autolander
P → Smooth landing, passenger Safety, precision.
E → Weather Conditions, runway Status, altitude.
A → Landing gear, throttle, flaps, brakes.
S → GPS, altimeter, wind sensors, Speed Sensors.

5. Essay Evaluator AI
P → Accuracy of grading, fairness, feedback quality.
E → Essays, grammer rules, evaluation criteria.

Q6. Differentiate between Model based and utility Based Agent?

→

Model Based Agent	utility Based Agent.
Maintains an internal model of the environment to make decisions.	chooses actions based on a utility ~~based~~ function that measures performances.
Uses stored knowledge of the world to predict outcomes	Select the action that maximizes overall benefit or Satisfaction.
e.g. GPS navigation predicting traffic conditions.	e.g. stock trading bot optimizing investment returns.
Moderate - requires maintaining an internal model.	High - needs continuous evaluation of different actions.

Q7. Explain the architecture of a knowledge based Agent and Learning Agent.

→ A knowledge base Aged uses a structural knowledge base to make informed decisions.

Components :

1. knowledge Base → Stores facts and rules about the environment.
2. Inference Engine → Derives new facts using logical reasoning.
3. Perception → Collects information from the environment.
4. Actuators → Executes decision based on knowledge and inference.

A Learning agent improves its performance over time through experience.

Components :

1. Learning Element → Adapts based on feedback.
2. Performance Element → Makes decisions based on learned knowledge.
3. Critic → Evaluates actions and provides feedback.
4. problem Generator → Suggests new experiences to improve learning.

Q9. Convert the following to predicates :

a. Anita travels by Car if available otherwise travels by bus.

b. Bus goes via Andheri and Goregaon

c. Car has punture so is not available.

Will Anita travel via Goregaon? Use forward reasoning.

→ 1. Given statements in predicate form :

(a) Travels (Anita, Car) :- Available (car).

Travels (Anita, Bus) :- ¬ Available (Car).

(b) GoesVia (Bus, Andheri).

GoesVia (Bus, Goregaon).

(c) Puncture (car).

¬ Available (car).

Forward Reasoning : Will Anita Travel via Goregaon?

1. Car has a puncture : Puncture (car) → ¬ Available. (car).
2. Car is not available → Anita travels by bus : ¬Available (Car) → Travels (Anita, Bus).
3. Bus goes via Goregaon : GoesVia (Bus, Goregaon).
4. Since Anita travels by bus and the bus goes via Goregaon, Anita will travel via Goregaon.

Yes, Anita will travel via Goregaon.

Q10. find the route from S to G using BFS.

→ Breadth first Search explores all neighboring nodes before moving to the next level. It follows a FIFO approach.

from the given graph, the nodes and their connection are,

S → { A : 2 , B : 5 , C : 5 }
A → { D : 3 }
B → { D : 2 , G : 4 }
C → { G : 3 }
D → { G : 3 }

BFS Traversal from S to G,

1. Start at S. Queue [S].
2. visit S, add its neighbors (A, B, C), . Queue [A, B, C]
3. Dequeue A, add its neighbor (D). Queue [B, C, D].
4. Dequeue B, add its neighbors (D, G). Queue [C, D, G].
5. Dequeue C, add its neighbors . Queue [D, G].
6. Dequeue D, add its neighbor (G). Queue [G].
7. Dequeue G → Goal Reached.

Shortest path from S to G.
from BFS traversal, the shortest path found is:
S → B → G (cost = 5+4 = 9).

Q11. What do you mean by depth limited Search ? Explain Iterative Deepening search with example!

→ DLS is a variation of DFS where a depth limit is set to prevent exploring beyond a certain level.
It avoids infinite loops in infinite state spaces but may fail to find a solution if it's deeper than L.
e.g. In a maze, if we set L=3, the search will not explore paths beyond depth 3, even if the goal exists at depth 4.

Iterative Deepening Search (IDS).
IDS repeatedly applies DLS, increasing the depth limit step by step until the goal is found.
Steps:
1. Start with L=0 and perform Depth Limited Search.
2. If no solution, increase L→L+1 and repeat.
3. Continue until the goal is found.
e.g.
For a goal node at depth 4, IDS runs:
- DLS (L=0) → No solution
- DLS (L=1) → No Solution
- DLS (L=2) → No Solution
- DLS (L=3) → No solution
- DLS (L=4) → Goal found.

It is used in AI problem-solving, game trees, and robot pathfinding.

Q12. Explain Hill Climbing and its drawbacks ~~with~~ in detail with example. Also state limitations of steepest-ascent hill climbing?

→ Hill climbing is a heuristic search algorithm that continuously moves towards a better solution by selecting the neighbor with the highest value (better heuristic).

Steps :

1. Start from an Initial solution.
2. Evaluate all neighboring solution.
3. Move to the neighbor with the highest improvement.
4. Repeat until no better neighbor exists.

Example :

In route optimization, Hill climbing can be used to find the shortest path by always selecting the next closest city. However, it might get stuck if a better route exists beyond the immediate neighbor.

Drawbacks :

- Local maxima → can get stuck in a suboptimal peak.
- plateau → stops if all neighbours have equal values.
- Ridges → struggles with diagonal movement.
- No Backtracking → Doesn't reconsider previous paths.

Limitations of Steepest - Ascent Hill climbing :

- High computation → Evaluates all neighbors, slowing performance.
- Local maxima issue → Ignores long-term better paths.
- plateau Stagnation → May halt without progress.
- Ridge Navigation → Struggles with multi-step improvement.

Solution → use Simulated Annealing or Genetic Algorithm for better exploration.

Q13. Explain Simulated annealing and write its algorithm?

→ Simulated annealing is an optimization algorithm that helps escape local optima by occasionally accepting worse solutions based on a probability function. Inspired by metallurgical annealing, it gradually reduces the probability of accepting worse solutions over time.

Algorithm :

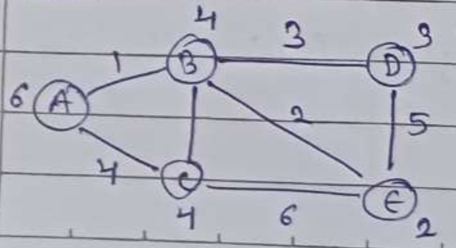
1. Set an initial solution S, temperature T, and cooling rate α.
2. Repeat until stopping condition :
   Generate a new neighbor S'.
   Compute change in cost ΔE = Cost (S') - Cost (S).
   If S' is better, accept it (S = S').
   If S' is worse, accept it with probability $p = \exp(-\Delta E / T)$.
   Decrease T using $T = \alpha * T$.
3. Return best found solution.

Q14. Explain A* Algorithm with an example?

→ A* is widely used search algorithm for finding the shortest path in a graph. It is an informed search algorithm that combines.

1. g(n) → The cost from the start node to node n.
2. h(n) → A heuristic estimate of the cost from node n to goal.
3. f(n) = g(n) + h(n) → The total estimated cost.

A* Select the node with the lowest f(n) value to expand next.



pathfinding from A to E:

1. Start at A, calculate $f(A) = g(A) + h(A) = 0 + 6 = 6$.
2. Expand $A \rightarrow \{B(g=1), C(g=4)\}$.
   $f(B) \rightarrow 1 + 4 = 5$
   $f(C) \rightarrow 4 + 4 = 8$
   choose B (lowest f-value)
3. Expand $B \rightarrow \{D: 4, E: 3\}$
   $f(D) \rightarrow 4 + 3 = 7$
   $f(E) \rightarrow 3 + 2 = 5$
   choose E (goal reached).

final path
$A \rightarrow B \rightarrow E$ with cost 3.

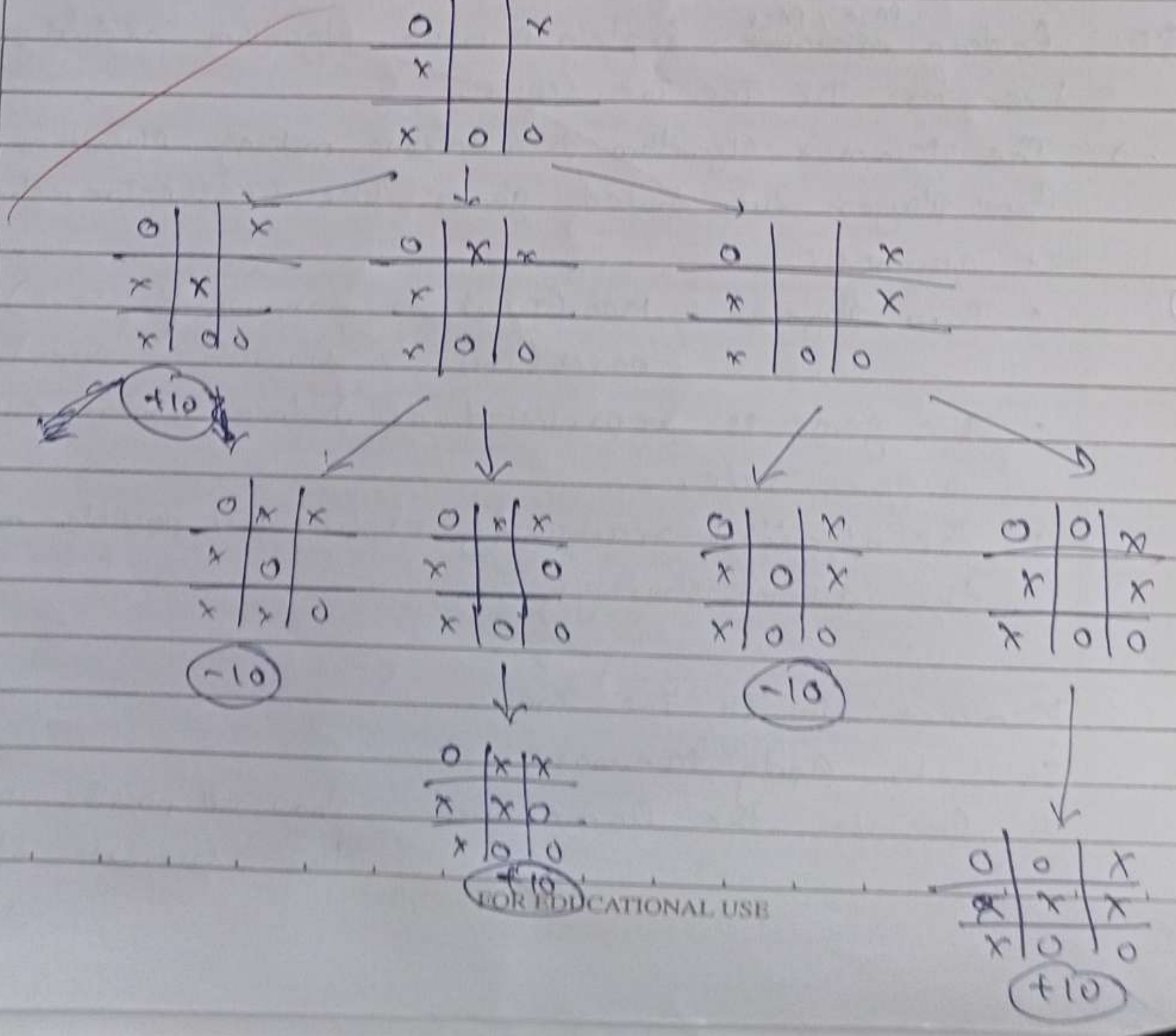
Q5. Explain ~~minimax~~ Min, Max, Explain Minimax Algorithm and draw game tree for Tic Tac Toe Game.

→ The Minimax Algorithm is a decision making Algorithm used in two player turn based games like Tic-Tac-Toe, chess.
It assumes:
- Two players : Max (Tries to maximize the score), MIN (Tries to minimize the score).
- The game is represented as a tree where each node is a game state.
- The algorithm recursively explores all possible moves to find the optimal strategy.

Minimax in Tic Tac Toe :
Steps to Apply Minimax:
1. Generate the Game Tree : show all possible moves from the current state.

2. Assign Score.
   - +1 for a win.
   - -1 for a loss.
   - 0 for a draw.
3. Backpropagate values :
   - Max chooses maximum.
   - Min choose minimum.
4. Select the Best move :
   max plays optimally by choosing the move leading to the highest Score.

Game tree for Tic Tac Toe,



Q16. Explain Alpha beta pruning algorithm for adversial search ~~[illegible]~~ with example?

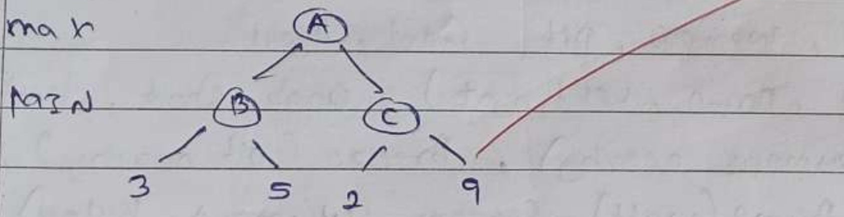
→ Alpha-Beta pruning is an optimization of the minimax Algorithm that eliminates unnecessary branches, improving efficiency, without affecting the result.

- α → Best Score ~~[illegible]~~ MAX can guarantee.
- β → Best Score MIN can guarantee.
- pruning Condition → If $\beta \le \alpha$, stop further exploration.

Algorithm:

1. perform minimax Algorithm while tracking α and β.
2. Prune branches where further evaluation won't affect the outcome.
3. Continue recursively until a terminal state is reached.

E.g.



Execution :

1. Evaluate (B) : MIN picks 3 (min of 3,5) → $\beta = 3$.
2. Evaluate (C) : ~~[illegible] [illegible] [illegible]~~ first node is (2), smaller than (3) → prune 9 (since MIN would never pick it).
3. Max pick max(3,2) → 3.

Final outcome:

- without pruning Evaluate 4 nodes.
- with pruning Evaluate 3 node (prunes 9).

Faster decision making by reducing space, optimal result same as minimax.

Q7. Explain Wumpus world environment giving its PEAS description. Explain how percept sequence is generated?

→ Wumpus world is a grid-based, partially observable, stochastic environment used in Artificial Intelligence to demonstrate intelligent agent behaviour.

- The world is 4x4 grid where an agent (player) must find gold while avoiding hazards.
- Hazards :
  - Wumpus (A monster that kills the agent if encountered).
  - Pits (deadly holes .
- Goal : Grab gold and escape safely.

PEAS Description :

P → +1000 for getting gold, -1000 for wumpus or pit, -1 per move -10 for shooting.

E → 4x4 grid, wumpus, pit, Gold, Agent.

A → Move (up, Down, Left, right), Grab, shoot, climb.

S → Stench ~~[illegible]~~ (wumpus nearby), Breeze (pit nearby), Glitter (Gold), Bump (wall), Scream (Wumpus killed).

How percept Sequence is Generated :

Example ~~Step~~ percept sequence,

Step	Agent's location	percept Received
1	(1,1) (start)	None
2	(2,1)	Breeze
3	(1,1)	None
4	(1,2)	stench
5	(1,3)	stench, Glitter (Gold found).

The agent uses a logical inference to avoid pits/Wumpus while searching for gold.

Q18. Solve the following Crypto-arithmetic problems
SEND + MORE = MONEY.

→ We need to find digits such that:
1. No two letters map to same digit.
2. Leading letters (here S and M) cannot be zero.
3. The sum SEND + MORE = MONEY must hold true arithmetically.

Solution:

1. We know M & S cannot be 0 as they are leading letters, The result is 5 digit, Implying M = 1.
$\boxed{\therefore M = 1}$.

2. Now S+M produces a carry, for a result to be a 5-digit, there must be carry of 1 out of that column:
$S + 1 \geq 10 \rightarrow S = 9.$
$\boxed{\therefore S = 9}$

remaining letter, {0, 2, 3, 4, 5, 6, 7, 8}.

3. ones column, D + E = Y or Y + 0.
The result as the ones digit Y (and possibly a carry into the tens).

4. Tens column, N + R + (Carry from ones) = E or E + 10.
~~wate~~ The result digit is E (and possibly a carry into the hundreds).

5. hundreds colum, E + O + (Carry from tens) = N or N + 10.
The result digit is N (and possibly a carry into the thousands).

6. By Systematically trying digits 0-9 (with logic or backtra
we find the consistent assignment.
E = 5, N = 6, D = 7, O = 0, R = 8, Y = 2.

7. Hence the final solution becomes,

	S (9)	E (5)	N (6)	D (7)	
+	M (1)	O (0)	R (8)	E (5)	
M (1)	O (0)	N (6)	E (5)	Y (2).	

SEND → 9567
MORE → 1085
MONEY → 10652.

Q. 19. Consider the following axioms:
All people who are graduating are happy.
All happy people are smiling.
Someone is graduating.
Explain the following:
1. Represent these axioms in FOPL.
2. Convert each formula to clause form.
3. prove that " IS Someone Smiling ? " using resolution technique. Draw the resolution tree.

→ 1. Represent the axioms in FOPL.
Let, $G(x)$: " x is graduating ".
$H(x)$: " x is happy ".
$S(x)$: " x is smiling ".
Domain of discourse : All " people ".

FOL Representation :

1. All people who are graduating are happy.
   $\forall x\ G(x) \rightarrow H(x)$
2. All happy people are smiling.
   $\forall x\ H(x) \rightarrow S(x)$.
3. Someone is graduating.
   $\exists(x)\ G(x)$.

Goal : $\exists x\ S(x)$ (i.e. Someone is smiling).

2. Clause form.
 1. from $\forall(x)\ [G(x) \rightarrow H(x)]$:
    $(\neg G(x) \vee H(x))$.
 2. from $\forall(x)\ [H(x) \rightarrow S(x)]$:
    $(\neg H(x) \vee S(x))$.
 3. from $\exists x\ G(x)$,
    $G(a)$.

 4. $\neg S(a)$.

To prove : ~~$\exists x\ S(x)$,~~
 $\forall(x) \neg S(x)$. or $\neg S(a)$.

3. ~~[illegible]~~ Resolution proof.
 - Resolve ① & ③
   $G(a)$ with $\neg G(x) \vee H(x) \rightarrow H(a)$.
 - Resolve $H(a)$ with ②.
   $\neg H(x) \vee S(x) \rightarrow S(a)$.
 - Resolve $S(a)$ with ④
   $\neg S(a) \rightarrow$ Contradiction $(\perp)$.

Because we derive $\perp$, $\exists(x)\ S(x)$ holds.
Conclusion : Someone is Smiling.

Q20. Explain modus ponen with suitable example.

→ modus ponen is a basic rule of inference in propositional logic. It allows you to derive a conclusion Q from two premises:

1. A conditional statements: "If P then Q", $(P \to Q)$.
2. A factual statement: P is true.

Result: from these two premises, we infer Q is true.

Example:

1. Premise (conditional):
   "If a student scores above 90, they get an A" $(S \to A)$.
2. premise (fact):
   "Alice scored above 90". (S).
3. Conclusion:
   "Alice gets an A." (A).

Here, the fact S ("Alice scored above 90") triggers the conditional rule $S \to A$ ("If above 90, then A"). leading to the conclusion A. ("Alice gets an A").

Q21. Explain forward chaining and backward chaining algorithm with the help of example.

→ forward chaining: Starts with known facts, applies rule to derive new facts, and continues forward until a goal is reached or no ~~[illegible]~~ new facts emerge.

process:

1. Begin with a set of facts.
2. Match facts to rule antecedents (If parts).
3. fire matching rules to derive new facts.
4. Repeat until goal is found or no more inferences be made.

e.g.

Facts : SunlightPresent, SoilMoist.

Rules :

1. If SunlightPresent AND SoilMoist → plantGrows.
2. If plantGrows → GardenlooksGreen.

Inference :

from the Initial facts, deduce plantGrows, then GardenlooksGreen.

Backward chaining : starts with a goal, works backwards to find facts or subgoals that supports it.

process :

1. Identify the goal.
2. find rules that conclude the goal.
3. Derive Subgoals from the rule's Conditions.
4. prove each subgoal by existing facts or further backword chaining.
5. Succeed if all subgoals are proven.

e.g.

- Goal : prove GardenLooksGreen.
- check rule concluding GardenLooksGreen → Subgoal plantGrows.
- Subgoal [illegible] plantGrows → prove SunlightPresent and SoilMoist.
- If both facts are true, conclude PlantGrows, then GardenLooks - Green.

Key differences :

- Forward chaining : Data driven ; start with facts ; derive Conclusion.
- Backward chaining : Goal-driven ; Start with the desired Conclusion ; find supporting facts.