Name :- Atharva Shinde
Roll no :- 54
class :- D15C

04/03

AIDS - Assignment - 1

Page 1

what is AI ? consedering the covID-19 pandemic situation how AI helped to survive and renovated our way of life with different applications?

Artificial Intelligence (AI) enables machines to think, learn with and make decisions like humans. It includes technologies like machine learning, NLP & robotics.

Applications :

1) Healthcare : AI helped in early diagosis, vaccine development, and chatbot-based health assitance

2) contact tracing : AI-powered apps tracked covID-19 exposure ensuring public safety.

3) Remote work & Education : AI enhanced virtual meetings online learning & productivity tools

4) Supply Chain & Delivery : AI optimized logistics & enabled autonomous deliveries

5) Mental Health support : AI-driven apps provided emotional & fitness assistance.

what are AI agents terminology, explain with examples

1) Agent : An entity that interacts with the environment & makes decision based on inputs

ex: A self-driving car perceives traffic signals & adjust speed accordingly.

2) Performance measures : Defines how sucessful an agent is in achieving its goal

Teacher's Sign.: _____

ex: self-driving cars performance measures could be minimizing accidents, fuel efficiency & travel time

b) Behavior / Action of Agent : The action an agent takes based on its percepts.
ex: A robotic vacuum cleaner moves around after detecting them

4) Percept : The data an agent receives at a specific moment from sensors
ex: A spam filter receives an email & detects keywords, senders info and attachments.

5) Percept sequence : The entire history of percepts received by an agent.
ex: A chess playing AI remembers all previous moves in the game before making its next move.

6) Agent function : A mapping from the percept sequence to do an action.
ex: A smart thermostat analyzes past temperature changes & adjusts heating accordingly.

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

→ It consists of a 3×3 grid with 8 numbered tiles & one empty space, where the objective is to move the tiles around to match a predefined goal configuration.

Initial state

1	2	3
4		6
7	5	8

Teacher's Sign.:

This is the random starting configuration of the 8 puzzle with the tiles placed in a non-goal configuration

2) Goal state : The goal is to arrange the tiles in a specific order with the blank space at the bottom right

Goal state

1	2	3
4	5	6
7	8	

* solving the 8 puzzle problem

• AI search algorithms, such as breadth first search (BFS) · depth - first search (DFS) and A*, are commonly used.

1) Breadth first search (BFS):

• BFS is an uninformed search algorithm that explores all possible state level by level, starting from the initial state.

• BFS guarantees that the solution found is the shortest in terms of number of moves, but it can be very slow.

Advantages

Guarantees to find the optimal solution

Disadvantages

BFS has a high memory requirement, as it must store all the states at each level of exploration.

2) Depth first search (DFS)

- DFS is another uniformed search algorithm that explores one branch of the state space as deep as possible before back tracking

Advantages

DFS is more memory efficient than BFS.

Disadvantages

DFS can get stuck in deep, non-optimal paths & may not find the shortest solution.

steps using A*

- compute Manhattan distance for each move
- choose the best move (lowest f(n))
- Repeat until reaching the goal state.

Q4) what is PEAS descriptor? Give PEAS descriptor for following

1) Taxi driver:-

- P: Minimize travel time, fuel efficiency, passenger safety, obey traffic rules.
- E: Roads, traffic, Passengers, weather, obstacles Pedestrains.
- A: steering, accelerator, brakes, turn, signals horn
- S: camera, GPS, speedometer, radar, LiDAR microphone.

2) Medical Diagonsis system.

- P: Accuracy of dignosis, treatment sucess rate, response time.

- E : Patient records, symptons, medical tests, hospital data.
- A : Display screen, printed prescription, notifications.
- S : Patient input, lab reports, electronic health records.

3) A music composer:

- P : Quality of music, adherence to genre, audience engageme
- E : Digital workspace, music production software, real time composition setting.
- A : Audio output, digital instrument selection, file saving export
- S : user inputs, style preferences, tempo · feedback from listners, musk theory constraints.

4) An aircraft Autolander:

P : smooth landing, accuracy in reaching runway, passengers safety, fuel efficiency.
E : Airspace, runway, weather, wind speed, visibility
A : Flight control, landing gears, brakes.
S : GPS, airspeed indicator, gyroscope, radar, climate.

5) An essay Evaluator

P : Accuracy of grading, consistency, fairness, grammer.
E : Digital text input, student essays, predefined grading
A : Feedback generation, score assignment, highlighting errors, suggesting improvement.
S : optical character recognition, NLP grammer & spell checkers.

6) Robotic sentry gun for the keck lab
P: Target accuracy threat detection efficiency, respon
speed.
E: keck lab premises, intruders, lighting conditions, obs
A: Gun aiming system, firing mechanism, camera
Panning alert system.
S: Motion detectors, infrared sensors, cameras, LI
radar.

categorize a shopping bot for an offline bookstore
according to each of six dimension (fully / partially
deterministic / stochastic / episodic / sequential, static /
dynamic, discrete / continous, single, multi agent).
1) partially observable: The bot may not have comp
visibility.
2) stochastic: The environment is unpredictable.
3) sequenstial: Each decision bot makes affects future
states.
4) Dynamic: The bookstore environment changes overtime
5) Discrete: Bot choose discrete choices (selecting books
6) multi agent: The bot interacts with multiple entities

06) Differentiate Model based & utility based agent

Model Based Agent	utility Based Agent
1) Maintains an internal model of the environment to make decision	1) uses a utility function to measure performance & make option choices.
2) Relies on stored knowledge & updates the model.	2) chooses actions based on maximizing expected utility.
3) can adapt to changing environment by updating the internal model.	3) More flexible & goal-oriented adapting to changes dynamically.
4) Moderate complexity due to model maintaince	4) Higher complexity due to the need to compute utilities for different actions.
5) Ex :- self-driving car that predicts pedestrain movement	5) ex :- A self driving car that dissects & selects the best path.

07) Explain the architecture of a knowledge based agent & learning Agent :-

→ Knowledge Based Agent Architecture.

- A knowledge-based Agent is an intelligent that makes decisions using knowledge base (KB) and reasoning machinism.

Architecture component.

1) knowledge base stores facts rules & heuristics about the world
2) Inference Engine : use logical reasoning (FOL) to derive new knowledge from the KB.

3) Perception module collects data from sensors & update the KB.
4) Action selection module chooses appropriate actions based on reasoning outcomes.
5) communication module Allows interaction with other agents.

working process:
- The agent perceives the environment & updates its KB
- The interference engine applies logical rules to infer new knowledge.
- The agent decides an action and executes it.
- The KB is continously updated to improve decision making

2) Learning Agent Architecture
- A Learning agent improves its performance over time by learning from past experiences & interactions with the environments.

Architecture components
1) Learning Element Analyses feedback from the enviroment and improves knowledge.
2) Performance Element make decisions & executes actions
3) Critic: Evaluates the agents action & provides feedback
4) Problem Generator suggests explotory actions to improve learning.

working process

- The performances element selects an action.
- The critic evaluates the action & provides feedback.
- The learning element updates the agents knowledge to improve future decisions
- The problem generator suggests new strategies to explores better solutions

eg) convert the following to predicates:

1) Anita travels by car if available otherwise travel by bus

→ - Available (car) ⇒ Travels (Anita, car)
- ¬ Available (car) → Travels (Anita, car)

2) Bus goes via Andheri & goregaon

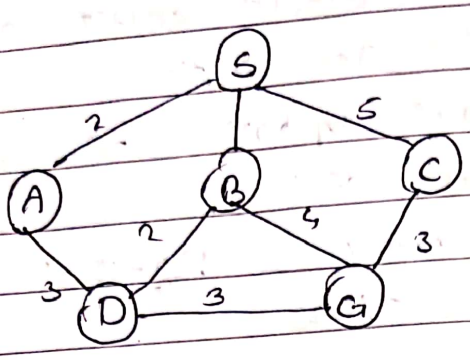
→ - Route (Bus, Andheri).
- Route (Bus, Goregaon).

3) car has a puncture, so it is not available.

→ - Punctyre (car) → ¬ Available (car)
- Given : Punctyre (car)
- Therefore ¬ Available (car)

1) From 3, ¬ Available (car) Ineg Available (car) ¬ Available (car)
2) From 1, ¬ Available (car) → Travels (Anita, Bus) Ineg Available (car) → Travels (Anita, Bus) ¬ Available (car)
→ Travels (Anita, Bus), so Anita must travel by bus.
3) From 2: Route (Bus, Goregaon) Route (Bus, Goregaon) Route (Bus, Goregaon), meaning the bus travels via goregaon.

Q10) find the route from S to G using BFS



current Node	queue	visited Node
S	A B C	S
A	B C D G	S → A
B	C D G	S → A → B
~~C~~	D G	S → A → B → ~~C~~
D	G	S → A → B → C → D
G		S → A → B → C → D → G

paths S → A → B → C → D → G

Q11 what do you mean by depth limited search? Explain iteractive Deepening search with example.

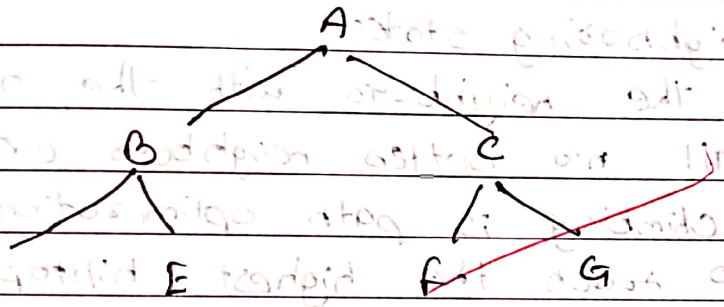
→

DLS is an uninformed search algorithm the that modify DFS by using depth limit L preventing exploration beyond defined level.

This prevents infinite loops in graphs, but risks missing goals beyond L.

Interation deeping search combines DLS with BFS by incrementally increasing the depth limit

example



Goal = G

Initial D.L = 0

limit	node visited	Goal
0	A	Not found
1	A, B, C	Not found
2	A, B, C, D, E, F, G	found G.

Teacher's Sign.: _____

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

→ Hill climbing is a local search algorithm used to find an optimal solution by iteractively making small changes to the current state and choosing the best improvement.

Algorithm steps

1. Start with an initial solution (state)
2. Evaluate neighboring states.
3. Move to the neighbour with the highest value
4. Repeat until no better neighbour exists.

example: Hill climbing is path optimization): A robot trying to reach the highest hilltop (goal) uses hill climbing It moves upwards step by step choosing the strongest ascent until no higher step is available.

Drawbacks of hill climbing

1. local Maxima - may get stuck at a peak that is not the global maximum.
2. Plateau - A flat region with no improvement, leading to stagnation.
3. Ridges - A narrow path of improvements that the algorithm may fail to follow.
4. No Backtracking - once a move is made, previous states are not reconsidered.

Teacher's Sign.:

Q13) Explain simulated annealing and write its algorithms

A probalilistic search method inspried by the cooling process of metas.

Allow occasional downfall moves or escape local option.

i) start with an initial solution temperature T.

ii) Generate a neighbouring solution

iii) If its better, accept it, otherwise accept it with probability $e^{(-\Delta E/T)}$

iv) Decrease T gradually and repeat until T is very small.

use case: Travelling salesman problem (optimized problems)

Q14) Explain A* Algorithm with an example.

→ A* is like best first search algorithm used in path finding and graph traversed. It used the following formulas:-

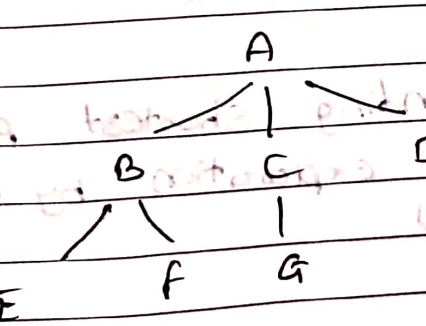
$f(n) = g(n) + h(n)$

$g(n)$ → use to reach n from start

$h(n)$ → heuristic estimate of cost to reach from goal to n

$f(n)$ → total estimated cost

e.g Goal: G



Node	g(A, n)	h(n, G)
A	0	6
B	1	4
C	2	2
D	4	7
E	3	5
F	5	3
G	6	0

steps

1. start at root node A

$f(A) = g(A) + h(A) = 0+6$

$= 6$

2. expand neighbours B, C, D

$f(B) = 1+4 = 5$

$f(C) = 2+2 = 4$

$f(D) = 4+7 = 11$

3. choose lowest value that is $f(C)$

4. Expand neighbours of C, $f(G)$

$f(G) = 6+0 = 6$

5. Goal reached at G with total cost 6

Advantages →

efficient for finding shortest path in weighted graphs balances exploration by considering both g(n) and h(n)

15] Explain min-max algorithm and draw game tree for tic tac toe game.

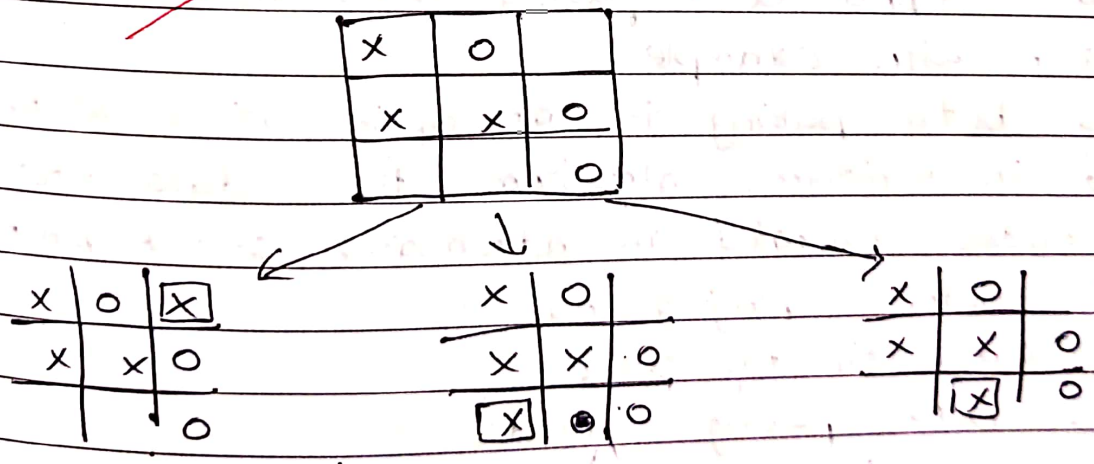
→ The min-max algorithm is a decision making algorithm used in 2 players games it ensures one player (Max) tries to maximize its score & other player (min) tries to minimize the score.

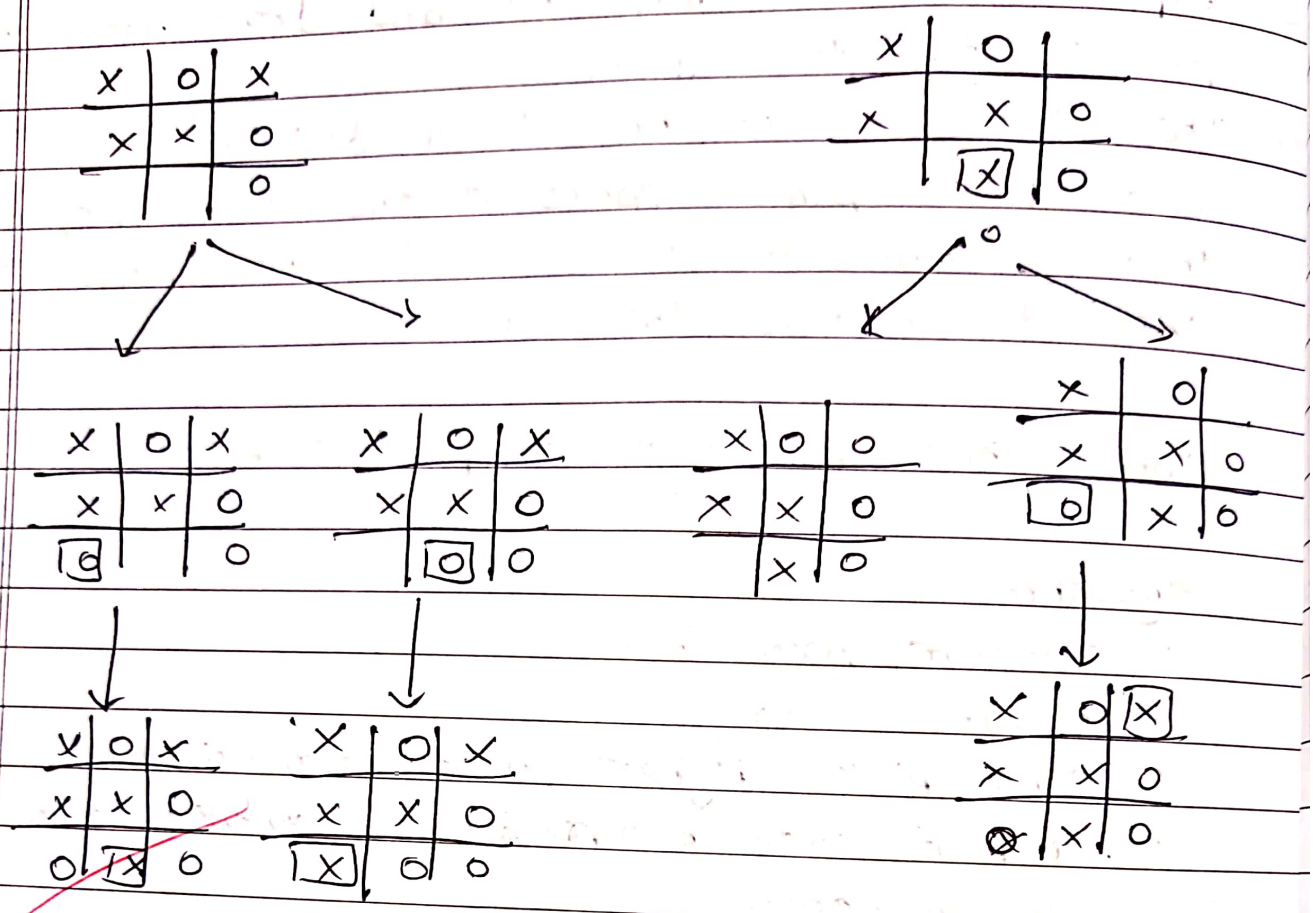
Game tree represents all possible moves

Algorithm

1. Generate game tree
2. Assign scores
3. Max picks highest value from children
4. Repeat until root node is evaluated a bottom up approach.

Game tree for the tic-tac toe game



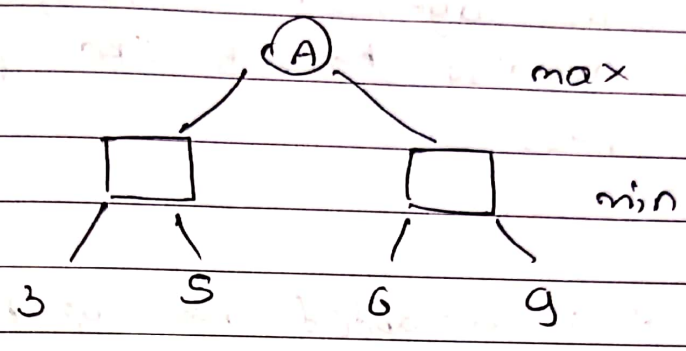


Q16) Explain alpha beta pruning algorithm for adversial search with example

→ Alpha beta pruning is an optimization technique used in minimum algorithm to reduce the number of nodes evaluated in adversarial search problems like game-playing AI
(eg class, tic-tac toe)
Alpha beta pruning includes
Alpha (α). The best maximum score that the maximising player can guarantee so far.

Beta (β): The best minimum score that the minimizing player can guarantee so far
The algorithm prunes branches that will not influence final decision.

e.g:-



1. Start at root node A
   $\alpha = -\infty$, $\beta = \infty$
2. • Check left min node (child of A)
   • Check first child value = 3 → update β = 3
   • Check second child value = 5 → β remains 3
   • Min node returns 3 to Max.
3. Right Min node (child of A)
   • Check first child value = 6 → β = 6.
   • Here α = 3 at MAX node but β (6) > α (3). so no pruning
   • Explores 2nd child (9) → Here pruning will occur
   • MIN node already has a value ≤ 6 it will never choose a 9 so use prune the node with value 9.
4. Max value = 6

Q17) Explain Wumpus world environment giving its PEAS description. Explain how percept sequence is generated?

→ The wumpus world environment is a simple grid environment used in AI to study intelligent agent behaviour.
In uncertain environments it is a turn based e ent where an agent must navigate a cave to f gold while avoiding hazards like pits and a mon called wumpus.

PEAS :-
P : The agent is rewarded for grabbing gold and exiting safely - Penalty is imposed for falling into pits and getting eaten by wumpus.
E : 4×4 grid world containing the agent, wumpus, pits
A : The agent can move forward, left, Right, shoot,
S : Agent perceives stench, breeze, glitter, bump a scream.

Percept sequence generation.
It is the history of all perceptions recieved by the agent. At each time step, the agent. At each time step the agent perceives information based on its current location and surroundings.

Example percepts sequence :
1. Agent starts at (1,1):
- No breeze, no stench, no glitter → safe square.
2. Agent moves to (2,1):
- Breeze detected → A pit is nearby but not in current square.

3. Agent moves to (1,2)
   stench detected → wmpus is in adjacent cell
4. Agent moves to (2,2)
   glitter detected → gold is here
5. Agent moves back to (1,1) and climbs out.

8) solve the cryptArithmetic SEND + MORE = MONEY

→ step 1 M must be 1
∴ sum of 2 '4' digit numbers comes to be greater than equal to 2000

```
  S E N D          S E N D
+ M O R E        + 1 O R E
-----------      -----------
M O N E Y        1 O N E Y
```

step 2 Assume E + O = 10 + N
then E+O generates carry & ∴ S has to be 8
otherwise if S is 9 & carry is 1 and 9+1 = 10
but O cannot be 1 ∴ M = 1

```
  S E N D          8 E N D          S = 9
+ M O R E        + 1 O R E          M = 1
-----------      -----------        O = 0
M O N E Y        1 O N E Y
```

step 3 but then E+O cannot generate carry. so initial assumption of carry is wrong and so S is 9

```
  9 E N D
+ 1 O R E
-----------
1 O N E Y
```

Step 4 Now E + 0 = N

$\Rightarrow$ E = N

but then is a carry so N+R must generate carry ! $\Rightarrow$ E + 1 + 0 = N

N = 1 + E

N + R + K1 = 10 + E

R + K1 = 10 + 1 = 9

K cannot be 0 otherwise R = 9

R + 1 = 9

R = 8

SEND $\longrightarrow$ 9 E N D

MORE 1 D 8 E

MONEY 10 N E Y

D + E = 10 + Y

also N + 8 + 1 = 10 + E

but N = E + 1 but assume E = 5 then N = E

9 5 6 D

1 0 8 5

1 0 6 5 7

So only requirement is D should be a number such that when added to 5 generate a query available space d 0, 1, 2, 3, 4, 5, 6, 7, 8, 9

2, 3, 4 cannot be as cannot generate carry

D = 7

9 5 6 7

1 0 8 5

1 0 6 5 2

its the solution !

consider the following axions

All people who are graduating are happy.

① Represent it predicate logic

G(x) ≙ x is graduating

H(x) is a happy.

1. collect clauses

(1) {¬ G(x), H(x)}

(2) {¬ h(x), S(x)}

(3) {G(x)}

2. Apply resolution

Resolve (1) {¬G(x), H(x)} with (3) & G(a)

substituting x=a

{¬ G(a), H(a)}

∴ we have h(a)' resolving gives

{H(a)}

∴ Resolves (2) {¬ H(a), S(a)}

since we derived S(a), we conclude that someone (a) is smiling

③ Prove 'is someone graduating' using resolution

1) collect clauses
   (1) { ¬ G(x) , H(x) }
   (2) { ¬ H(a) , S(a) }
   (3) { G(a) }
   (4) Assume contradiction (¬ ¬ S(a))

2) Apply resolution
   Resolve (1) { ¬ G(x), H(x) } with (3) { G(a) }
   substitute x = a ⇒ { G(a) }
   { ¬ G(a) , H(a) }
   we have G(a), resolving gives { H(a) }

   Resolve (2) { ¬ H(a), S(a) } with { H(a) }
   substitute x = a
   { ¬ H(a) , S(a) } & { H(a) }

   Now left with { S(a) }
   we can resolve it with { ¬ S(a) }
   ∴ we are left with
∴ our assumption of contradiction is wrong

Q20) Explain Modus ponen with suitable example.
→ Modus ponen is a fundamental rule of inference is propositional logic that allows us to deduce a conclusion from a conditional statement and its antecedent

It follows the form

1) $P \rightarrow Q$ (if P then Q)

2) P (P is true)

$\therefore$ Q (Q must be true)

Example:

1) If it rains, the ground will be wet : $P \rightarrow Q$

2) If is raining P

$\therefore$ Ground is wet → SS

11) Explain forward and backward chaing algorithm

→ forward chaining :- It starts with given facts and applies inference chaining - It rules to derive new facts until the goal is reached It is a data driven approach because it begins with known data and works forward to reach a conclusion.

Example Digonosing a disease

Rules

1. If a person has a fever and cough they might have flu.
2. If a person has a throat and fever, they might have cold.

facts:

- The patient has a fever.
- The patient has cough.

Inference : 1) fever + cough → flu (rule 1 applies)

2) conclusion the patient might have flu.

Backward chaining: It starts with goal and works backward by checking what facts are needed to support it. It is a goal driven approach.

Example : Diagnosing a disease

Goal : Determine if paitent has flue

Rules

1. (feaver ∧ cough) → flue
2. (sore Throat ∧ fever) → cold

process using backward Chaining

1. we want to prove flue
2. looking at rule 1 (fever ∧ cough) → flue we need to check if patient has fever and cough
3. we check our known facts
   - Patient has fever.
   - patient has cough.
4. Since both confits are met, we confirm flue is true.