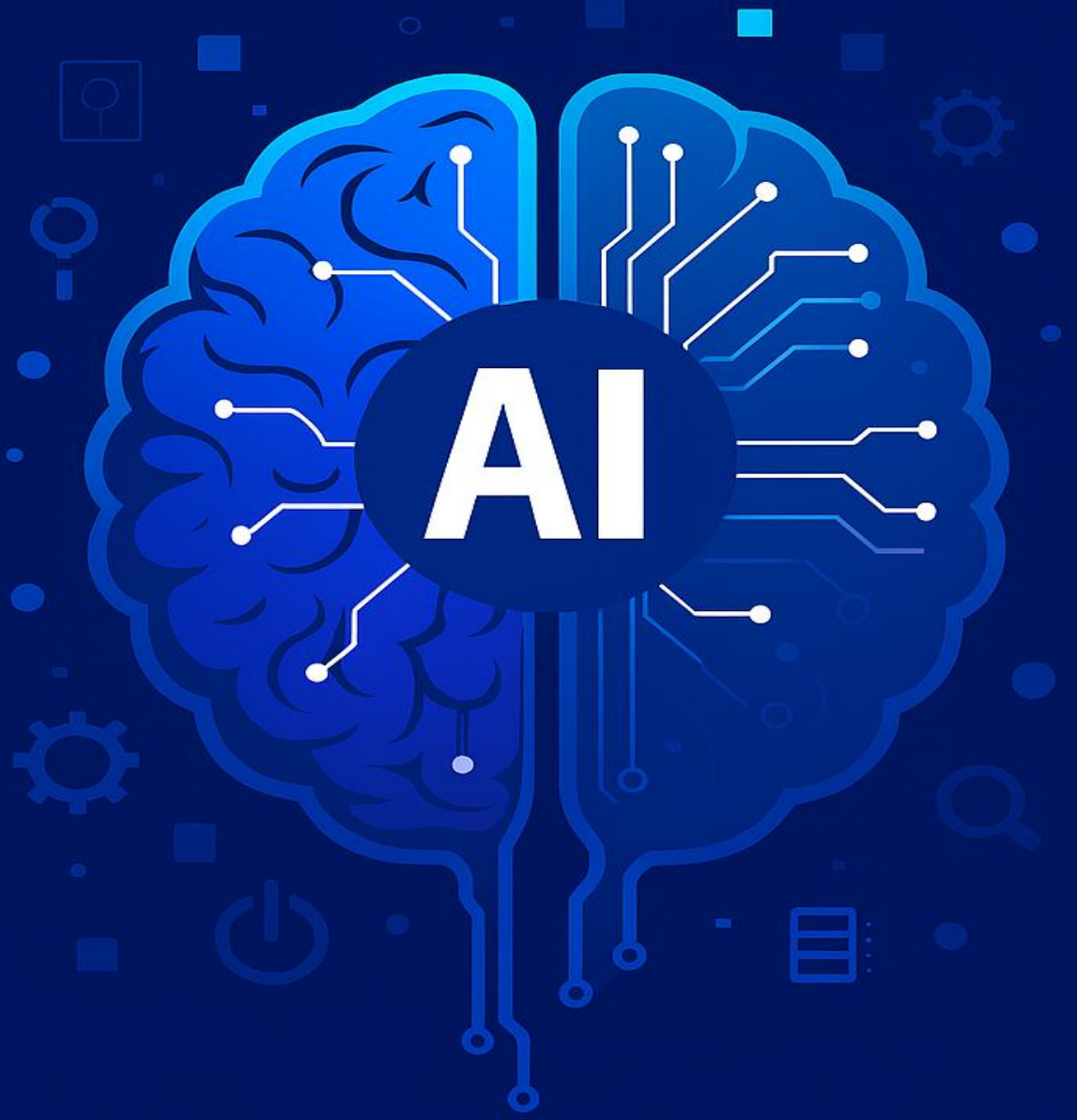
ARTIFICIAL INTELLIGENCE

TOPIC :

DEPTH LIMITED SEARCH

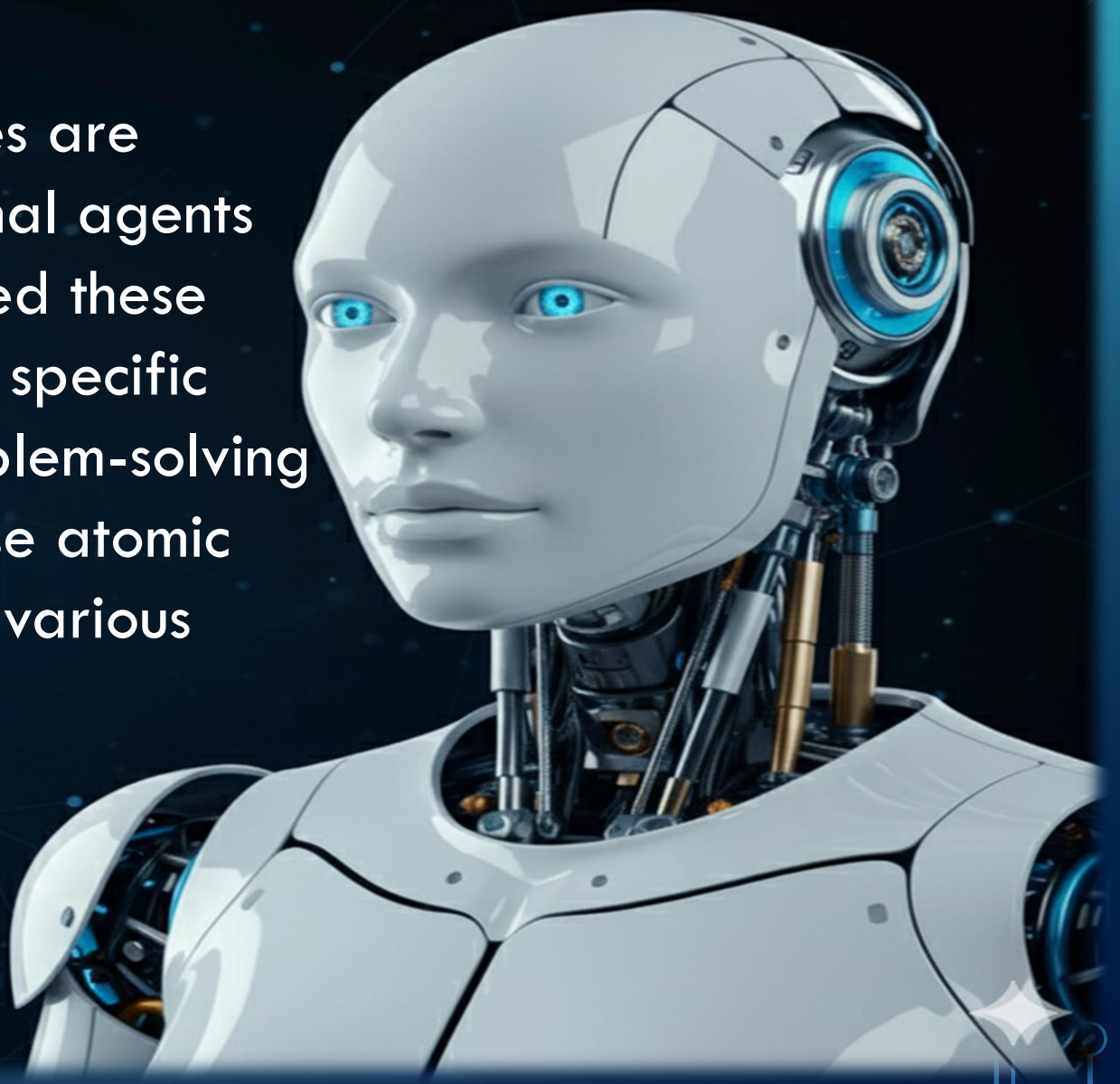
ITERATIVE DEEPENING DEPTH-FIRST SEARCH



Presented by:- Aman Kr. Gupta

INDRODUCTION TO SEARCH TECHNIQUES

"In Artificial Intelligence, Search techniques are universal problem-solving methods. Rational agents or Problem-solving agents in AI mostly used these search strategies or algorithms to solve a specific problem and provide the best result. Problem-solving agents are the goal-based agents and use atomic representation. In this topic, we will learn various problem-solving search algorithms."



Search Algorithm Terminologies:

Search: Searching is a step by step procedure to solve a search-problem in a given search space. A search problem can have three main factors:

1. Search Space: Search space represents a set of possible solutions, which a system may have.

2. Start State: It is a state from begins the search.

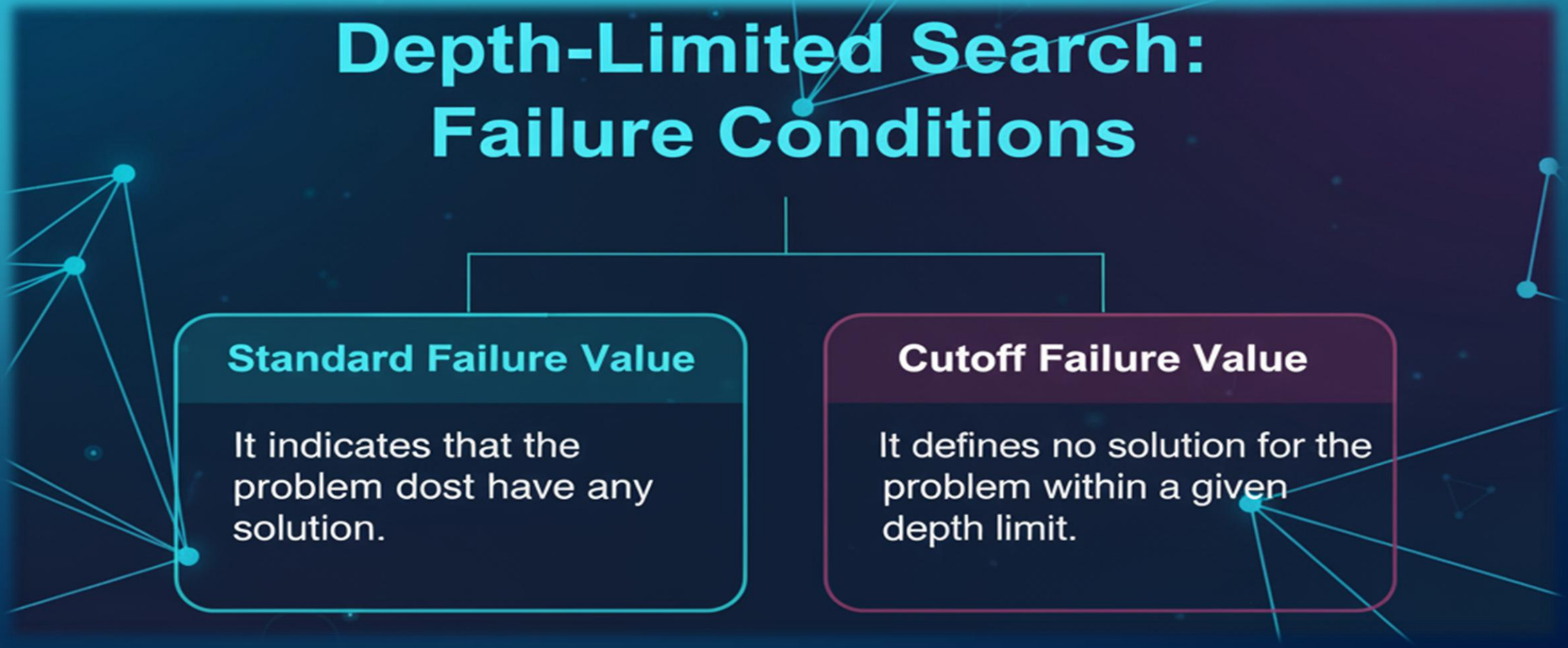
3. Goal test: It is a function which where agent observe the current state and returns whether the goal state is achieved or not.

Properties of Search Algorithm

1. ***Completeness***: A search algorithm is said to be complete if it guarantees to return a solution if at least any solution exists for any random input.
2. ***Optimality***: If a solution found for an algorithm is guaranteed to be the best solution (lowest path cost) among all other solutions, then such a solution is said to be an optimal solution.
3. ***Time Complexity***: Time complexity is a measure of time for an algorithm to complete its task.
4. ***Space Complexity***: It is the maximum storage space required at any point during the search, as the complexity of the problem.

Depth Limited Search

A depth-limited search algorithm is similar to depth-first search with a predetermined limit. Depth-limited search can solve the drawback of the infinite path in the Depth-first search. In this algorithm, the node at the depth limit will be treated as if it has no successor nodes further.



Depth Limited Search(Algorithm):-

- *A depth-limited search algorithm is similar to depth-first search with a predetermined limit.*

- *Depth-limited search can solve the drawback of the infinite path in the Depth-first search.*

- *In this algorithm, the node at the depth limit will treat as it has no successor nodes further.*

- *(It may not be optimal if the problem has more than one solution.)*

Example:-

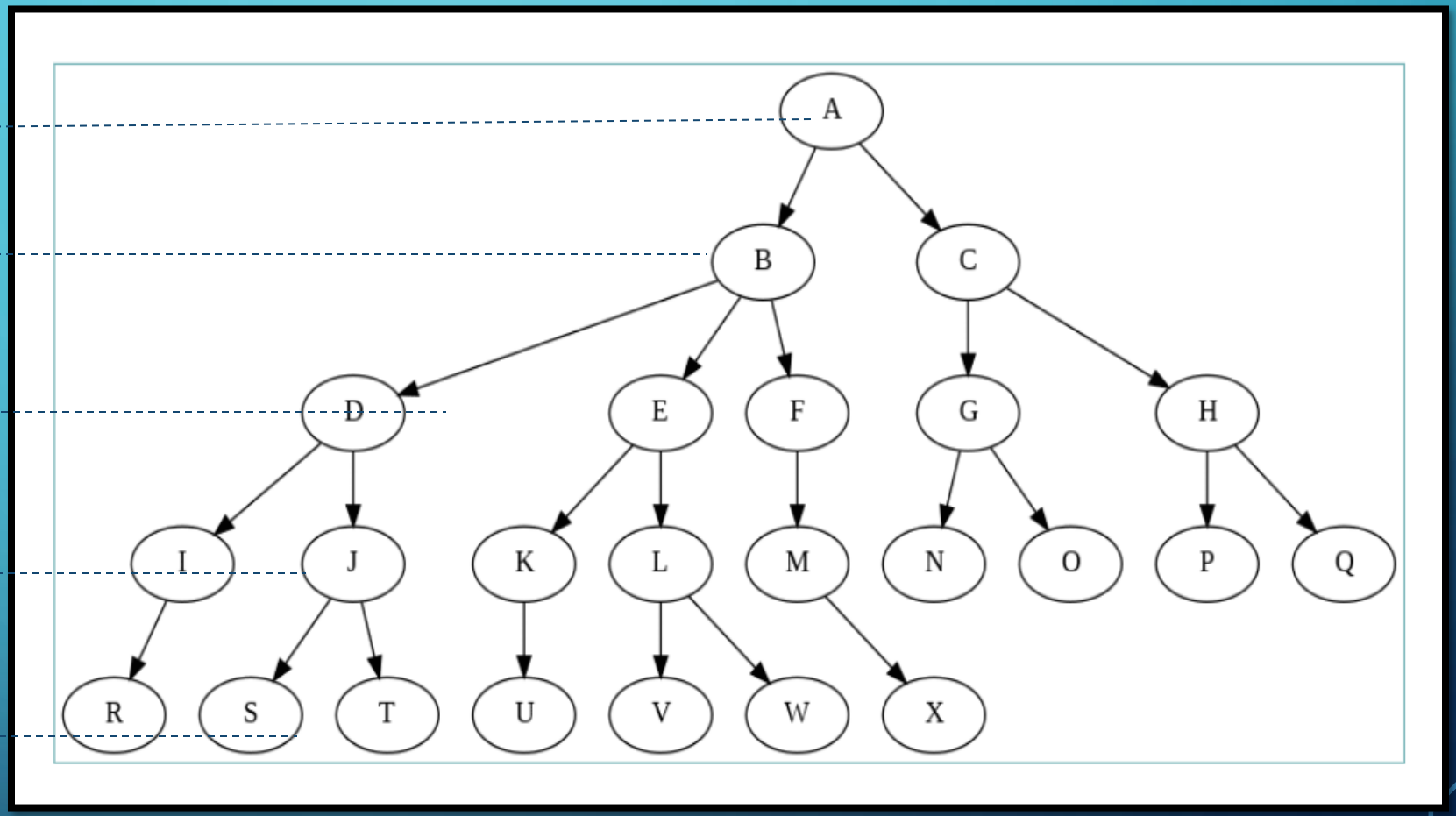
Level 0

Level 1

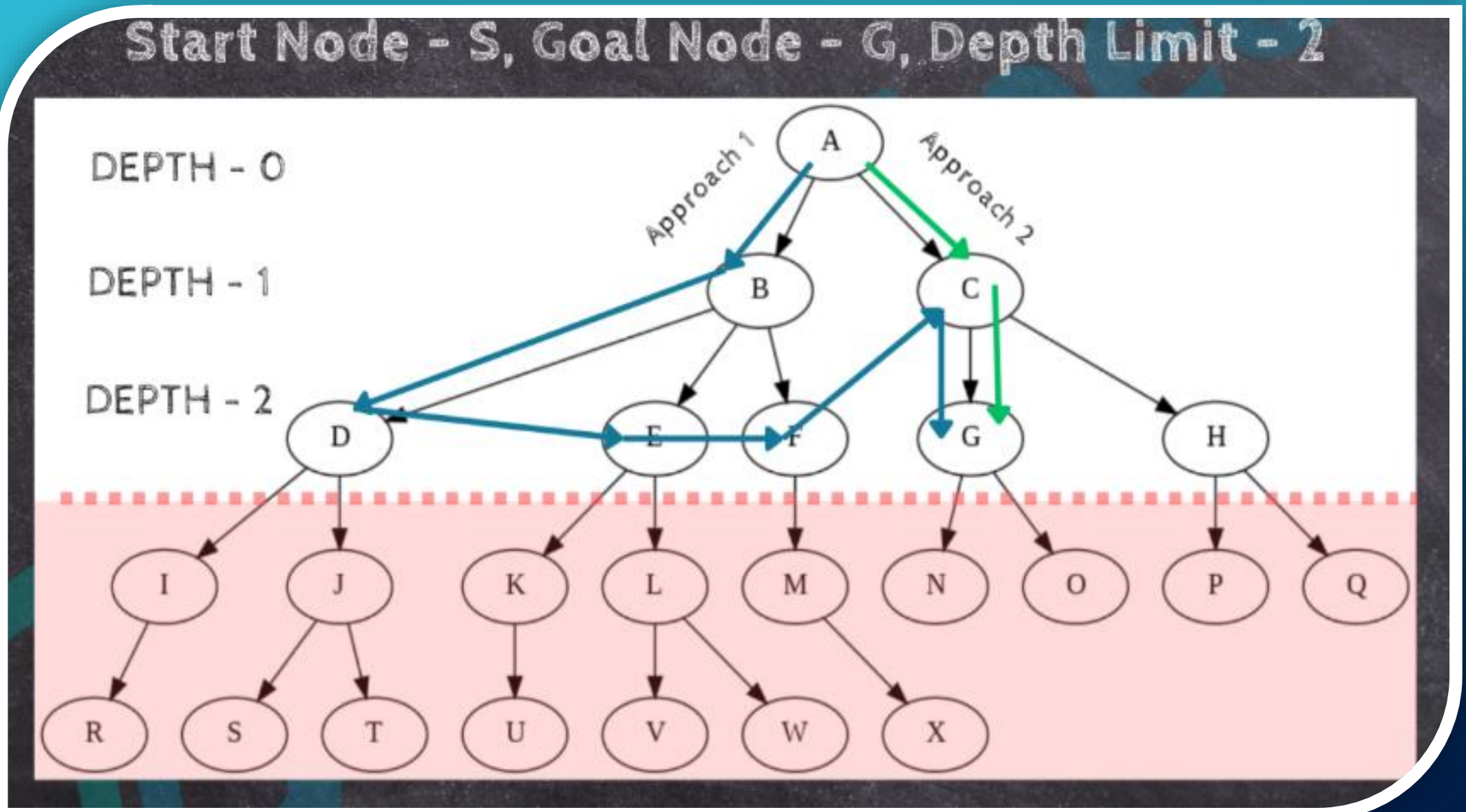
Level 2

Level 3

Level 4



Solution:-


Start Node - S, Goal Node - G, Depth Limit - 2
DEPTH - 0
DEPTH - 1
DEPTH - 2
Approach 1
Approach 2
A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
T
U
V
W
X

Advantages:-

- **Memory Efficient**: Stores only the current path up to the depth limit.
- **Avoids Infinite Loops**: Prevents DFS from going down endless paths.

Disadvantages:-

- **Incomplete**: May miss solutions beyond the depth limit.

- **Not Optimal**: Might return a longer or costlier path even if a better one exists.

Depth-Limited Search

Completeness: DLS search algorithm is complete if the solution is above the depth-limit.

Time Complexity: Time complexity of DLS algorithm is **$O(b^{\ell})$**.

Space Complexity: Space complexity of DLS algorithm is **$O(b \times \ell)$**.

Optimal: Depth-limited search can be viewed as a special case of DFS, and it is also not optimal even if $\ell > d$.

Iterative Deepening Depth-First Search:

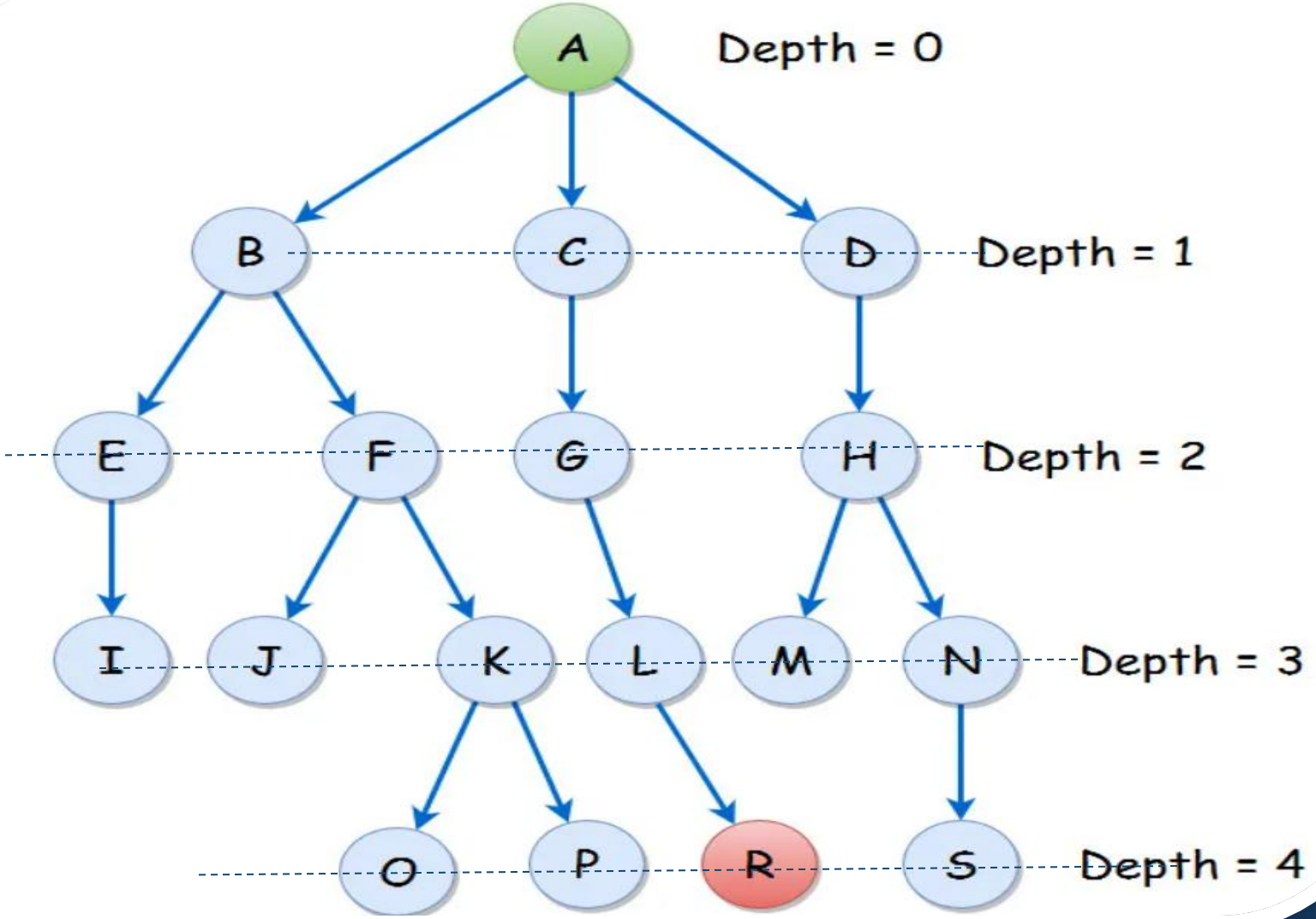
- *The iterative deepening algorithm is a combination of DFS and BFS algorithms. This search algorithm finds out the best depth limit and does it by gradually increasing the limit until a goal is found.*

- *This algorithm performs depth-first search up to a certain "depth limit", and it keeps increasing the depth limit after each iteration until the goal node is found.*

- *This search algorithm combines the benefits of Breadth-first search's fast search and Depth-first search's memory efficiency.*

- *The iterative search algorithm is useful in uninformed search when the search space is large, and the depth of the goal node is unknown.*

Example:-



Iteration's:-

A

A B C D

A B E F C G D H

A B E I F J K C G L D H M N

A B E I F J K O P C G L R D H M N S

Solution:-

DEPTH	DLS traversal
0	A
1	ABCD
2	ABEFCGDH
3	ABEIFJKCGLDHMN
4	ABEIFJKOPCGLRDHMNS

Advantages:-

- *Combine the advantage of both DFS and BFS*
- *It is Complete & Optimal*
- *Will not go in infinite loop*

Disadvantages:-

- *Regressive Recursion is required*
- *May require more memory*

IDDFS(Properties):

- **Completeness:**

This algorithm is complete if the branching factor is finite.

- **Time Complexity:**

Let's suppose $b$is the branching factor and depth is d, then the worst-case time complexity is $O(b^d)$.

- **Space Complexity:**

The space complexity of IDDFS will be $O(bd)$.

- **Optimal:**

IDDFS algorithm is optimal if path cost is a non-decreasing function of the depth of the node.

Thank you

SOURCE: - GEEKS FOR GEEKS
LINKEDIN POSTS