Searching

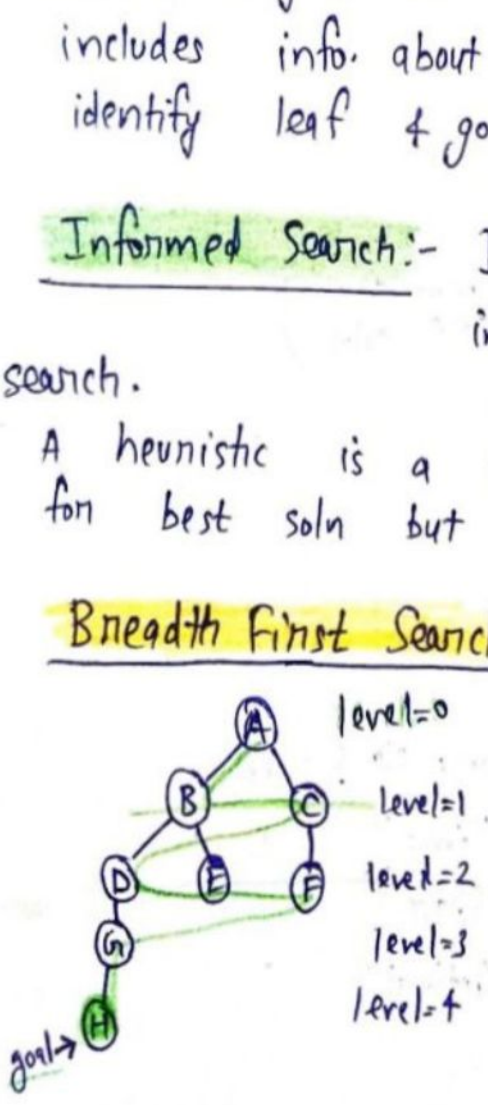
- Uninformed | Blind
  - Breadth first Search
  - Uniform cost search
  - Depth first search
  - Depth limited Search
  - Iterative Deeping depth first Search
  - Bidirectional Search
- Informed | Heuristic
  - Best first Search
  - A* search
  - AO* Algorithm
  - Problem Reduction
  - Hill Climbing

Uninformed | Blind Search :- It doesn't contain any domain knowledge such as closeness, location of goal. It operates in Brute force way, as it only includes info. about how to traverse the tree & how to identify leaf & goal nodes.

Informed Search :- It uses domain knowledge, the problem inf. is available which can guide the search.

A heuristic is a way which might not always be guarented for best soln but guaranteed to find a good soln is reasonable time

Breadth First Search (BFS) :- Traversing a tree



FIFO Queue

Time Complexity :- $b^0+b^1+b^2+b^3$
$\dots b^d$

b = max.branching
d = depth

$1+b+\dots b^d$

$O(b^d)$

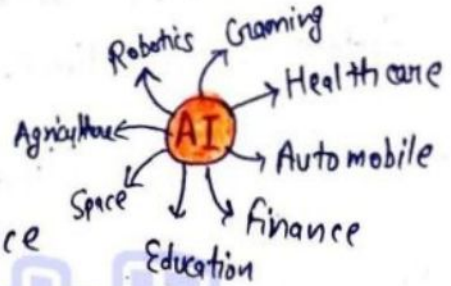
Space :- $O(b^d)$

Artificial Intelligence

Artificial → man made thing

Intelligence → Thinking power

→ AI refers to simulation of human intelligence in machines that are programmed to <u>think like humans</u> & mimic their actions.



Type of AI

- Weak / Narrow
- Strang
- Superintelligence

Components of AI ⇒

- Applications :— Image Recognition
  Speech Recognition
  Chatbots
  Natural languageGeneration
  Sentiment Analysis
- Types of Models:—
  Deep learning
  Machine learning
  Neutral Networks
- S/W & H/W for training & running models:—
  GPUs , parallel processing tools(like spark)
  , cloud data storage & compute platforms
- Programming langugges:— Python, TensonFlow, Java, C

Different approaches of AI

- Reactive Machine
- Limited Memory
- Theory of Mind
- Self-Awareness

1. Reactive Machine ⇒ These machines are most basic form of AI application
Eg - Deep Blue, IBM's chess-playing supercomputer.

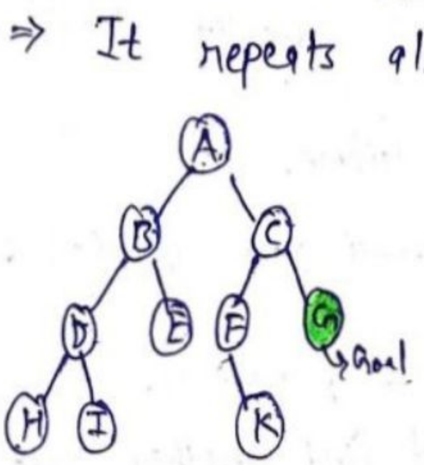
AI teams do not use any training sets to feed machines nor do latter store data for future references. Based on move made by opponent, the machine decides next move.

Iterative Deepening DFS:- It is a combination of DFS & BFS.

→ This algo. performs depth-first search up to a certain "depth-limit" & it keeps increasing the depth limit after each iteration until the goal node is found.

Adv. ⇒ This algo. combines the benefit of BFS's fast search & DFS's memory efficiency.

Dis ⇒ It repeats all work of previous phase.



level 0 DL=0 I_1: A

DL=1, I_2: A, B, C

level 1 DL=2, I_3: A, B, D, E, C, F, G

level 2 DL=3 I_4:- A, B, D, H, I, E, C, F, K, G ✓

level 3 4th ✓

Bi Directional Search:- It does forward search & backward search.

It uses BFS, DFS, DLS etc.

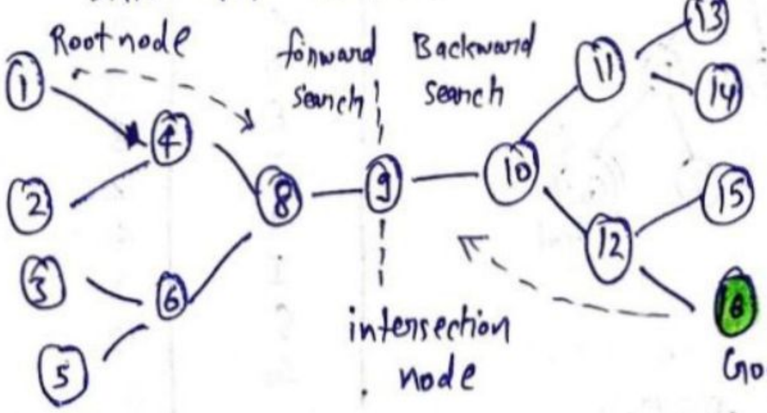
→ one start from start node and other from goal node.

→ Stop when intersect.

Adv. → fast, less memory

Dis. → Implementation is difficult.

→ One should know the goal state in advance.



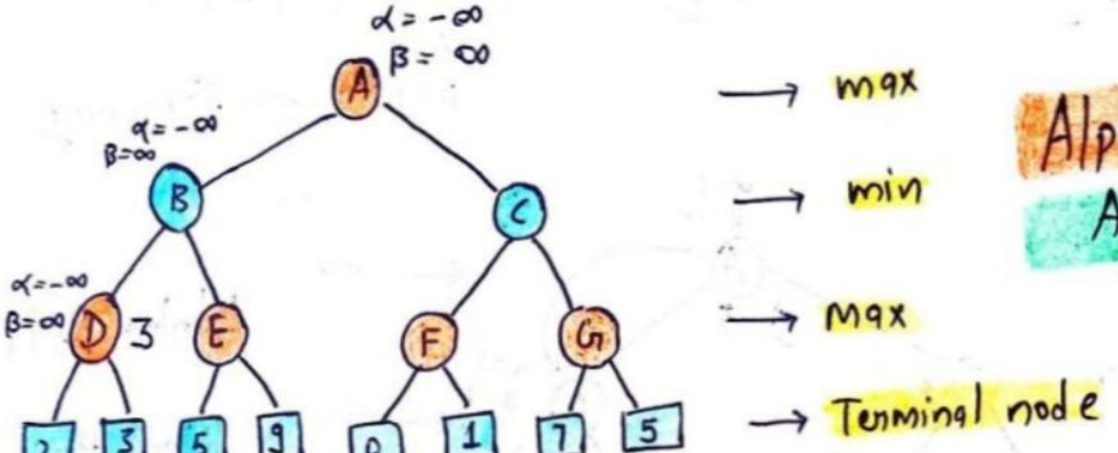
Alpha :- The best (highest value) choice we have found so far at any point along the path of maximizer. (Max player) The initial value of alpha is $-\infty$.

Beta:- The best (lowest value) ——— of minimizer. $+\infty$. (Min player)

→ It removes all the nodes which are not really affecting the final decision.

7

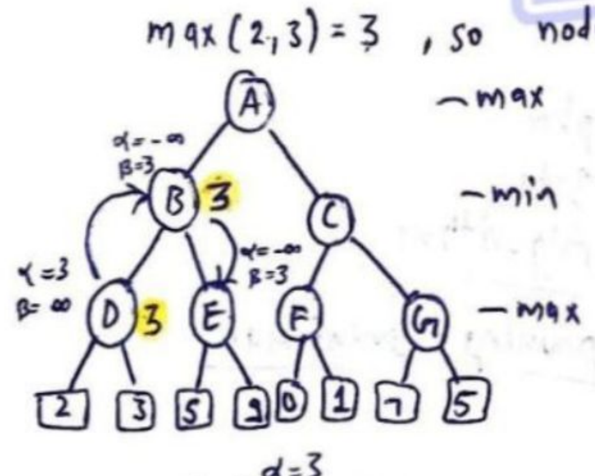
Alpha Beta Algorithm



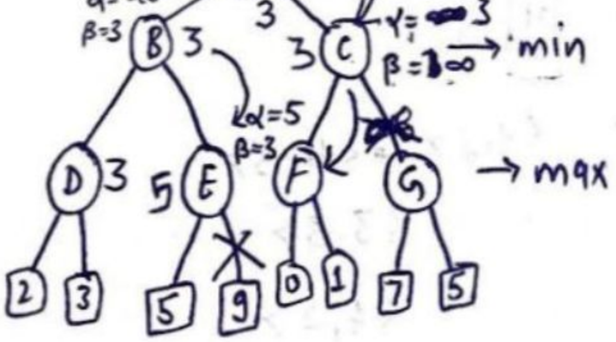
→ while Backtracking, node values will be passed to upper nodes instead of values of alpha & beta.

→ We will only pass alpha, beta values to child nodes.

max (2, 3) = 3 , so node D value will also 3. , $\alpha=3$



$\alpha=-\infty$, $\beta = \min(3, \infty) = 3$



$\max(-\infty, 5) = 5$ = node

$\alpha=5$ $\boxed{\alpha >= \beta}$

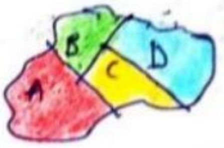
B to A , backtracking

$\alpha = \max(-\infty, 3) = 3$

$\alpha=3$

Constraint Satisfaction Problem [CSP] ⇒

- → CSP consists of 3 components V, D, C
- → V is set of variables { v_1, v_2 -- v_n }
- → D is set of Domains { D_1 D_2 --- D_n } one for each variable.
- → C is set of constraints that specify allowable combination of values C_i = (scope, rel)



D = { R, B, G }
V = { A, B, C, D }

C_1 = { (A, B), A ≠ B }
C_1 = { (A, B), (R, G) (G, R) (R, B) (G, B) (B, R) }

crytarithmetic

```
  TWO
+ TWO
 FOUR
```

Ay - (B) (i)

Mini-Max Algorithm :-

(5)

- → It is a recursive or backtracking algorithm which is used in game theory and decision-making.
- → It is mostly used for game playing in AI. Such as Chess, tic-tac-toe, go, checkers etc.
- → There are 2 players MAX & MIN.
- → Max for maximized value & MIN for minimized value.
- → minimax performs a DFS algorithm

Properties of Mini-Max :-

Complete - Yes
Optimal - Yes
Time Complexity - $O(b^m)$
Space Complexity - $O(bm)$

Limitation :-

It is slow for complex games such as chess, go etc. coz these have branching factor. So this limitation of minimax can be improved from alpha-beta pruning.

A* search Algorithm :- It is most commonly known form of best-First search.

→ It uses h(n) & cost to reach the node n from start state g(n).

→ It has combined features of UCS & greedy best-First Search.

$$\boxed{f(n) = g(n) + h(n)}$$

Estimated Total cost

Cost to reach node n from start state

cost of path from n to goal

Step-① Place starting node in Open list.

Step-② Check if open list is empty.

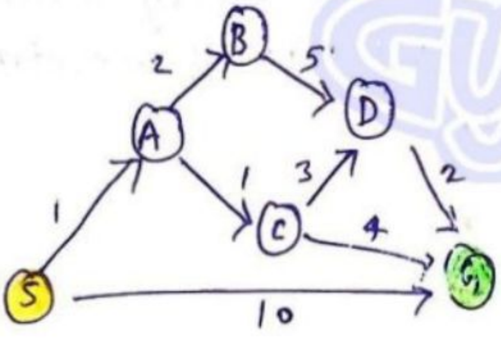
Step-③ Select node from open list which has smallest value of evaluation function (g+h)

Step-④ Expand n and generate all of its successors & put n into Closed list.

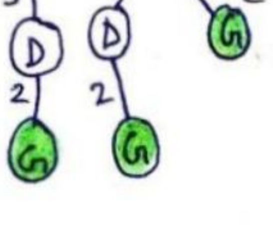
Step-⑤ If node n' is already in open & closed list, then it should be attached to back pointer which reflects lowest g(n') value.

Step-⑥ Return to step-②

A*



State	h(n)
S	5
A	3
B	4
C	2
D	6
G	0



0+5=5

Initial {S, 5}

Iteration-1:- {(S→A, 4), (S→G, 10)}

I_2:- {(S→A→B, 7), (S→A→C, 4), (S→G, 10)}

I_3:- {(S→A→C→D, 11), (S→A→C→G, 6), (S→G, 10), (S→A→B, 7)}

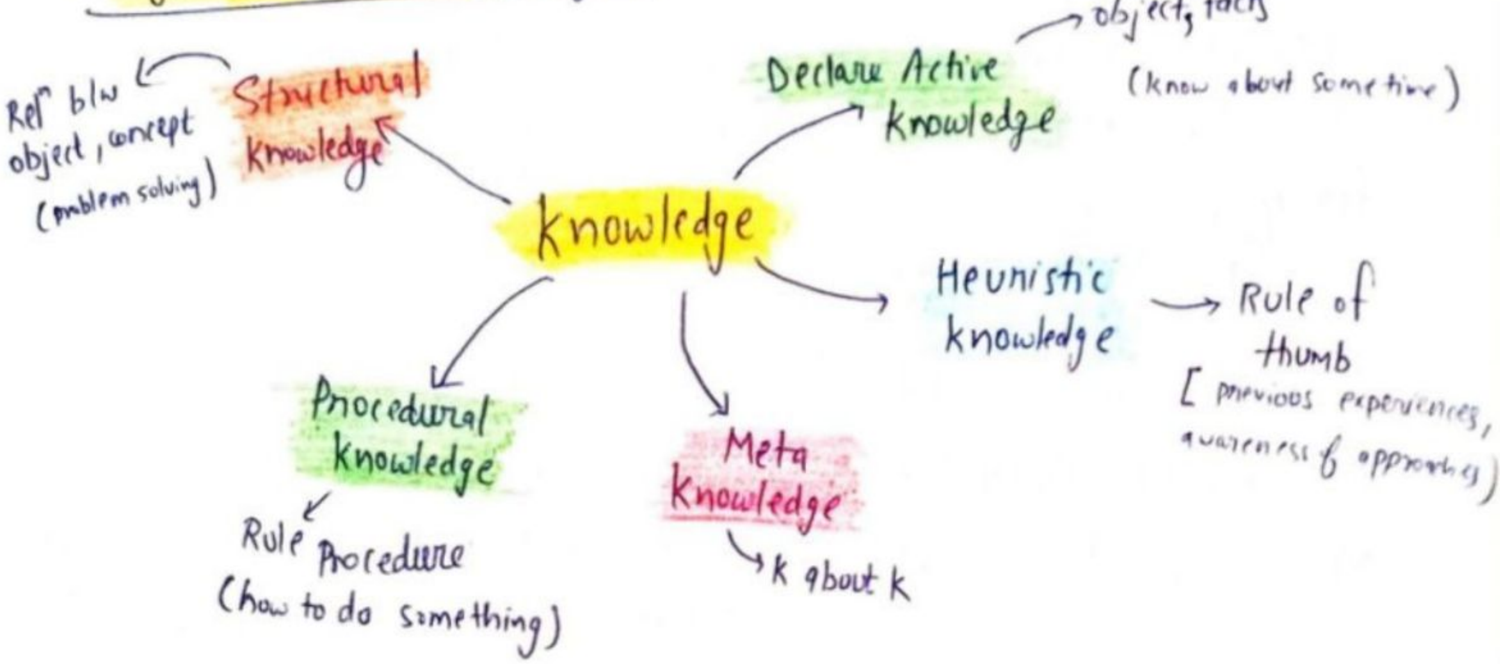
S→A→C→G, ⑥

Knowledge Base Representation & Reasoning ⇒

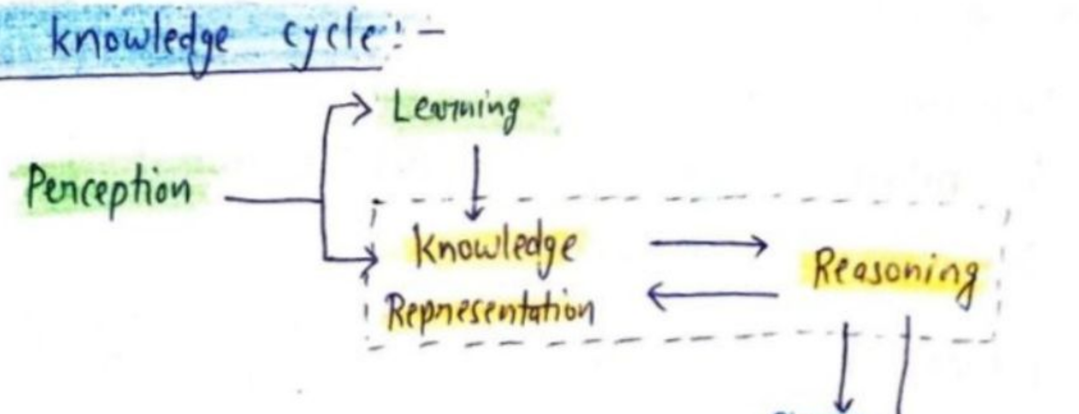
Knowledge Based Agent:- It is those agents who have capability of maintaining an internal state of knowledge, reason over that knowledge, update their knowledge after observations and take actions.

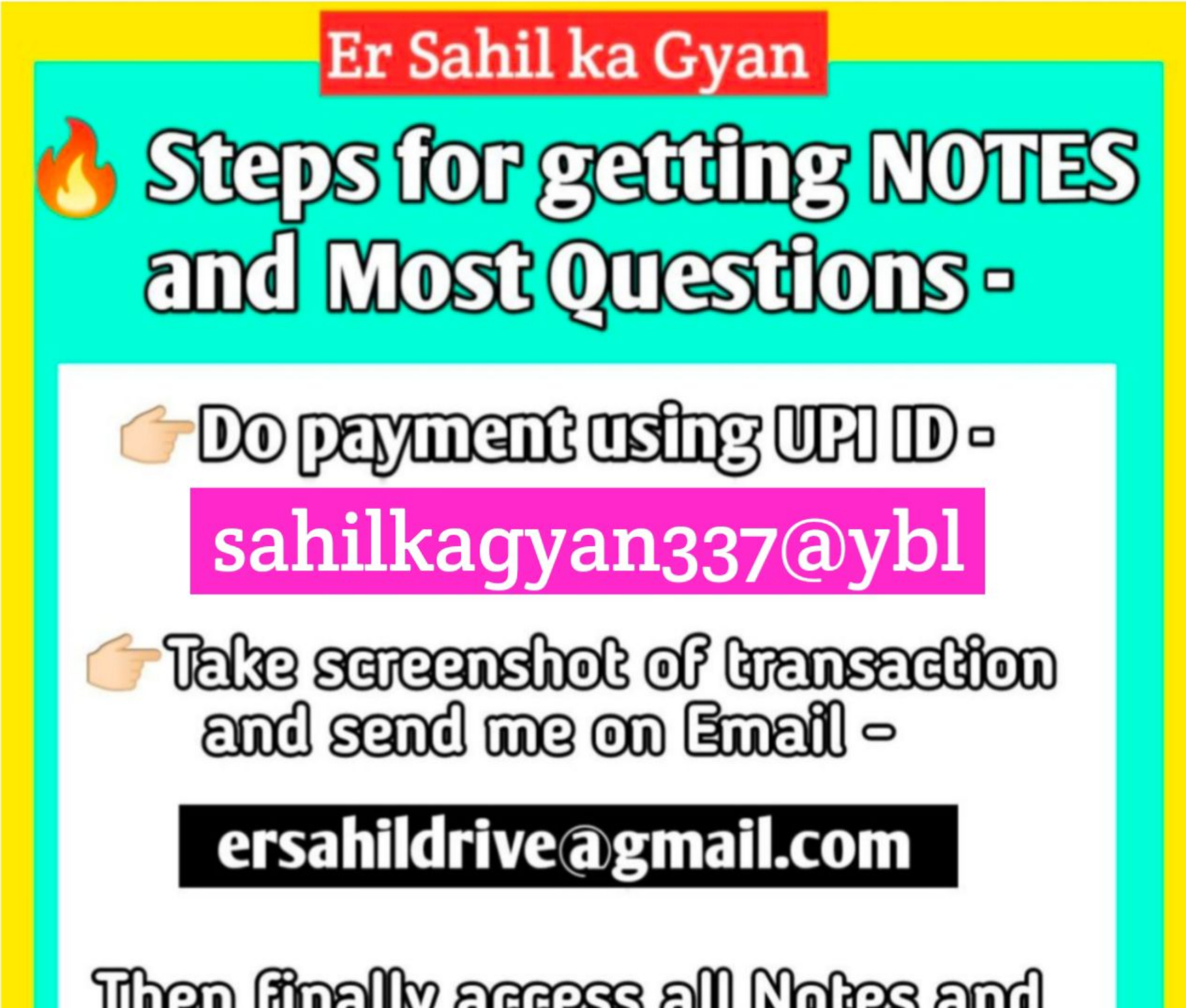
→ knowledge Base is required for updating knowledge for an agent to learn with experiences and take action as per the knowledge.

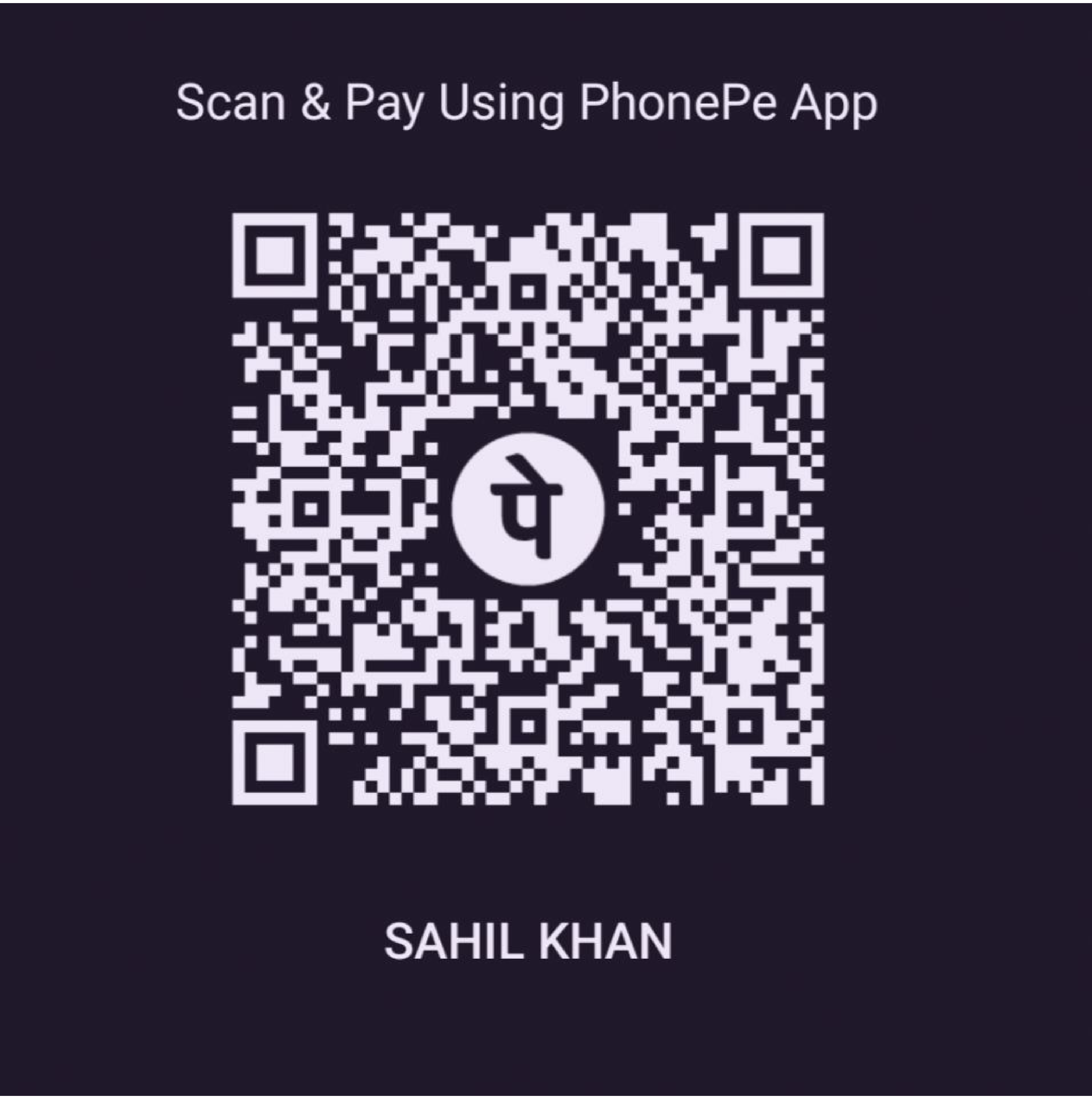
Types of knowledge ⇒


Knowledge
Structural Knowledge
Rel^n b/w object, concept (problem solving)
Declare Active knowledge
object, facts (know about sometime)
Heuristic knowledge
Rule of thumb [previous experiences, awareness of approaches]
Procedural knowledge
Rule Procedure (how to do something)
Meta Knowledge
k about k

AI knowledge cycle:-







Scan & Pay Using PhonePe App

SAHIL KHAN