Ayush Subhashchandra Maurya
D15C

AIDS (34)

05/05

34

Assignment 1

What is AI? Considering the covid-19 pandemic situation, how AI helped to survive & renovated our way of life with different application?

Ans: ① Artificial Intelligence is made up of 2 words.

Artificial → refers to something which is made by human.
Intelligence → refer to ability to acquire & apply knowledge & skills.

② AI can be defined as the ability of computer system i.e hardware & software, to do tasks that normally required human beings to use intelligence.

③ Term can also be applied to any machine that exhibits traits associated with a human mind such as learning & problem-solving.

AI applications During Covid-19 Pandemic

1] Contact Tracing & Tracking: AI driven contact tracing apps helped identify individual who came into contact with infected persons, enabling authorities to track & contain outbreaks more effectively.
2] Vaccine Development & Distribution: AI accelerate vaccine development by analyzing vast amounts of data, identifying potential vaccine candidates & optimizing distribution logistics.
3] Education & Learning: AI powered adaptive learning platforms helped students continue their education remotely, providing personalized learning experiences & automating grading process.
4] Public Health Surveillance: AI driven systems monitored social media, news, & other data sources to track public health trends, detect early warnings of outbreaks, & provide insights for policy makers.

What are AI Agents terminology, explain with examples.

Ans:

1] Agent: An entity that perceives its environment through sensors & acts upon it through actuators to achive specific goals.

Eg:- A robot vaccume cleaner perceives dust & obstacles through sensor & navigates around cleaning floors.

2] Environment: The external surroundings in which an agent operates.

Eg:- For a self-driving car, the environment includes roads, traffic signals, other vehicles.

3] Percept: Information received by agent from environment through sensor.

Eg:- AI receives current board state in games like chess

4] Percept sequence: The complete history of percepts an agent has received. Eg:- The history of all board positions a chess AI has observed.

5] Agent function: A mapping from percept sequences to actions.

Eg:- A function that maps a sequence of observed chess positions to the next best moves.

6] Performance measure: A criteria used to evaluate the success of agent's behaviour.

Eg:- Recomendation system, the percentage of suggested items that users actually purchased.

7] Rationality: The ability of an agent to select actions that maximizes its performance based on percept sequence & in build knowledge.

Eg:- Navigation system that chooses the shortest route.

8] Autonomy: Ability of agent to work without human intervention.

eg:- A mars rover performing navigating terrain.

3] Check if John has cough (yes)

4] If

Conclusion: John has flu.

9] Knowledge Base :- A collection of rules, fact & information that agent use while reasoning.
Eg :- A medical diagnosis system's database of symptoms, disease & treatment protocols.

10] Actuators :- A component that allows agent to act on environment.
Eg :- Motors in cars, drone,

Q3] How AI technique is used to solve 8 puzzle?
Ans :- The 8 puzzle is a sliding puzzle that consists of eight numbered tiles (1-8) placed randomly on 3x3 grid along with one empty slot. The player can move adjacent tiles into the blank space & the objective is to arrange the tiles in a specific goal state by sliding them on at a time.

Initial State :- This is the random starting configuration of the 8 puzzle, with the tiles placed in a non-goal configuration.
eg

1	2	3
4		6
7	5	8

Goal State :- In the 8 puzzle, only tiles adjacent to the blank space can be moved.
The following moves are allowed.
> Move the blank space up.
> Move the blank space down.
> Move the blank space left.
> Move the blank space right.

1	2	3
4	5	6
7	8	

Solving the 8 puzzle requires systematically searching through possible states (configurations) to find sequence of moves that lead to the goal state. AI search algorithms, such as Breadth first search, Depth first search & A^* are commonly used.

Heuristic plays crucial role in improving efficiency of solving the 8 puzzle using informed search algorithm like A^*. A heuristic is a function that estimates the cost from current state to the goal state, helping the search algorithm.

Two types

1] Misplaced Tile Heuristic → Count No. of misplace tiles.

2] Manhattan Distance Heuristic → Calculate sum of the grid distance between tile's current position & its goal position.

Q4] What is PEAS discriptor? Gives PEAS descriptor for following.

Ans: The PEAS (Performance measure, Environment, Actuator, Sensors) descriptor is a framework used to describe and analyze intelligent agents. It provides a structured way to identify the key components of an agent & its interactions with the environment.

	Performance measure	Environment	Actuator	Sensors
Taxi Driver	Safe & efficient transportation of passengers	Roads, traffic, weather, passengers	Steering wheel, accelator, brake	Eyes, ears, GPS, traffic sensors.

Q7 (b)

Condition: John has flu.

	Performance Measure	Environment	Actuator	Sensors
Medical Diagnose System.	Accurate diagnosis & effective treatment recomendation	Patient data, medical knowledge, diagnosis environment.	Display, screen pointer, Communication interference	Patient input device, medical imaging equipment.
Music Composer.	Creation of aesthetically pleasing & original music	Music theory, composition software, audio equipment.	~~Flight control surfaces, engine landing gear~~ MIDI controller audio output.	~~Radar, GPS, air data~~ keyboard, Mouse.
Aircraft Autolander	Safe & controlled landing of the aircraft.	Aircraft systems, weather, air traffic control	Flight control surfaces, engine landing gear.	Radar, GPS air data Sens.
Essay Evaluator	Accurate assessment of essay quality & content.	Essay text, grading criteria.	Display screen, pointer, communication interface.	Text input device, natural language
Robotic Sentry Gun for the Keck Lab	Effective detection & deterrence of intruders.	Keck lab premises, intruder behaviour -or.	Gun turret ~~Mount~~ Movement Systems, alarm system	Camera, motion, detectors, infrared sensors.

8] Differentiate Model based & Utility based agent.

Ans:

Model-based Agent	Utility-based Agent
1] Uses a model of the environment	1] Uses a utility function.
2] Can plan & reason about future actions.	2] Focus on maximizing expected utility.
3] Relies on internal model	3] Uses probabilistic utility functions.
4] Maintains internal model of environment	4] Interacts with environment through utility function.
5] Can handle complex goals	5] Maximize expected utility based on goals.

Q5] Categorize a shopping bot for offline bookstore according to each of the six dimensions.

Ans: 1] Fully / Partially Observable.

→ Partially Observable: The shopping bot may not have complete knowledge of the bookstore's inventory, customer preferences or sale data. It may rely on partial information such as customer inputs, inventory levels, or sales trends.

2] Deterministic / Stochastic

→ Stochastic: The shopping bot operates in an uncertain environment where customer preference, inventory levels, and sale data are subject to change. The bot must adapt to these uncertainties & make decision based on probabilities.

3] Episodic / Sequential

→ Sequential: The shopping bot interacts with customers in a sequential manner, where each interaction builds upon the previous one. The bot must maintain context & adapt its recommendations based on the customer's previous interactions.

4] Static / Dynamic

→ Dynamic: The shopping bot operates in a dynamic environment where inventory levels, customer preferences, and sales data changes over time. The bot must adapt to these changes and update its recommendations accordingly.

5] Discrete / Continous

→ Discrete : The shopping bot deals with discreate entities, such as books, customers, and inventory levels. The bots recommendations are based on discrete ~~data~~ data, such as book titles, authors & genres.

6] Single / Multi agent

→ Single agent : The shopping bot operates as a single agent interacting with customers & accessing inventory data without collaborating with other agents. However it may interact with other system, such as inventory management or customer relationship management system.

37] Explain the architecture of a knowledge base agent & learning Agent.

Ans : Knowledge based Agent Architecture.

A knowledge based Agent (KBA) is type of intelligent agent the uses knowledge to make decision and take action.

Component : 1] Knowledge Base : A repository of knowledge that the agent uses to make decision.

2] Interference Engine : A component that applies logical rules to the knowledge base to derive conclusions

3] Working Memory : A temporary storage area for information the agent is currently processing

Process:

1] Preception: The agent preceives its environment through sensors or input devices.
2] Knowledge Retrivel: The agent retrieves relevant knowledge from its knowledge.
3] Interface: The agent uses its inference engine to apply logical rules to the knowledge & derive conclusions.
4] Action: The agent takes action based on the conclusions derived.

Learning Agent Architecture.

Learning Agent is a type of intelligent agent that can learn from its environment & improve its performance over time.

Component:
1] Learning Component: A component the enable the agent to learn from its environment.
2] Performance Element: A component that selects actions based on the agent's current knowledge & goal.
3] Critic: A component that evaluate the agent's performance & provides feedback to the learning component.
4] Problem Generator: A component that generates new problem or situation for the agent to learn from.

Process:
1] Pre Perception: The agent preceives its environment through sensor or input device.

2] Action: The agent selects an action based on its environment, which is evaluated by the critic.
3] Feedback: The agent receive feedback from its environment, which is evaluated by the critic.
4] Learning: The agent learns from the feedback & updates its knowledge & behaviour.

Q 9] Convert the following to predicates:

(a) Anita travels by Car if available otherwise travel by bus.

⇒ $\forall(x)$ (Car Available(x) → Travels By(Anita, x)) ∧
$\forall(x)$ (¬ Available(car) → Travels By(Anita, Bus))

(b) Bus goes via Andheri & Goregaon.

⇒ Goes Via(Bus, Andheri) ∧ Goes Via(Bus, Goregaon)

(c) Car has puncture, so it is not available.

⇒ Puncture(car) → ¬ Available(car)

Will Anita travel via Goregaon? Use forward reasoning.

Forward Reasoning

1] From (3), we know: Car has a puncture so it is not available.
¬ Available(car)

2] From (1), Since car is not available, Anita will travels By bus
TravelBy(Anita, Bus)

3] From (2), Bus goes via Goregaon
Goes Via(Bus, Goregaon)

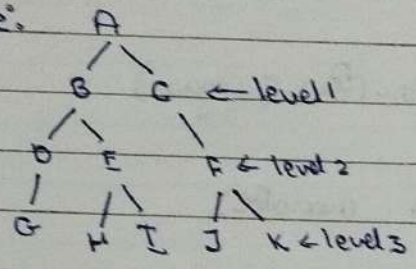
4] Since Anita travels by Bus, & Bus goes via Goregaon, Anita will travel via Goregaon.

Conclusion : Yes, Anita will travel via Goregaon.

11] What do you mean by depth limited search? Explain Iterative Deepning search with example.

→ Depth limit search (DLS) is a modified version of Depth-first search (DFS) that limits the depth of exploration to a fixed level L. If a solution is not found within the depth limit, it stops searching further.

Example:

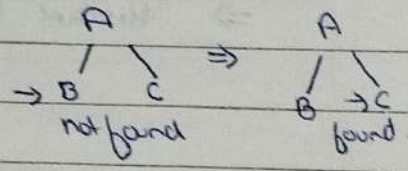


find G

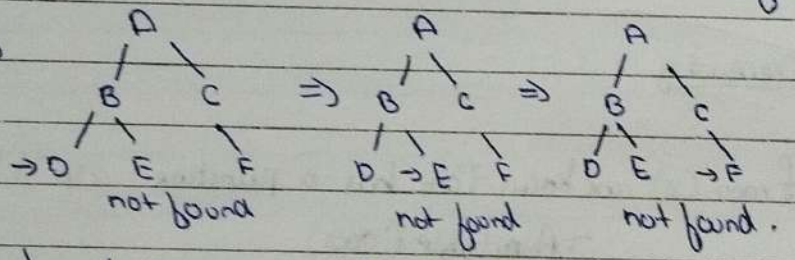
Depth = 2

Depth 0 → A

Depth 1 →

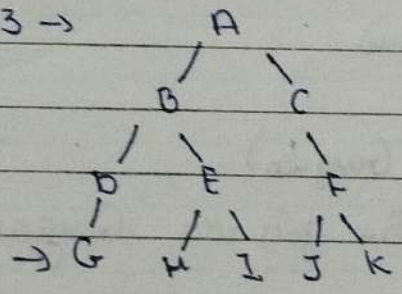


Depth 2 →



Not found increase depth to 3

Depth 3 →



found

Q 12] Explain Hill Climbing & its drawbacks in detail with example. Also state limitation of steepest-ascent hill climbing.

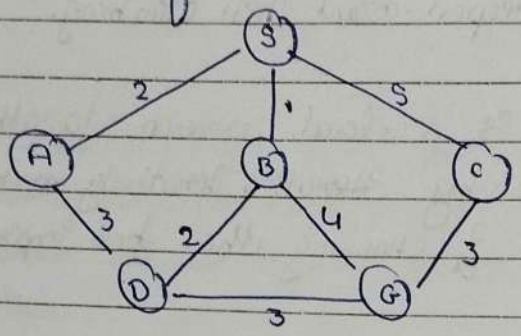
→ Hill climbing is a local search algorithm used to find an optimal solution by iteratively making small changes to the current state & choosing the best improvement.

Algorithm Steps:
1] Start with an initial solution (State)
2] Evaluate neighbouring state.
3] Move to the neighbour with the highest value.
4] Repeat until no better neighbours exist.

Example: (Hill climbing in Path Optimization): A robot trying to reach the highest hill top (goal) uses hill climbing. It moves upwards step by step, choosing the steepest 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.

Q10] find the route from S to G using BFS



A → Starts from source, put in queue with cost (initial 0) remove from queue calculate new distance (cur + edge weight) check if the current is the goal if not then put in queue & repeat.

children { (A,2), (B,1), (C,5) }
None of them is goal add in queue.

A,2	B,1	C,5

for A
children
(D,3)
dist : (D, 2+3)
Not a goal put in
queue | B,1 | C5 | D,5 |

for B
children { (D,2), (G,4) }
dist | D, 1+2 | G, 1+4 |
D is not a goal put in queue
G is goal
∴ Distance found!

for C
children (G, 3)
distance G, 3+5
(G, 8)
Goal found
But
Distance is 8 > 5
not accepted.

G is 5

Q.13] Explain Simultaneous annealing & write its algorithm.

→ 1] The process of annealing can be simulteneously with the Metropolis algorithm, which is based on Monte Carlo technique.
2] Monte Carlo ~~algorithm~~ technique: Model used to predict the probability of variety of outcome (Criteria).
3] This algorithm to generate a solution to combinatorial optimization problem assuming an analogy between them & physical many-particle system with the following equivalences:
- Solutions in the problem are equivalent to state in physical system.
- The cost of solution is equivalent to the 'energy' of state.

Algorithm Steps
① Initialize
- Choose an initial solution randomly.
- Set an initial temperature (T) (high value)
- Define a cooling rate (α)

② Repeat until stopping Condition is Met.
- Generate a new solution by making a small change to the current solution.
- Calculate the energy difference (ΔE) between the new & current solution.
- Acceptance Criteria.
  - If the new solution is better ($\Delta E < 0$), accept it.
  - If the new solution is worse ($\Delta E > 0$), accept it with probability $P = e^{-\Delta E/T}$,, T → temperature.
- Update $T = T \times \alpha$
- Repeat until T is close to 0 or no better solution.

Example of Simulated Annealing.

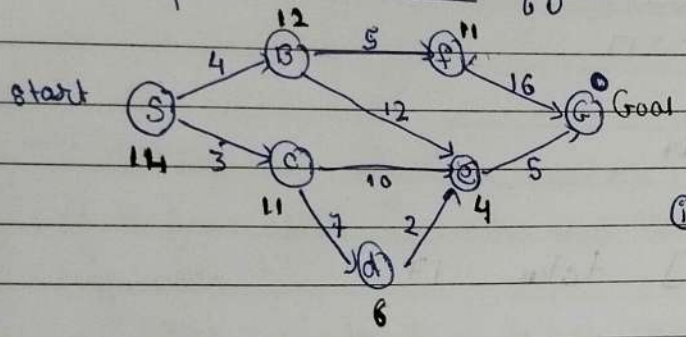
Imagine you are trying to find the lowest point in a hilly landscape.
> If you always move downhill, you might get stuck in small valley that is not lowest point
> If you sometimes jump to higher points, you are explore different areas and find a deeper valley.
> Over time, as the 'temperature' decreases, you stop making big jumps & focus on going downhill.
> Eventually, you reach the lowest possible valley (best solution)

Q14] Explain A* algorithm with a example.

→ i] A* is powerful graph traversal & path finding algorithm widely used in artificial intelligence & computer science.
ii] It is mainly used to find the shortest path between 2 node in graph, given the estimated cost of getting from the current node the destination node.
iii] The main advantage of the algorithm is its ability to provide an optimal path by exploring the graph in a more informed way compared to traditional search algorithm such as Dijkstra's algorithm.
iv] Algorithm A* combines the advantages of two other search algorithm: Dijkstra's algorithm & Greedy Best-~~Search~~ First Search.
v] A* guarantee to find optimal solution
The main idea of A* is to evaluate each node based on two parameters:

g(n) = actual cost to get from initial node, h(n) = Heuristic cost.
f(n) = g(n) + h(n).

Example: S → G of goal.



$f(s) = 0 + 14 = 14$

(i) S → B (4) S → C

$4 + 12 = 16$ $3 + 11 = 14$

(ii) path [SC]

SC → e

$g(n) = 3 + 10 = 13$

$h(n) = 4$

$f(n) = 13 + 4 = 17$

SC → d

$g(n) = 3 + 7 = 10$

$h(n) = 96$

$f(n) = 10 + 6 = 16$

S → B & SC → D have same cost we can traverse both.

(iii) for SB

SB → f

SB → e

SB → f

$f(n) = 5 + 4 = 11 = 20$

SB → e

$f(n) = 4 + 12 + 4 = 20$

values are higher that SC → d

iv) So let's explore SCD

SCD → e

$f(n) = 4 + 3 + 7 + 2 + 4 = 16$

SCD

Choose minimum which is 16 code over SBe SBf

iv) Scde → G

$g(n) = 3+7+2+5 = 17$

$h(n) = 0$

$f(n) = 17 + 0 = 17$

path [Scdeg] takes 17

v) If we want we can explore SBF to find whether it is to be done or not.

SBf → G

$g(n) = 25$, $h(n) = 0$

$f(n) = 25$

which is greater than SCDEG

~~form~~ ~~this~~ we ~~get to know~~ that/ h,

Ans : ~~SCDEG~~ S → C → D → E → G

Time Complexity = $O(V + E)$ V → vertices, E → edges

$= O(b^d)$ → b → branch factor

d → depth

Space complexity = $O(b^d)$

5] Explain Mini max Explain Minimax Algorithm & Draw game tree for Tic Tac Toe game.

→ i] Minimax algorithm evaluate decision based on the present status of the game. This algorithm need deterministic environment with perfect/exact information.

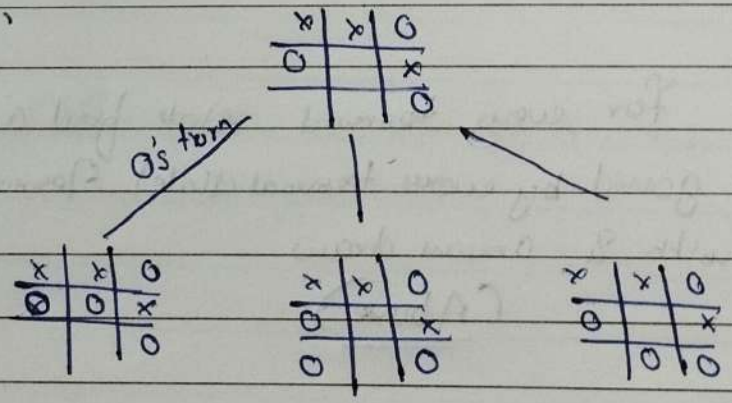
ii] Minimax algorithm directly implements the defining equation. Every time based on the successor state minimax value is calculated with the help of simple recursive computations.

iii] In case of minimax algorithm the selected action with highest minimax value should be equal to the best possible payoff (outcome) against best play.

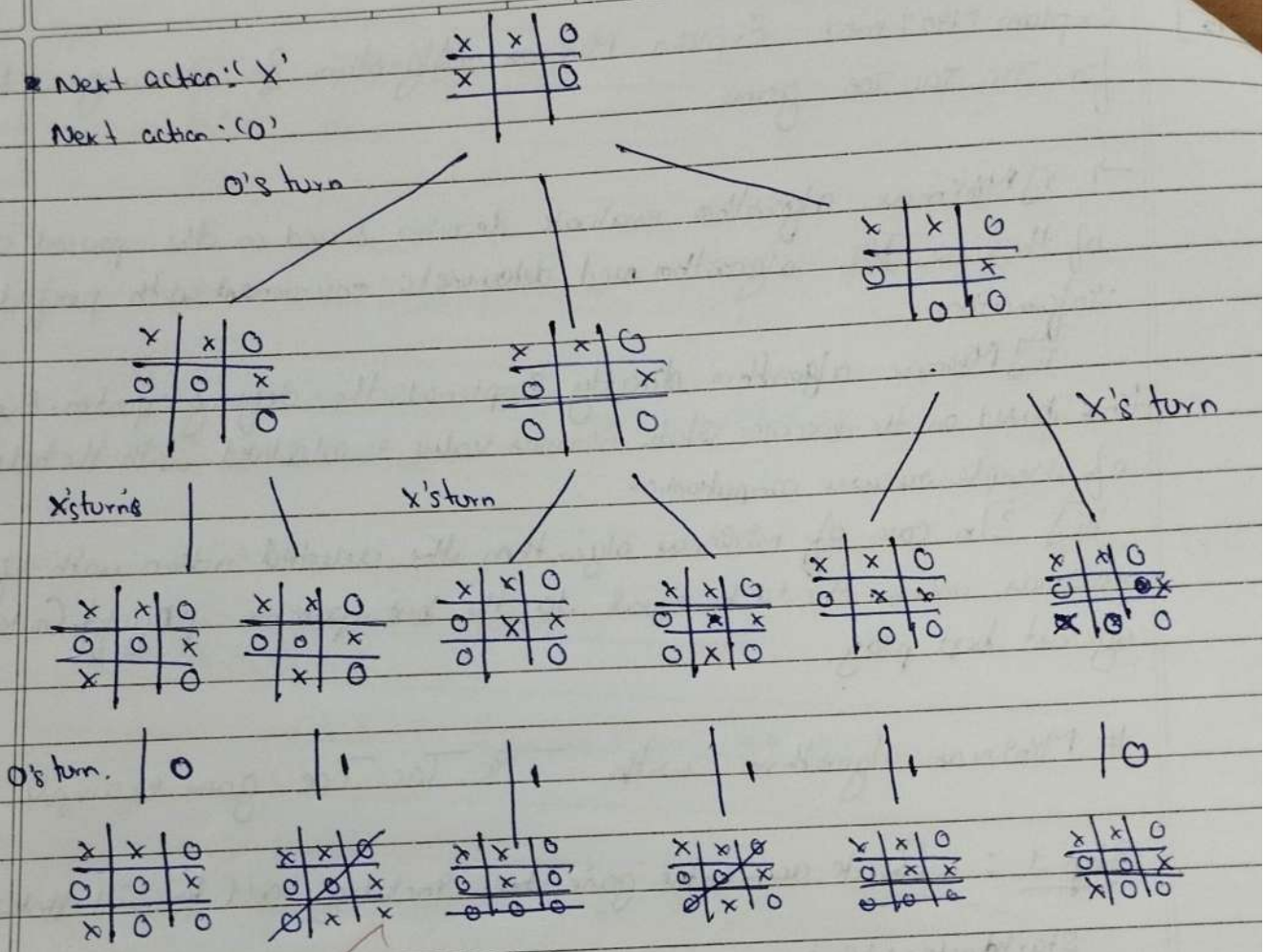
Minimax algorithm with Tic Tac Toe game example

Step 1: Create an entire game tree including all terminal states.

Start action: 'O'



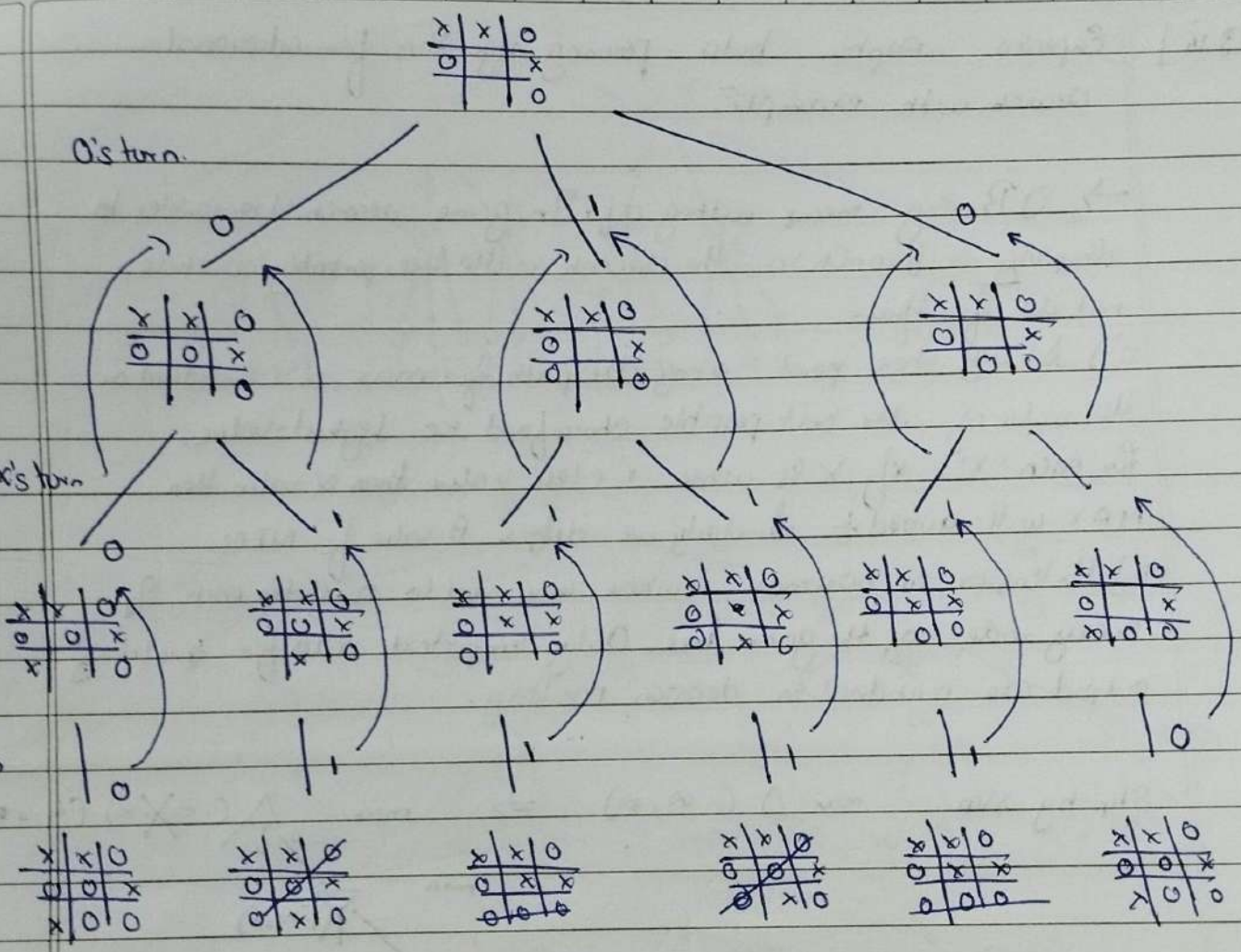
Next action



Step 2: For every terminal state find out utility (playoff points gained by every terminal state). Terminal position where 1 means win & 0 means draw (Above)

Step 3: Apply Min & Max operators on the nodes of the present stage & propagate the utility values upward in the three.

Step 4: With the max (of the min) utility value (playoff value) select the action at the root node using minimax decision



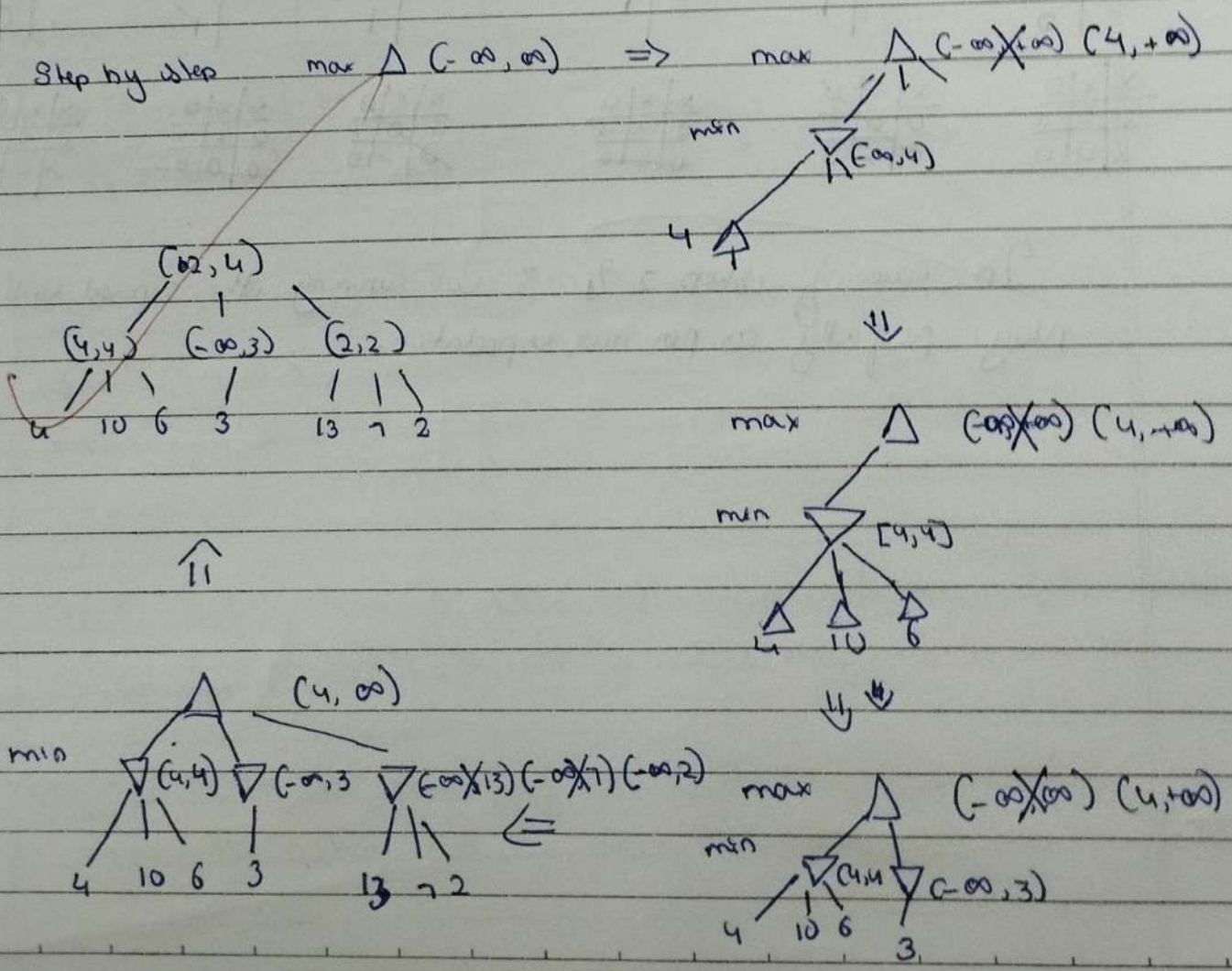
In case of step 2 & 3 we assuming the opponent will play perfectly as per our expectation.

Q16] Explain Alpha beta pruning algorithm for adversarial search with example.

→ i] Pruning means cutting off. In game search it resembles to skipping a branch in the search in the tree, probably which is not so fruitful.

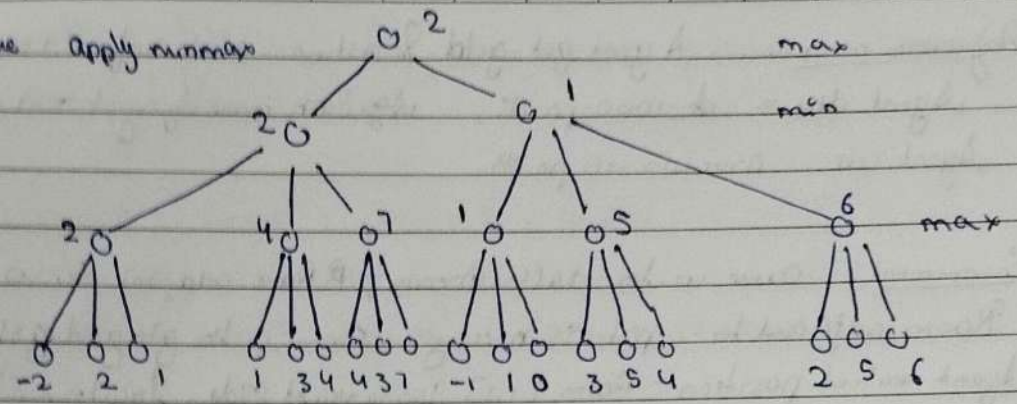
ii] At any choice point along the path for max, α is considered as the value of the best possible choice found i.e highest-value. For each 'x', if x is worse i.e less value than α value then, MAX will avoid it. Similarly we define β value for MIN.

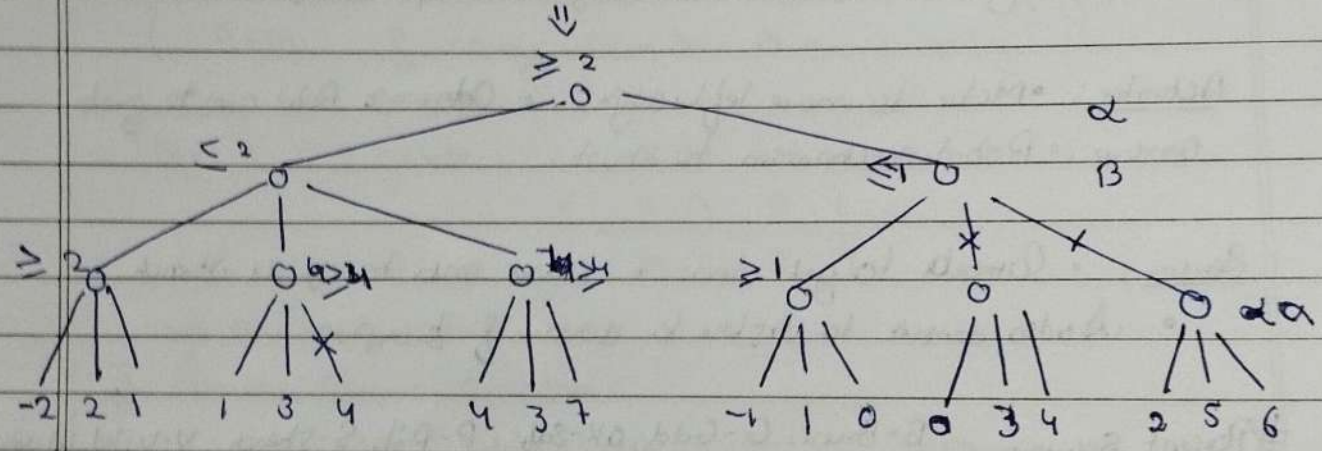
iii] Extension in minimax where no need to consider each & every node of the game tree. Only important nodes for quality output are considered in decision making.



$\alpha \geq \beta$ → pruned

Example apply minmax





Q17] Explain wumpus world environment giving its PEAS description. Explain how percept sequence is generated.

→ i] It is computer Game in which agent explore a cave with 16 rooms (4×4). Each room connected by walkway.

ii] Lurking there in cave is the Wumpus, a beast that eat any agent that enter room. Agent starts from [1,1]

iii] Some room contain bottomless pits that trap any agent that wander.

iv] Occasionally there is heap of gold in a room.

v] The goal is to collect the goal & exit the world without being eaten.

warnings of outbreaks. It

Performance measure: Agent get gold & return back safe = +1000 points
Agent dies = -1000 points, Each move of agent = -1 point
Agent use arrow = -10 points.

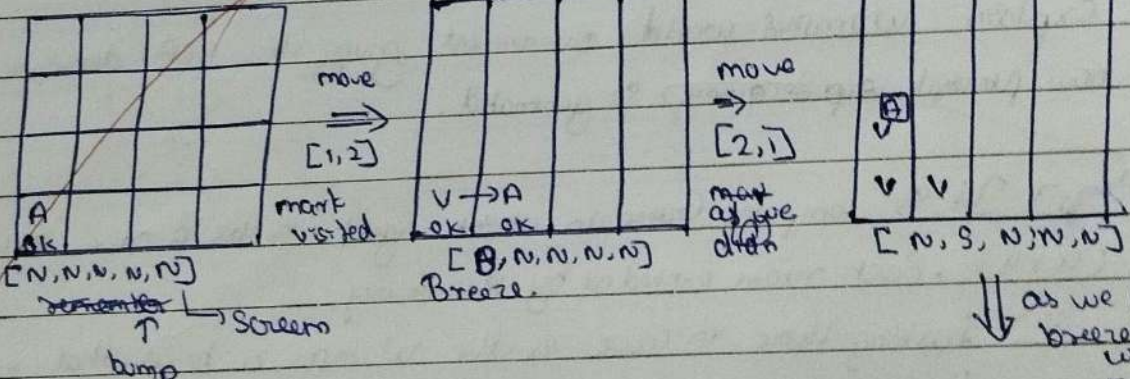
Environment: Cave with 16 (4×4) room, Room adjacent to wumpus is stinky Room adjacent to pit is breezy, Room with gto gold glitter.
Agent initial position - Room [1,1] facing right side, Location of Wumpus, Gold & 3 pits can be anywhere except [1,1]

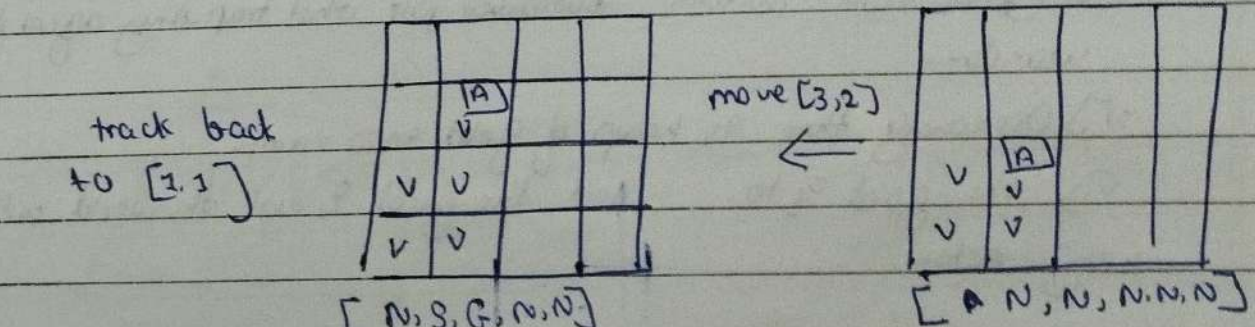
Actuator: • Motor to move left, right, • ~~Odour~~ Robot arm to grab arrow, • Robot mechanism to shoot.

Sensor: • Camera to get view. • Odour sensor to smell stench.
• Audio sensor to listen to scream & bump.

#Percept Sequence B-Breez, G-Gold, OK-Safe, P-Pit, S-Stench, V-Visited, W-Wumpus
N-none.

1)





Solve the following Cryptoarithmetic problems

1. SEND + MORE = MONEY

→ $1000S + 100E + 10N + D = SEND$

$1000M + 100O + 10R + E = MORE$

$10000M + 1000O + 100N + 10E + Y = MONEY$

1) Since M has value equal to carry generate by S+M, & carry can be 0 or 1 only.

So, **M=1**, Substitute.

$$\begin{array}{r} S\ E\ N\ D \\ +\ 1\ O\ R\ E \\ \hline 1\ O\ N\ E\ Y \end{array}$$

~~we can conclude that~~.

2) form above equation.

we know that

$C_4 = 1$, $S + 1 + C_3 = 10 + O$

$\therefore S = 8 + O$

$S = 8 + 1$

↳ not possible.

O = 0 (zero)

or $C_4 = 0$

$S + 1 = 10 + O$

$S = 9 + O$

S can't be greater than 9

So, O = 0 (zero)

then **S = 9**

S can be 8, or 9 but $S = 8 + M = 1 < 10$

so ~~8~~ S = 8 not pos

S = 9

Substitute

$$\begin{array}{r} C_4\ C_3\ C_2\ C_1 \\ 9\ E\ N\ D \\ 1\ O_{(z)}\ R\ E \\ \hline 1\ O_{(z)}\ N\ E\ Y \end{array}$$

warnings of outbreaks, & provide insights for policy makers

iii] There is no carry from $C_3 \to C_4$

~~$E + O_{(2)} + 1 = O$~~

Since $E \neq N$

so there is carry from $C_2 \to C_3$

$E + 1 + O_{(2)} = N \Rightarrow E + 1 = N$

$N + R = 10 + E$

$E + 1 + R = 10 + E$

$R = 9$

not possible

$C_1 \to C_2 = 1$

$N + R + 1 = 10 + E$

$E + 1 + R + 1 = 10 + E$

$\boxed{R = 8}$

S E N D

1 0 8 E

1 O N E Y

iv] There is $C_1 \xrightarrow[+1]{} C_2$ so

$Y \neq 0$ & $Y \neq 1$, $E + 1 = N$

$D + E \geq 12$

D & $E \neq 8, 9$

$7+5$, $7+6$

$E = 7$

$D = 5$ or 6

$E + 1 = N$ ✗

$7 + 1 = N$

$N = 8$ not possible

if $E \neq 7$ then $D \neq 5$ or 6

$D = 7$

$E = 5/6$

$D = 7$

$E = 5/6$

$E + 1 = N$

$E + 1 = N$

$6 = N$

$E + 1 = N$

$6 + 1 = N$

$7 = N$ not possible

since $D = 7$

$\boxed{D = 7}$, $\boxed{N = 6}$ $\boxed{Y = 2}$

$\therefore$ 9

9 5 6 7

+ 1 0 8 5

1 0 6 5 2

G is good

not acceptable

Distance found

8.19] Consider the axioms.

i] Represent in First order Predicate logic.

a] All people who are graduating are happy
→ ∀(x) (GradGraduating(x) → Happy(x))

b] All happy people are smiling
→ ∀(x) (Happy(x) → Smiling(x))

c] Someone is graduating
→ ∃x Graduating(x)

ii] Convert Each formula to Clause form.

(a) Convert to Implication - free form.
- Rewrite implication using $P \to Q \equiv \neg P \vee Q$
  - i] ∀(x) (¬Graduating(x) ∨ Happy(x))
  - ii] ∀(x) (¬Happy(x) ∨ Smiling(x))
  - iii] ∃(x) Graduating(x)

(b) Standardizing variable
Ensure variable in diff statement are distinct. x, y, z for each

(c) Convert existential quantifier to new constant a.
- Graduate(a)

(d) Convert to clause form.

1] ¬Graduating(x) ∨ Happy(x)
2] ¬Happy(y) ∨ Smiling(y)
3] Graduating(a)

iii] Prove 'Someone is Smiling' using Resolution Technique.

→ we need to prove $\exists(x)$ Smiling(x)

1. Assume negation of Goal
   Assume $\neg$ Smiling(x) for proof by Contradiction.

2. Resolution Step
   Graduating(a) ⟶ from clause 3
   $x=a$, $\neg$ Graduating$(a) \vee$ Happy(a) ⟶ from (1)
   • Since Graduating(a) is true, resolve with $\neg$ Graduating(a) to get Happy(a).

3. From clause 2 substituting $y=a$.
   $\neg$ Happy$(a) \vee$ Smiling(a)
   → Since Happy(a) is true, resolve with $\neg$ Happy(a) to get Smiling(a).
   ∴ This contradict $\neg$ Smiling(x), so our assumption is false.
   5. Hence 'Someone is Smiling' is proved

8 20] ~~Modus Ponens Example~~ Explain Modus Ponen's with Example in detail.

→ Modus Ponens is a fundamental rule of inference in logic. It is used to derive conclusion from conditional statements.

G is goal
∴ Distance found!
not accepted

G is 5

Rules of Modus Ponen.

If we have 2 premises

1] $P \rightarrow Q$ (if P true, then Q is true)
2] P (P is true)

Then we can conclud that Q is true.

Example: 1] If it is raining, then ground is wet
• Rain(x) → WetGround(x)
2] It is Raining
• Rain(x)

Conclusion: Since both premises are true, we conclude that : The ground is wet.

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

→ (I) Forward chaing : Also known as forward reasoning, is method of inference that start with known fact & applies rule to derive new conclusion. Process continues until goal is reached or no new fact can be inferred.

Steps
Start with initial facts → Apply inference rules to generate new fact → Continue until desire conclusion is reached

Example
(1) Given fact
1] If person has a fever & cough, they may have flu.
2] John has fever.
3] John has cough.

(2) Rules

1] If person has fever & cough, then they have flu.
2] If person has flu, they should rest & drink fluid.

(3) Inference Using Forward Reasoning.

1] John has fever & cough.
2] Apply Rule 1 → John has flu.
3] Apply Rule 2 → John should rest & drink fluid.

Conclusion: John should rest & drink fluids.

(4) Backward reasoning: Also aka Backward chaining, as inference method that starts with goal & work backward to check if the given facts support it.

Start with goal → Look for rule that lead to goal → Check if known facts support these rules
If not keep working backward until the facts confirm the goal. ←

Example

(1) Goal → Prove John has flu.

(2) Rules

1] If person has fever & cough, then they have flu.
2] If a person has flu, they should rest & drink fluids.

(3) Backward Reasoning steps

1] To prove John has flu, check if he has fever and cough.
2] Check if John has fever. (Yes)
3] Check if John has cough (Yes)
4] ~~If~~

Conclusion: John has flu.