Name - Anish Kulkarni
Roll No. - 29
Class - D15C

05/05

AIDS Assignment - 1

Q1) What is AI? Considering the COVID-19 pandemic situation, how AI helped to survive and renovated our way of life with different applications?

Ans) Artificial Intelligence (AI) enables machines to perform human-like tasks such as problem-solving, learning, and decision-making using algorithms and data-driven techniques.

AI helped during the COVID-19 pandemic in the following ways:-

i) Healthcare: Predicted outbreaks, diagnosed cases, and aided drug discovery.

ii) Contact Tracing: AI apps and thermal cameras tracked infections.

iii) Remote Work: AI chatbots and tools improved virtual collaboration.

Post-pandemic, AI transformed telemedicine, remote work, e-learning, automation, and entertainment, becoming a vital part of daily life.

Q2) What are AI agents terminology? Explain with examples.

Ans) AI agents are systems that perceive their environment, process information, and take actions to achieve specific goals. Key terminologies include:

i) Agent - An entity that interacts with the environment (e.g. a self driving car).

ii) Environment : The external system in which the agent operates (e.g. traffic conditions for a self-driving car).
iii) Perception - Data collected from sensors (e.g. a robot's camera detecting obstacles).
iv) Actuators - Components that execute actions (e.g. robotic arms in manufacturing).
v) Rationality : The ability to make optimal decisions based on available information (e.g. AI trading bots adjusting stock investments).
vi) Autonomy - The degree of independence an agent has (e.g voice assistants like Alexa operating without human input).
vii) Types of Agents :
a) Simple Reflex Agents - Act based on current perceptions (e.g thermostats).
b) Model-Based Agents - Use internal models to predict future states (e.g GPS navigation systems).
c) Goal-Based Agents - Take actions to achieve a specific goal (e.g. chess-playing AI)
d) Utility-Based Agents - Optimise outcomes for maximum benefit (e.g. recommendation systems in e-commerce).
e) Learning Agents - Improve performance over time using past experiences (e.g. self-learning robots in warehouses).

Q3) How is AI technique used to solve 8-puzzle problem?
Ans) The 8-puzzle problem involves arranging numbered tiles in a 3x3 grid using AI to find the shortest solution. The AI techniques used are:
1. Uninformed Search (Explores all moves blindly) :-
i) BFS: Finds the shortest path by exploring level-by-

level.

ii] DFS: Explores deeply but may not be optimal.

2. Informed search (Uses heuristics for efficiency):-

i] A* : Combines path cost and heuristics for optimal moves.

ii] Heuristics (Best-first): Calculates the estimated distance from the goal node based on two points of the original position.

3. Optimisation Methods:-

i] Genetic Algorithms: Evolves solutions over multiple iterations.

ii] Simulated Annealing: Avoids getting stuck in bad solutions.

Q4] What is PEAS descriptor? Give PEAS descriptor for following:

i] Taxi Driver

ii] Medical diagnosis system

iii] A music composer

iv] An aircraft autolander.

v] An essay evaluator.

vi] A robotic sentry gun for the Keck Lab.

Ans] PEAS (Performance Measure, Environment, Actuators, and Sensors) is a framework used to define the structure and functioning of an AI agent. It describes how an agent interacts with its enviroment to achieve its goal.

Teacher's Sign.:

AI System	Performance Measure (P)	Environment (E)	Actuators (A)	Sensors (S)
i) Taxi Driver AI	Safety, efficiency, fuel usage, passenger comfort	~~Rod~~ Roads, traffic, passengers, weather.	Steering wheel, brakes, signals, accelerator.	GPS, cameras, speedometer, proximity sensors, microphone.
ii) Medical Diagnosis System	Accuracy, speed, reliability, patient satisfaction	Patient symptoms, medical history, lab reports	Display screen, reports, prescriptions	Patient input, medical test results, wearable health monitors
iii) AI Music Composer	Melody quality, harmony, originality	Music theory rules, user preferences, compositions	Music generation software, speakers.	User feedback, genre data, music databases.
iv) Aircraft Autolander	Smooth landing, accuracy, passenger safety	Airplane altitude, weather, runway conditions	Flaps, landing gear, thrust control, brakes	Altimeter, GPS, radar, wind sensors, airspeed indicator.
v) AI Essay Evaluator	Grammar accuracy, coherence, structure, relevance.	Essays, grammar rules, grading rubrics.	Screen output, grading suggestions, feedback	Text input, grammar checking tools.
vi) Robotic Sentry Gun	Accuracy, response time, threat detection efficiency	Lab surroundings, potential threats, movement patterns	Gun turret, alert system, camera movement	Motion detectors, thermal sensors, facial recognition cameras.

Date
Page

Q5) Categorise a shopping bot for an offline bookstore according to each of the six dimensions (fully / partially observable, deterministic / stochastic, episodic / sequential, static / dynamic, discrete / continuous, single / multi agent).

Ans) A shopping bot for an offline bookstore can be categorised using the six AI environment dimensions as follows:

- Partially observable: The bot does not have complete visibility into real-time inventory, customer preferences, or book availability.
- Stochastic: Customer choices, stock updates, and staff interventions introduce uncertainty, making the environment unpredictable.
- Sequential: Each customer interaction depends on prior actions, such as recommendations based on past selections.
- Dynamic: The environment changes as books are purchased, restocked, or reordered, affecting the bot's decision making.
- Discrete: The bot operates through distinct actions like checking stock, suggesting books, and processing purchases.
- Multi-Agent: The bot interacts with multiple entities, including customers, bookstore staff, and inventory management systems.

Q6) Differentiate between Model-based and Utility-based agent.

Ans)

Model-based Agent	Utility-based agent
i) It uses an internal model of the environment to make decisions.	i) It chooses actions based on maximising a utility function.

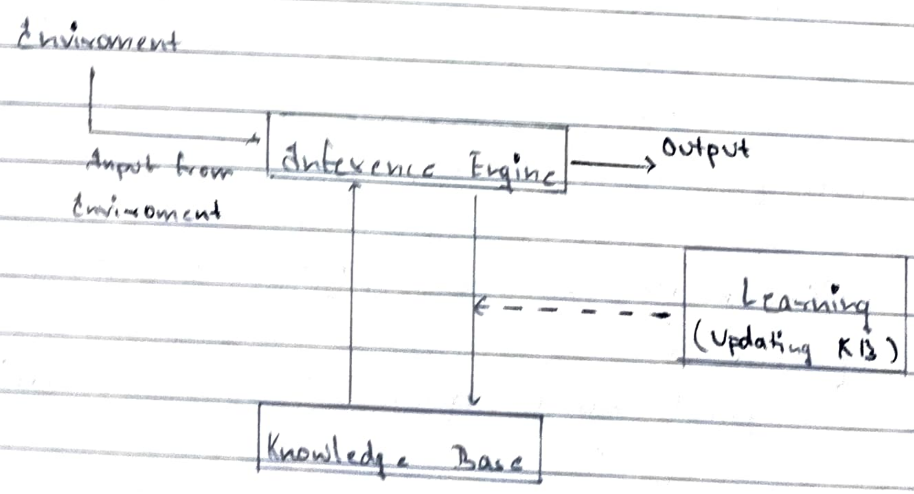
Teacher's Sign.:

ii] It predicts future states using the model before acting.	ii] It evaluates multiple possible actions and selects the best based on utility.
iii] It is goal-oriented but focuses on how the environment works.	iii] It is goal-oriented but aims to achieve the best possible outcome.
iv] It may not always take the best possible action, only a feasible one.	iv] It always selects the action that provides the highest benefit.
v] For example, a self-driving car using a traffic model to predict congestion.	v] For example, a stock trading AI selecting the trade with the highest expected profit.

Q7] Explain the architecture of a knowledge based agent and learning agent.

Ans]

- Architecture of a Knowledge-Based Agent :-

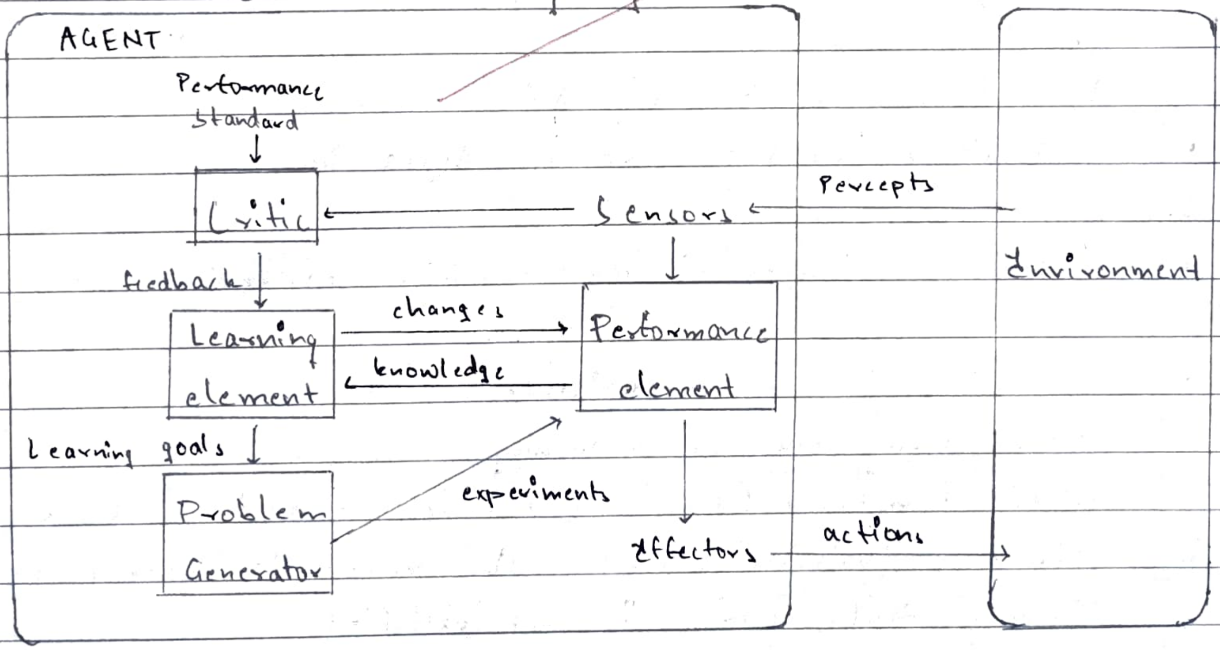


Date
Page

A Knowledge-Based Agent uses stored knowledge to make decisions and reason about the environment. Its architecture consists of:

i) Knowledge Base (KB): Stores facts, rules, and background knowledge in a structured format.
ii) Inference Engine: Applies logical reasoning to derive conclusions from the knowledge base.
iii) Perception Module: Gathers inputs from sensors or external sources.
iv) Action Module: Executes actions based on derived conclusions.
v) Learning Mechanism: Updates the knowledge base with new information.

• Architecture of a Learning Agent:-



A Learning Agent improves its performance over time by learning from past experiences. It has four main components:

i) Learning Element: Learns from interactions and updates

Teacher's Sign.:

knowledge.

ii) Performance Element: Uses the learned knowledge to make decisions.

iii) Critic: Evaluates the agent's performance by comparing actions with desired outcomes.

iv) Problem Generator: Suggests new exploratory actions 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 puncture so it is not available.

Will Anita travel via Goregaon? Use forward reasoning.

Ans) a.

- Available (Car) → Travels (Anita, Car)
- ¬ Available (Car) → Travels (Anita, Bus)

b.

- ~~Route (Bus, Andheri)~~ GoesVia (Bus, Andheri ∧ Goregaon)

c.

- Puncture (Car) → ¬ Available (Car)

Now, using forward reasoning:-

i) Given Punctured (Car), we infer: ¬ Available (Car)

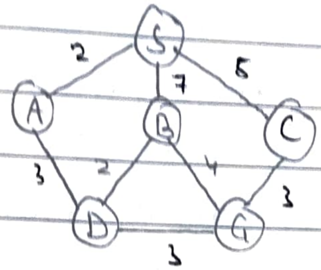
ii) Since ¬ Available (Car), the rule Available (Car) → Travels (Anita, Car) is not satisfied.

iii) Therefore, the alternative rule applies: ¬ Available (Car) → Travels (Anita, Bus)

iv) Since Goes Via (Bus, Goregaon) and Anita is travelling by bus, she will travel via Goregaon.

Teacher's Sign.:

Q10) Find the route from S to G using BFS.

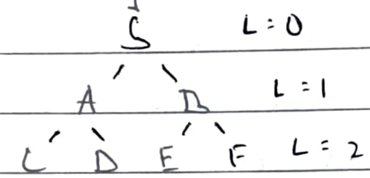


Ans) We know that, in BFS, queue is used.
So starting S, it gets popped from queue and its neighbours get added in order of priority :-
[(A, 2), (C, 5), (B, 7)]
Then, A gets ~~poppe~~ dequeued and its neighbours get added :-
[(D, 3), (C, 5), (B, 7)]
Then, D gets dequeued and its neighbours get added to the queue :-
[(B, 2), (G, 3), (C, 5), (B, 7)]
We see that, the target node G is found.
Thus, route S → A → D → G is correct (path length = 2 + 3 + 3 = 8)
Similarly,
S → C → G (path length = 8)
Thus, either of the two paths are optimal.

Q11) What do you mean by Depth Limited Search? Explain Iterative Deepening Search with example.
Ans) i) Depth-limited search is a variation of Depth-First Search where the search is restricted to a specific depth limit L. If a goal is not found within this limit, the search terminates. It helps in preventing the search from

being infinitely stuck in a branch. For ex: if L = 2, only nodes upto depth = 2 are checked through DFS.

iii) Iterative Deepening Search (IDS) combines memory efficiency of DFS and completeness of BFS by repeatedly running DLS with increasing depth limits until the goal is found. For ex (searching for E):-



→ At DLS with L = 0, checks S, goal isn't found.

→ At DLS with L = 1, check A, B, goal isn't found.

→ DLS with L = 2, checks C, D, E, F, goal is found.

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

Ans) Hill climbing is an iterative optimisation algorithm that starts with an initial solution and makes incremental changes (using greedy method) to improve it. The goal is to reach the best possible solution by always moving towards higher values in the search space.

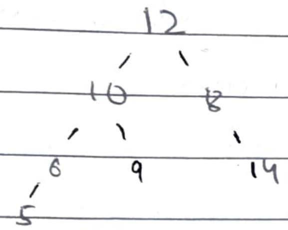
Drawbacks of Hill Climbing :-

i) Local Maxima - It may stop at a node which is the local maxima but not the global maxima.

ii) Plateau - Flat regions can prevent further movement in Hill Climbing algorithm.

iii) Ridges - It struggles if progress requires a temporary decline because it assumes it as a peak.

For ex:-



→ Starts at 5, neighbours are 6.
→ Moves to 6, neighbours are 10, 9.
→ Chooses 10 as it is higher than ~~10~~ 9.
→ Then chooses 12 as it is highest value and then stops.

Here, we see that 12 is local maxima and 14 is global maxima but search stops at 12. This is the major drawback.

Limitations of Steepest Ascent Hill Climbing:-

i) High computational cost as evaluating all neighbours is expensive.
ii) Still prone to getting stuck in local maxima.
iii) It can take longer than simpler versions.

Q12) Explain simulated annealing and write its algorithm.

Ans) Simulated annealing is a ~~probabistic~~ probabilistic optimisation algorithm inspired by the annealing process in metallurgy. It helps in finding a global optimum by allowing occasional moves to worse solutions to escape local optima. It uses Metropolis algorithm, which is based on Monte Carlo technique. Here, a solution for a problem is equated to states in a physical system and the cost of a solution is equated to the energy of a state.

Algorithm of simulated annealing:-

1. Initialise the current state and temperature T.

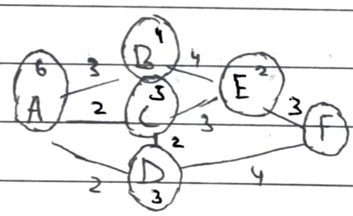
2. Repeat until stopping condition is reached:
   i] Select a random neighbour state.
   ii] Compute energy difference: $\Delta E = E_{new} - E_{current}$
   iii] If $\Delta E < 0$, accept the new state.
   iv] Else, accept it with probability $e^{-\Delta E/T}$ (Metropolis criterion)
   v] Reduce the temperature 'T' according to cooling schedule.
3. Return the best-found solution.

Q14] Explain A^* algorithm with an example.

Ans] The A^* algorithm is an informed search algorithm that finds the optimal path by considering both the cost to reach a node ($g(n)$) and the estimated cost to the goal ($h(n)$) using the formula:

$$f(n) = g(n) + h(n)$$

For ex:-



To go from A to F.

→ at A, there are 3 neighbours:-
i] B : $h(n) = 4$, $g(n) = 3$ → ∴ $f(n) = 4+3 = 7$
ii] C : $h(n) = 5$, $g(n) = 2$ → ∴ $f(n) = 5+2 = 7$
iii] D : $h(n) = 3$, $g(n) = 2$ → ∴ $f(n) = 3+2 = 5$

Thus, as $f(n)$ is minimum for D, we move to D.

→ at D, there are 2 neighbours:-
i] C : $h(n) = 5$, $g(n) = 2$ → ∴ $f(n) = 5+2 = 7$.
ii] F : $h(n) = 0$, $g(n) = 4$ → ∴ $f(n) = 0+4 = 4$

Now, F is goal node. Thus, the search is complete.

FOR EDUCATIONAL USE

Q15) Explain Minimax algorithm and draw game tree for Tic-Tac-Toe game.

Ans) The Minimax algorithm is a decision-making strategy used in two-player games like chess and tic-tac-toe. It helps find the best move by simulating all possible future moves and assuming that the opponent plays optimally.

It works ~~s~~in the following way :-

i) Maximising Player (MAX): Tries to maximise the score.

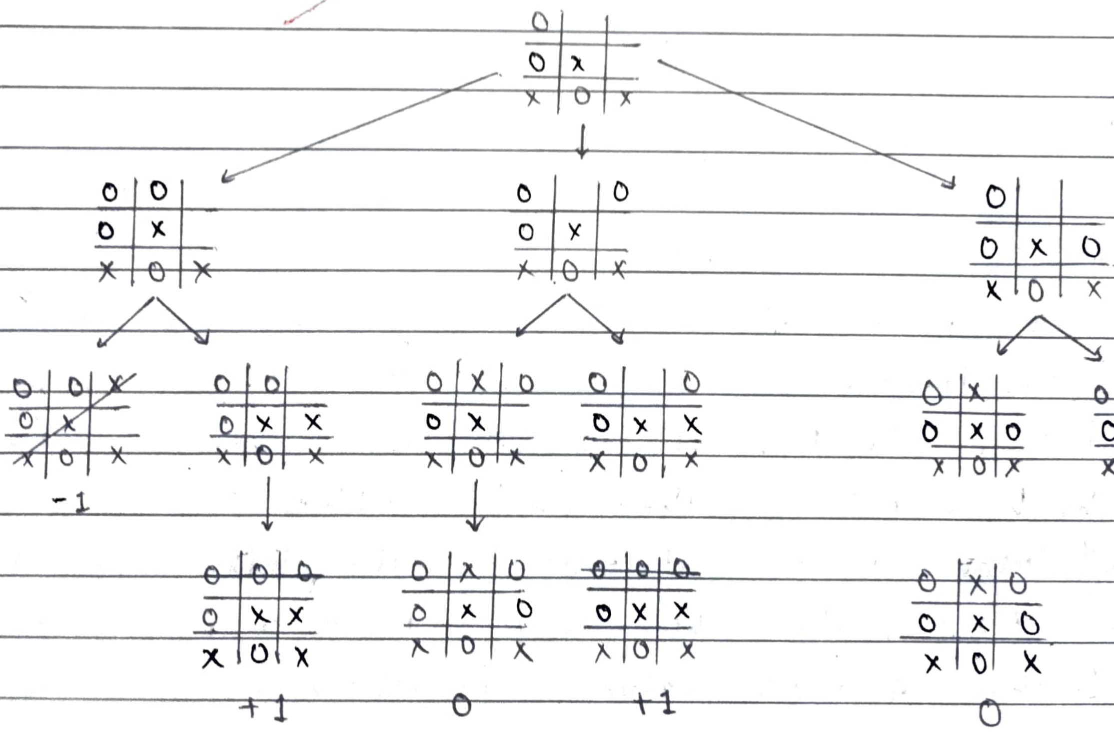
ii) Minimising Player (MIN): Tries to minimise the score.

iii) ~~Fo~~ It generates all possible game states up to a certain depth.

iv) It assigns scores to terminal states based on winning chances.

v) The best move is chosen assuming both players play optimally.

Game tree for Tic-Tac-Toe :-



Q16) Explain Alpha-Beta Pruning Algorithm for ~~adversial~~ adversarial search with an example.

Ans) Alpha-Beta Pruning is an optimisation technique for the Minimax algorithm that reduces the no. of nodes evaluated, making it more efficient. Instead of exploring all game states, the algorithm prunes branches that cannot influence the final decision. This is how it works:

i) Alpha(α) : Best value MAX can guarantee.

ii) Beta(β) : Best value MIN can guarantee.

iii) While traversing the game tree:

a) If a node's value is worse than the best alternative, stop exploring ($\alpha \geq \beta$).

b) This avoids evaluating unnecessary nodes.

For ex:-

```
        (ROOT - MAX)
         /    |    \
        A     B     C
       MIN   MIN   MIN
      / | \  / \    |
     3  5  2 6  9   7
```

i) At Root, $\alpha = -\infty$, $\beta = +\infty$.

ii) At A, it evaluates the child values (3, 5, 2) and selects the lowest value(2). Thus, $\beta = 2$ for this branch.

iii) Move to B. It evaluates the child values (6, 9). As both 6, 9 > 2, the entire subtree is pruned (as $\alpha \geq \beta$).

iv) Move to C. It evaluates the child value (7). As 7 > 2, the subtree is pruned (as $\alpha \geq \beta$).

v) Thus, A with value 2 is chosen.

Q17 Explain WUMPUS World environment giving its PEAS description. Explain how percept sequence is generated.

Ans The Wumpus World is a grid-based AI environment where an agent explores a cave while avoiding hazards like pits and the Wumpus monster.

PEAS Descriptor :-

- Performance Measure : +1000 points for reaching the gold and returning back safely, -1000 for dying, -1 for every move of the agent, -10 points when the agent uses the arrow.
- Environment : A 4x4 grid with pits, gold, and Wumpus.
- Actuators : Move forward, turn left, turn right, grab, shoot, climb.
- Sensors : Perceive stench (Wumpus in an adjacent room), breeze (Pit in an adjacent room), bump (if agent runs into a wall), a scream (if Wumpus is killed), glitter (gold is found).

Percept Sequence Generation :-

i) Agent starts at (1,1), sensing its surroundings.
ii) If stench is detected, Wumpus is in an adjacent room.
iii) If breeze is detected, a pit is in an adjacent room.
iv) The agent infers safe paths and navigates toward the gold while avoiding hazards.

Q18 Solve the following crypo-arithmetic problem: SEND + MORE = MONEY.

Ans

```
   SEND
 + MORE
  MONEY
```

→ As M is the carry, it's value can only be 1.

$\therefore$ M=1.

→ Now, S+M=O. As M=1, S+1=O. So, S is a number which creates a carry when added with 1. Thus, S=9, O=0.

→ Now, E+O=N. But, O=0. Thus, E=N. But, this is not possible. Thus, we infer that a carry is involved.

Thus, 1+E=N. — (i)

→ Next, N+R=E — (ii)

We solve (i), (ii), we get E=5, N=6.

→ Now, from above, N should be 9 but it is already assigned. Thus, R=8 and a 1 comes from a carry.

→ Thus, D+E=Y should generate a carry. ~~Thus~~ But, E=5.

$\therefore$ D+5=Y and a carry is generated.

→ Thus, D is greater than 4. But 5, 6, 8 and 9 are taken already. Thus, D=7 and Y=2.

Thus, solution:

M=1, S=9, O=0, E=5, N=6, R=8, D=7, Y=2.

In the given form:-

$$\begin{array}{r} 9\;5\;6\;7 \\ 1\;0\;8\;5 \\ \hline 1\;0\;6\;5\;2 \end{array}$$

Q19 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 first order predicate logic.

2) Convert each formula to clause form.

3. Prove that "Is someone smiling?" using resolution technique. Draw the resolution tree.

Ans) 1) FOL:-

i) $\forall x\,(Graduating(x) \rightarrow Happy(x))$

ii) $\forall x\,(Happy(x) \rightarrow Smiling(x))$

iii) $\exists x\,(Graduating(x))$

2) Converting above to clauses form:-

i) $-Graduating(x) \vee Happy(x)$

ii) $-Happy(x) \vee Smiling(x)$

iii) $Graduating(A)$

3. Resolution:-

→ We have Graduating (A). From Graduating (A) and $-Graduating(x) \vee Happy(x)$, we infer:-

$((Graduating(A)), (-Graduating(x) \vee Happy(x))) \Rightarrow Happy(\overset{A}{\cancel{x}})$

This is because an empty clause is created for Graduating (A).

→ We have Happy (A) and $-Happy(x) \vee Smiling(x)$. From this, we infer:-

$((Happy(A)), (-Happy(x) \vee Smiling(x))) \Rightarrow Smiling(A)$.

Thus, someone is smiling.

Resolution Tree:-

Graduating (A) $-Graduating(x) \vee Happy(x)$

Happy (A) $-Happy(x) \vee Smiling(x)$

Smiling (A)

Q.20) Explain Modus ponen with suitable example.

Ans) Modus Ponens is a fundamental rule in propositional logic that states:

$P \to Q$ (if P then Q).

and P is true, then Q must be true.

For ex:-

→ Premise 1: If it rains, the ground will be wet ($P \to Q$)

→ Premise 2: It is raining (P is true)

→ Conclusion: The ground is wet (Q is true).

Q.21) Explain Forward Chaining and Backward Chaining algorithm with an example.

Ans) • Forward ~~Chectin~~ Chaining: It is a method of inference that starts with known facts and applies rules to derive new conclusions. The process continues until the goal is reached or no new facts can be inferred.

For ex: Diagnosing a disease:-

1. Given facts:-

i] Fact 1: If a person has a fever and cough, they may have the flu.

ii] Fact 2: John has a fever.

iii] Fact 3: John has a cough.

2. Rules:

i] Rule 1: If a person has a fever and cough, they have the flu.

ii] Rule 2: If a person has the flu, they should rest and drink fluids.

3. Inference using forward reasoning:-

i] Step 1: John has a fever and cough.

ii] Step 2: Apply Rule 1 → John has the flu.
iii] Step 3: Apply Rule 2 → John should rest and drink fluids.

Thus, conclusion: John should rest and drink fluids.

- Backward Chaining: It is an inference method that starts with the goal and works backward to check if the given facts support it.

For ex: Same example as Forward Chaining:

1. Facts and Rules same as that of Forward Chaining.
2. Goal: Prove that John has the flu.
3. Backward Chaining steps:-

i] To prove John has the flu, check if he has a fever and a cough (from Rule 1)
ii] Check if John has a fever (Fact 1: Yes).
iii] Check if John has a cough (Fact 2: Yes).
iv] Since both conditions are met, we can conclude that John has the flu.

Thus, conclusion: The hypothesis is proven (John has the flu).