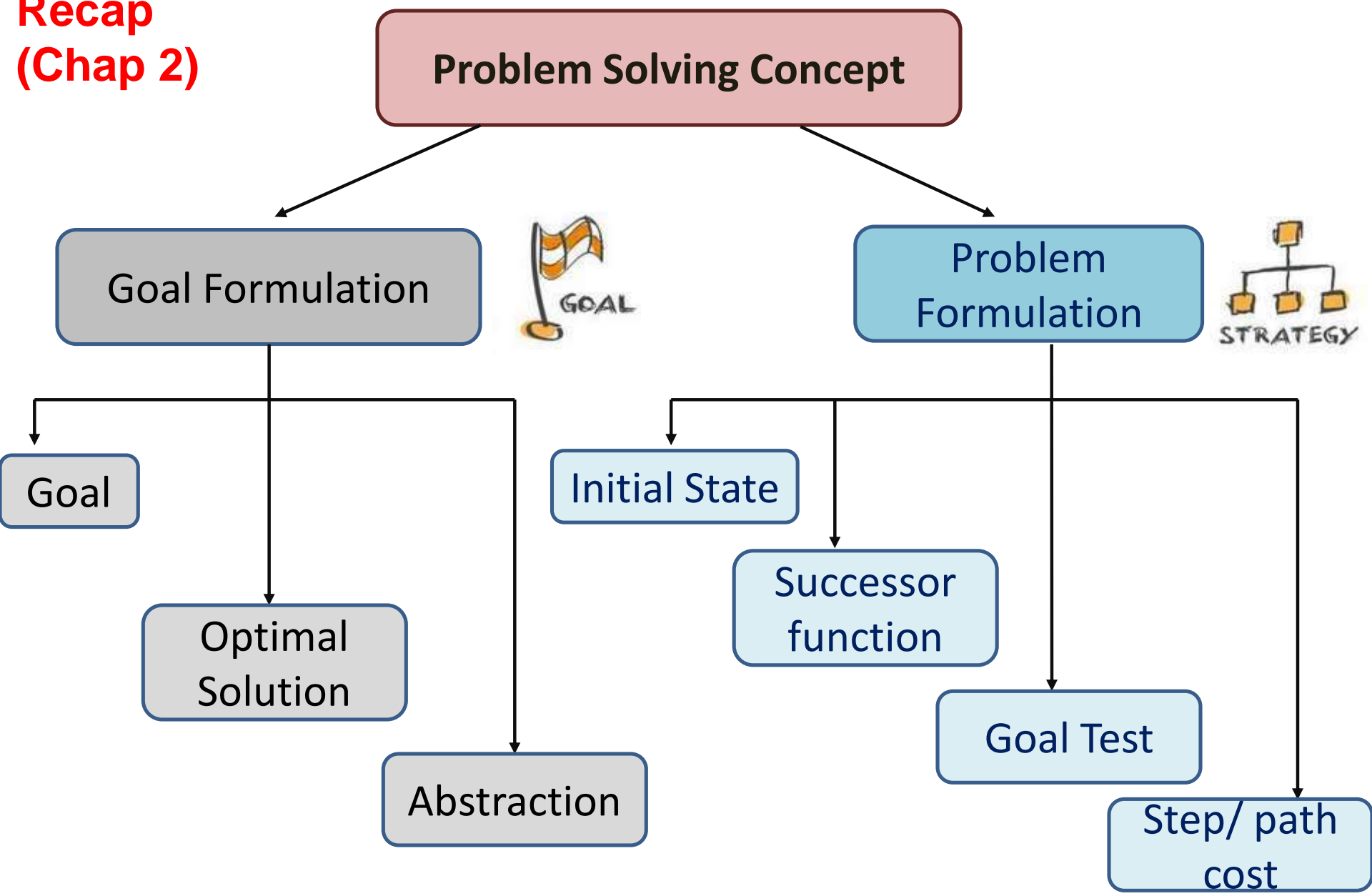


Artificial Intelligence

Chapter 3 Uninformed Search

Recap
(Chap 2)

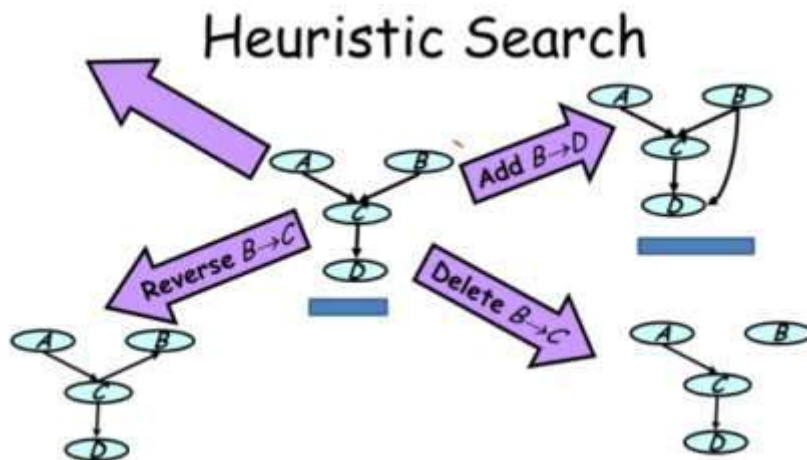


Search Strategies

- Uninformed search / blind search



- Informed search / heuristic search





Uninformed Search

- Breadth-first search
- Depth-first search
- Depth-limited search
- Iterative deepening depth-first search
- Bidirectional search



Breadth-first search

- All the nodes are expanded at a given depth in the search tree before any nodes at the next level are expanded.

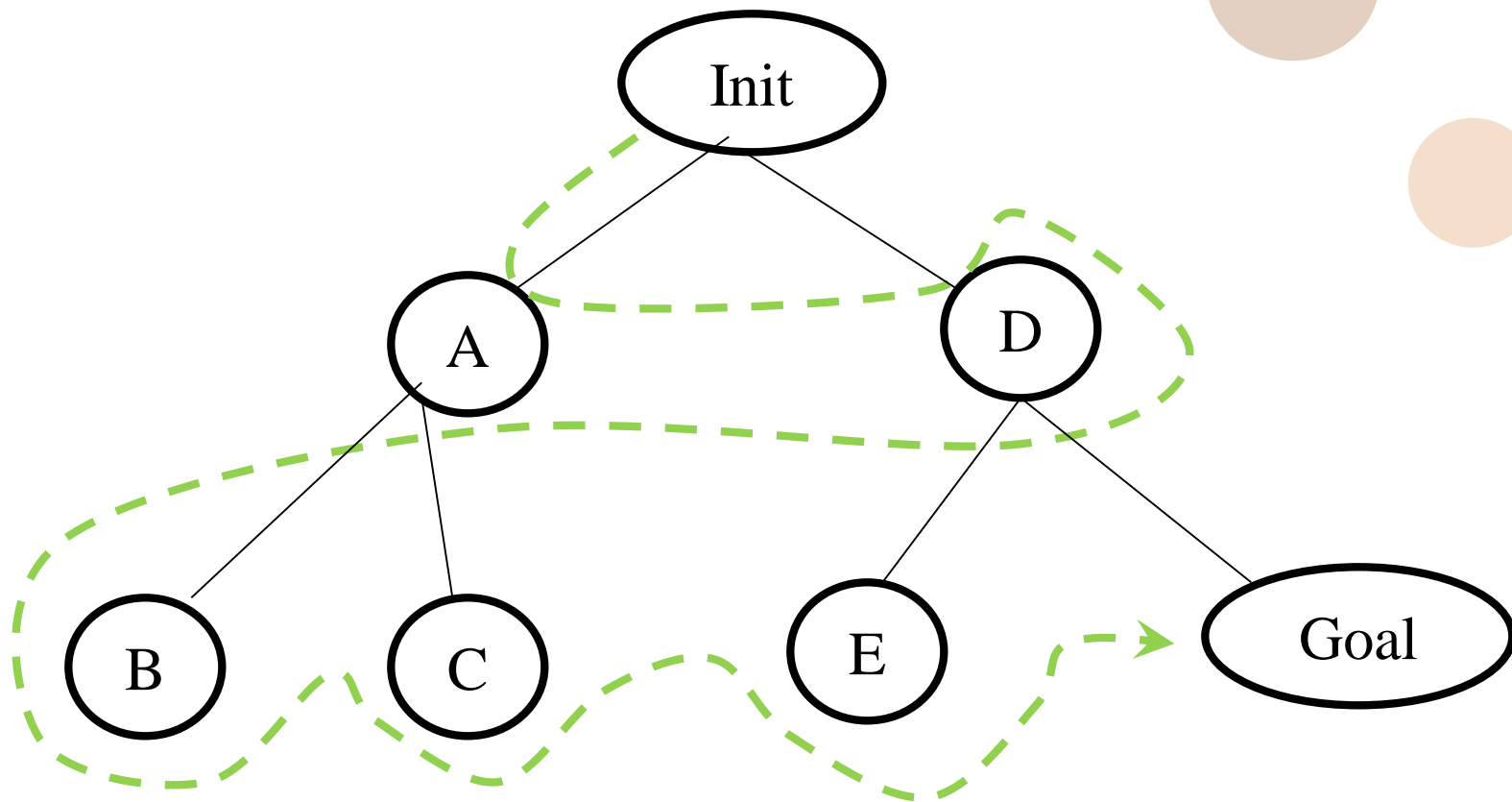


Algorithm

1. Begin with the root node.
2. Examine all of the nodes in each level before moving on to the next level.
3. Repeat the process and work downward from left to right until a solution is found.
4. Return Fail if there is no solution.



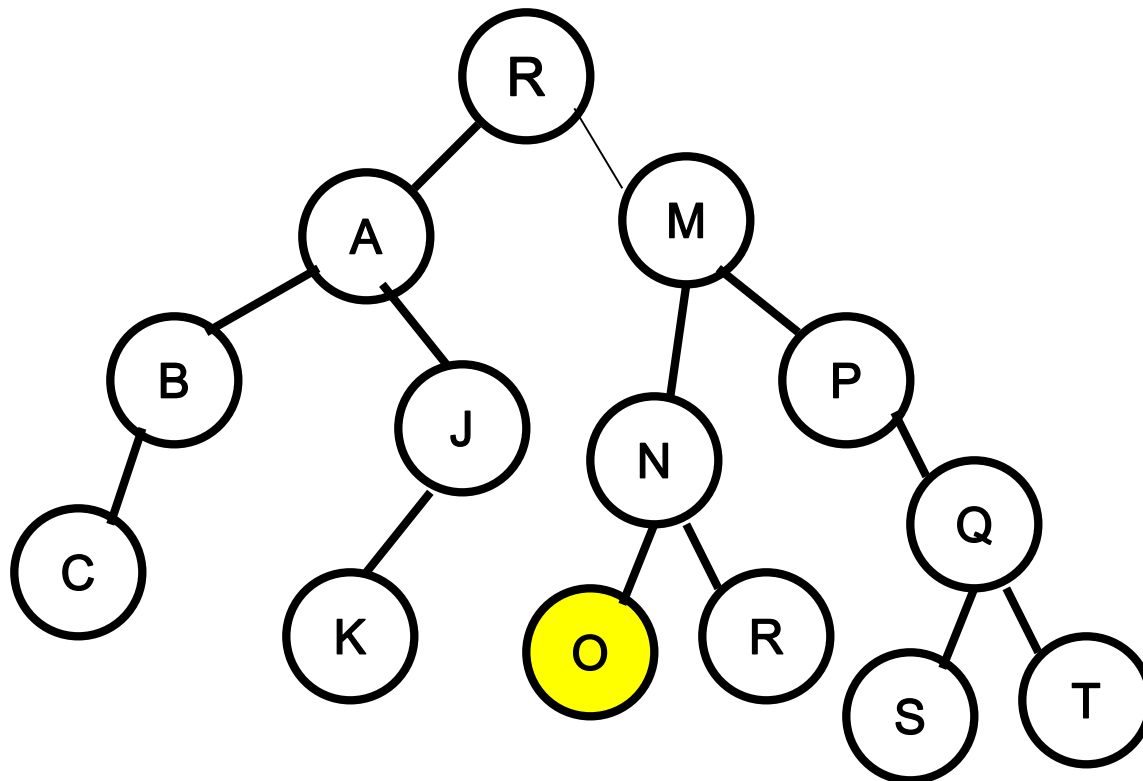
Breadth First Search





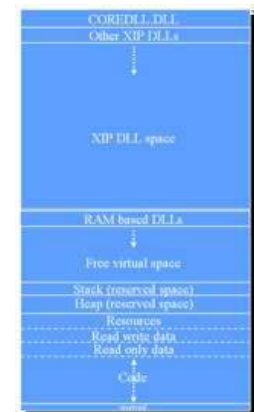
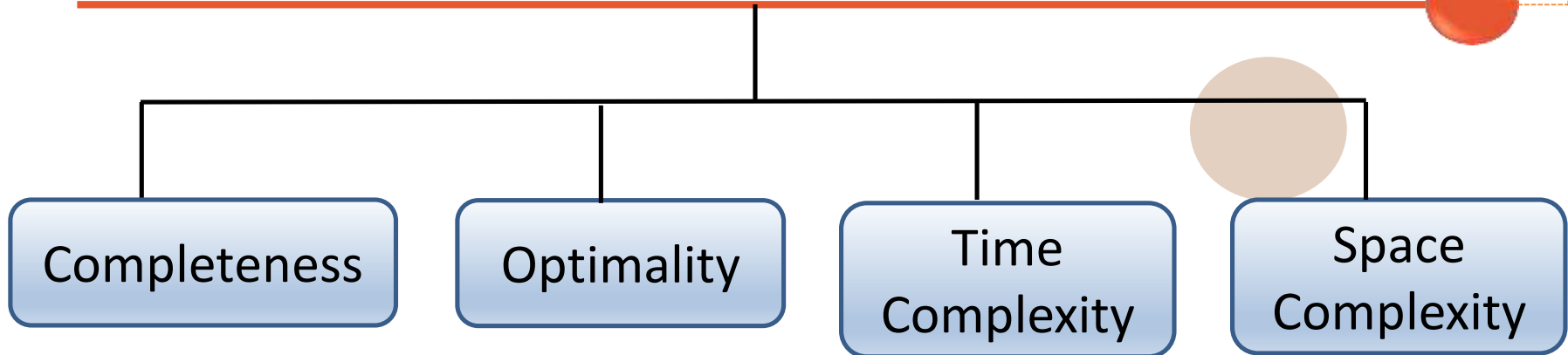
Breadth First Search

- Can you tell me the sequence of the states if using breadth-first search?





Measuring problem-solving performance





Advantages

Completeness

- Guarantees a solution.

Optimality

- Will always find the shortest path between initial state and goal state, with the least number of steps.



Limitations

Time Complexity

- Need longer time to get the goal if the goal state is on the far deeper level.

Space Complexity

- Need to expand all the branches before proceed to next level.
- Memory consumption is high

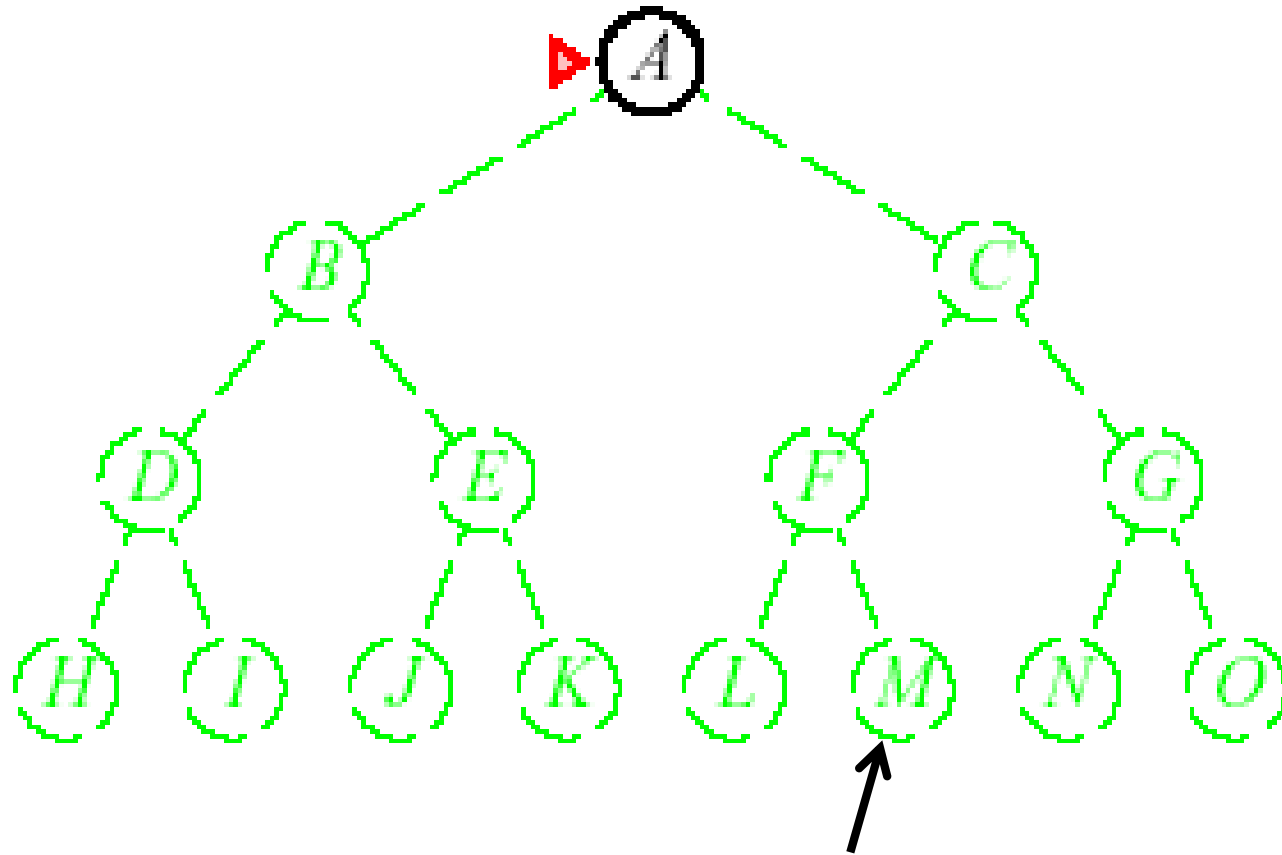


Depth-First Search

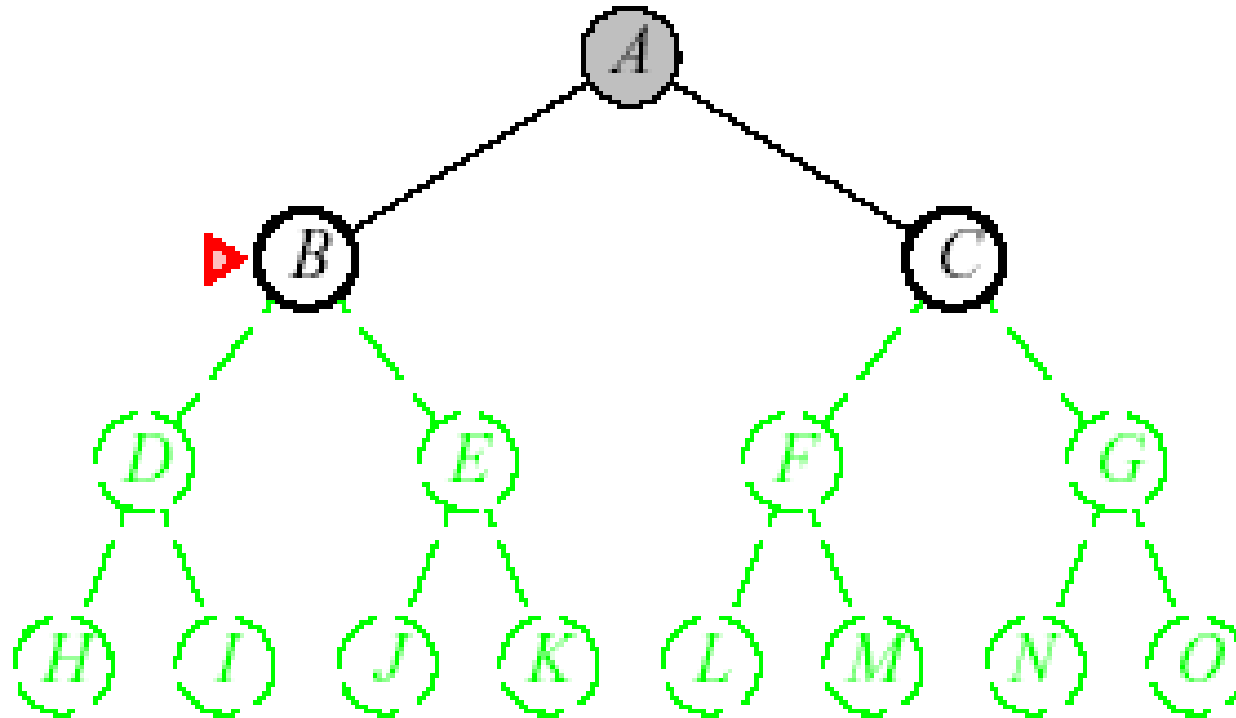
- Always expands the deepest node in the current fringe of the search tree.
- Nodes leading from dead end will be discarded from memory.
- Then the search will back up to the next shallowest node that still has unexplored successors.
- Implemented by calling tree search using **LIFO (last-in-first-out)** strategy.



Example

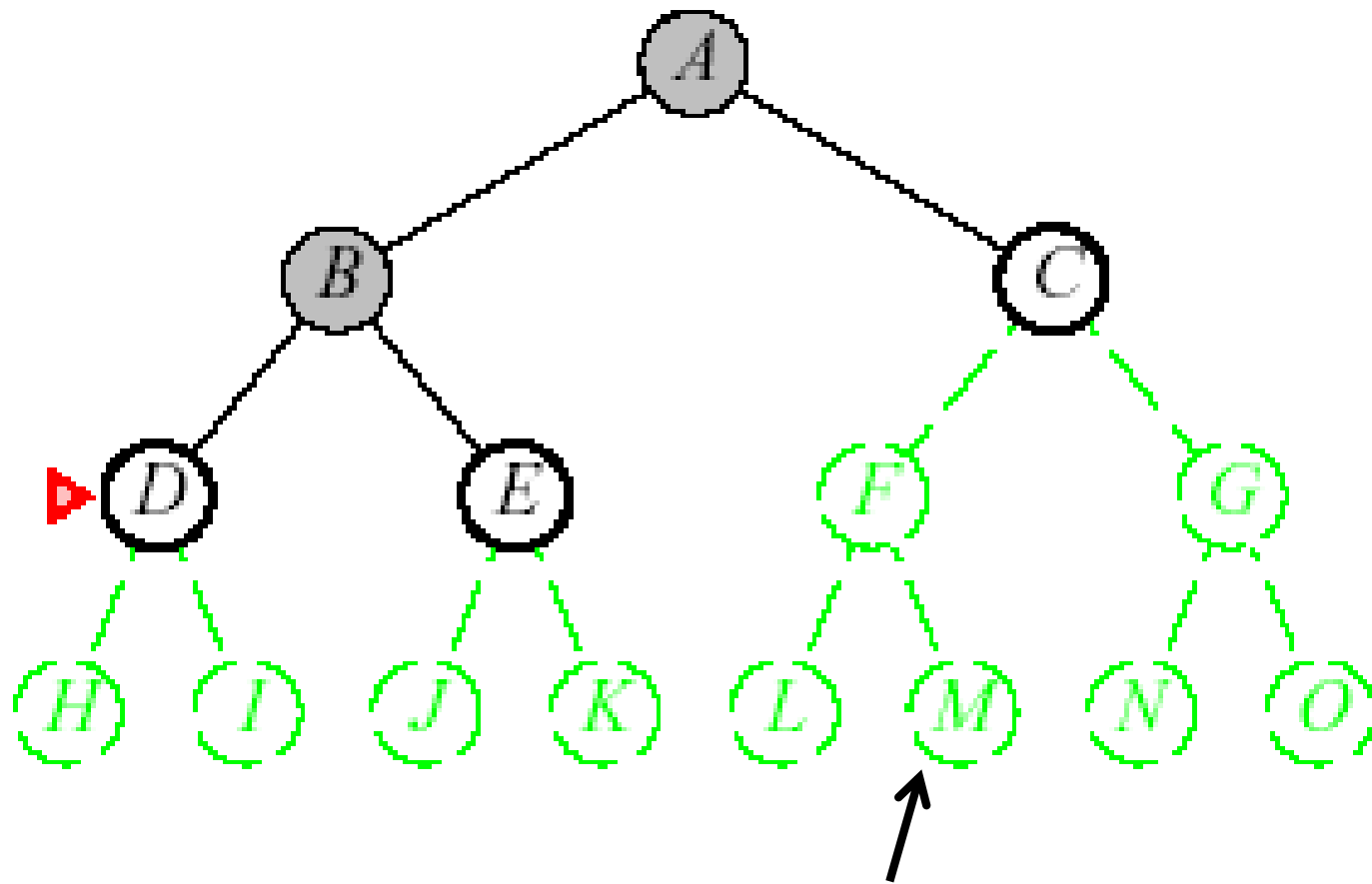


Continue.....



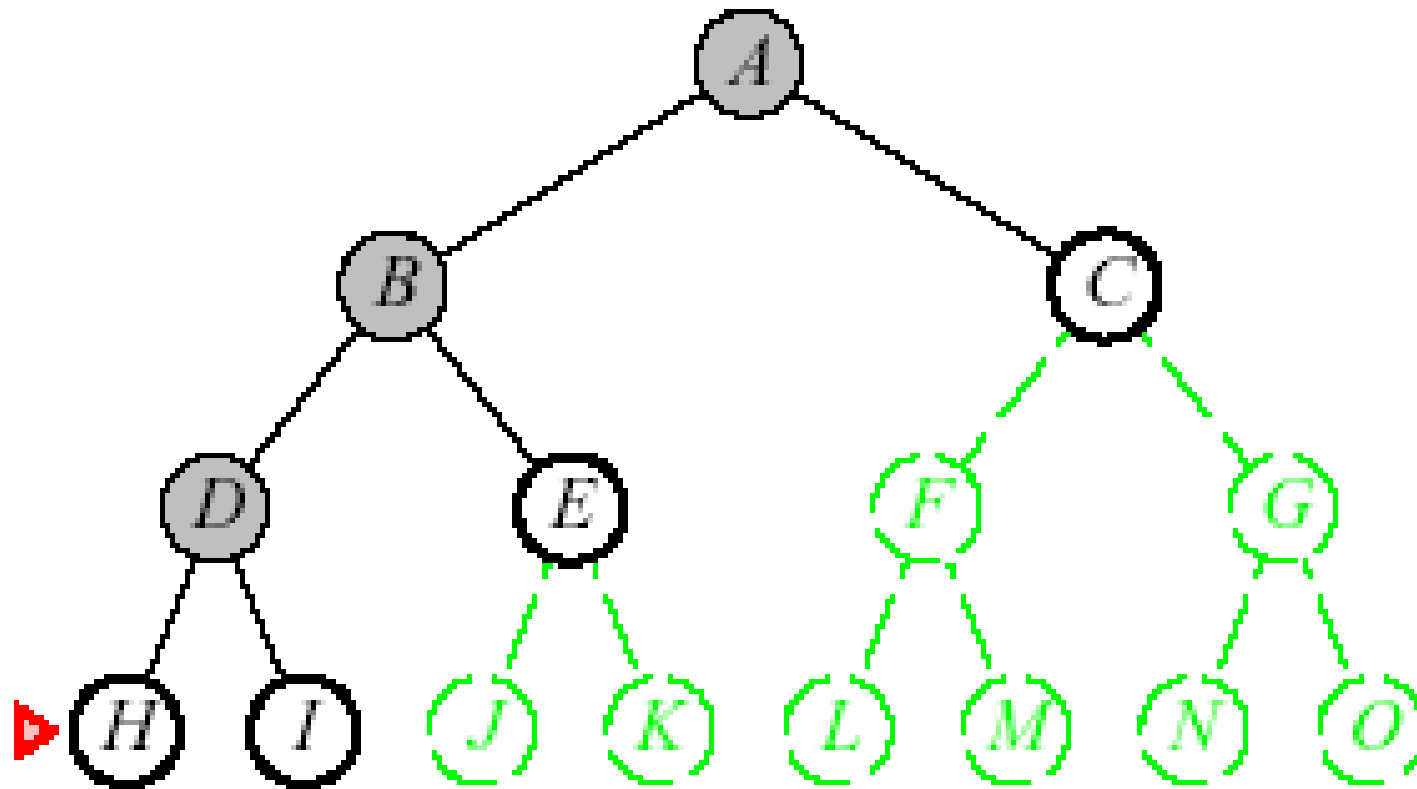


Continue.....



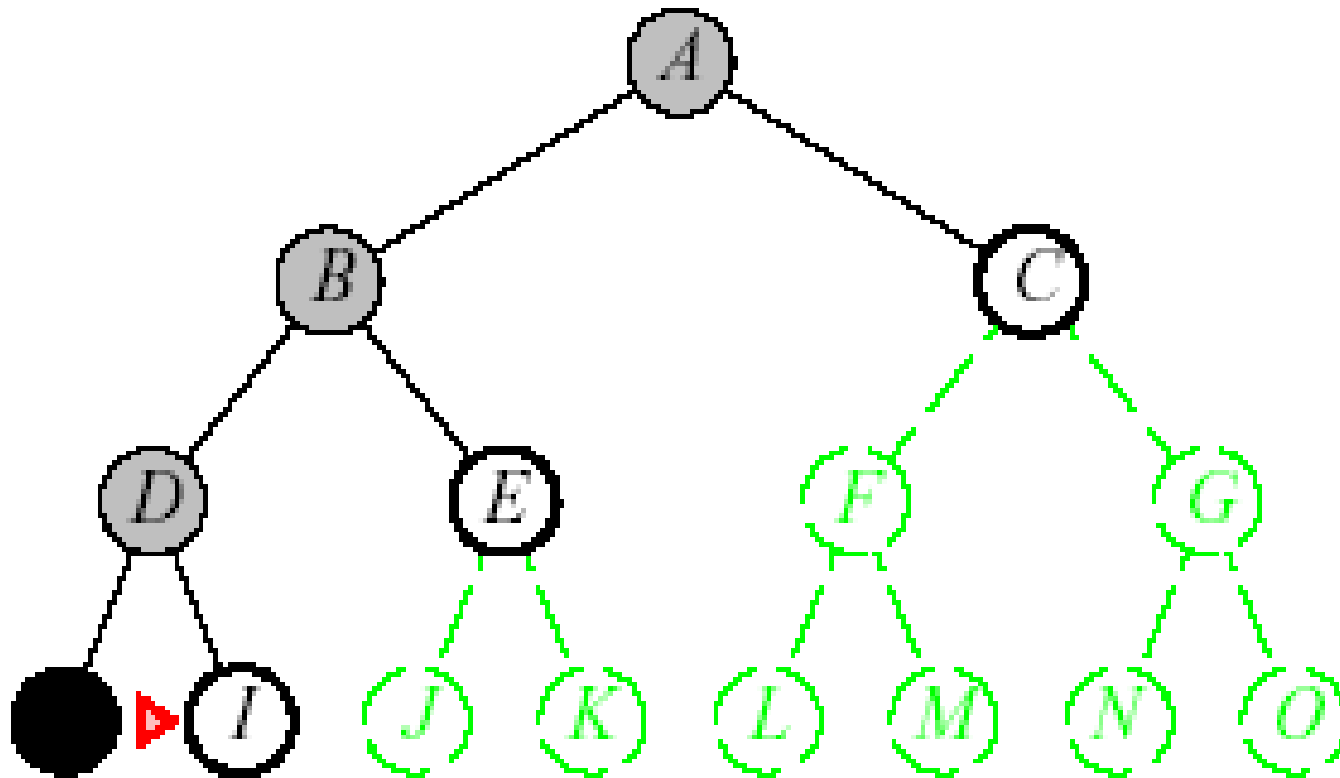


Continue.....



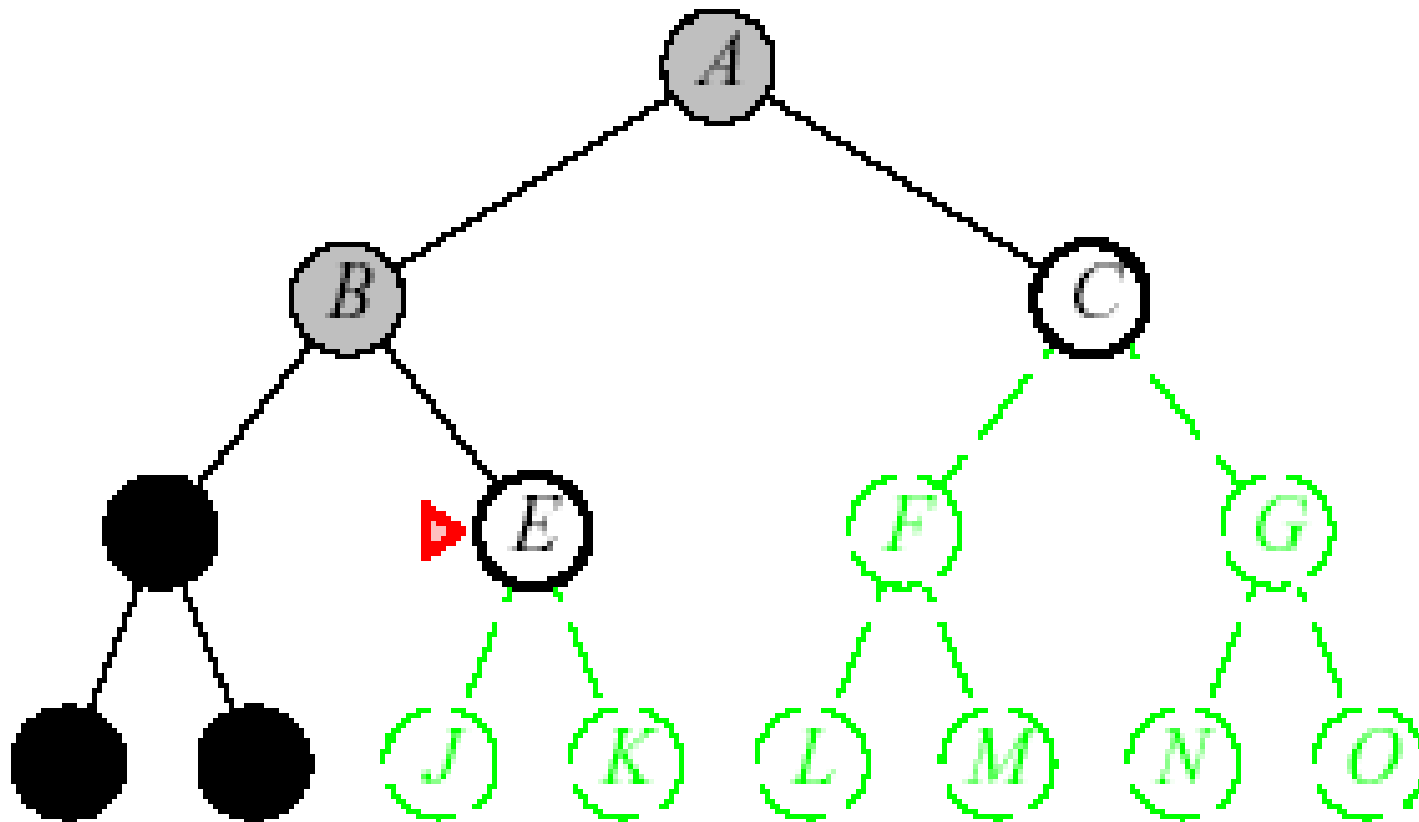


Continue.....



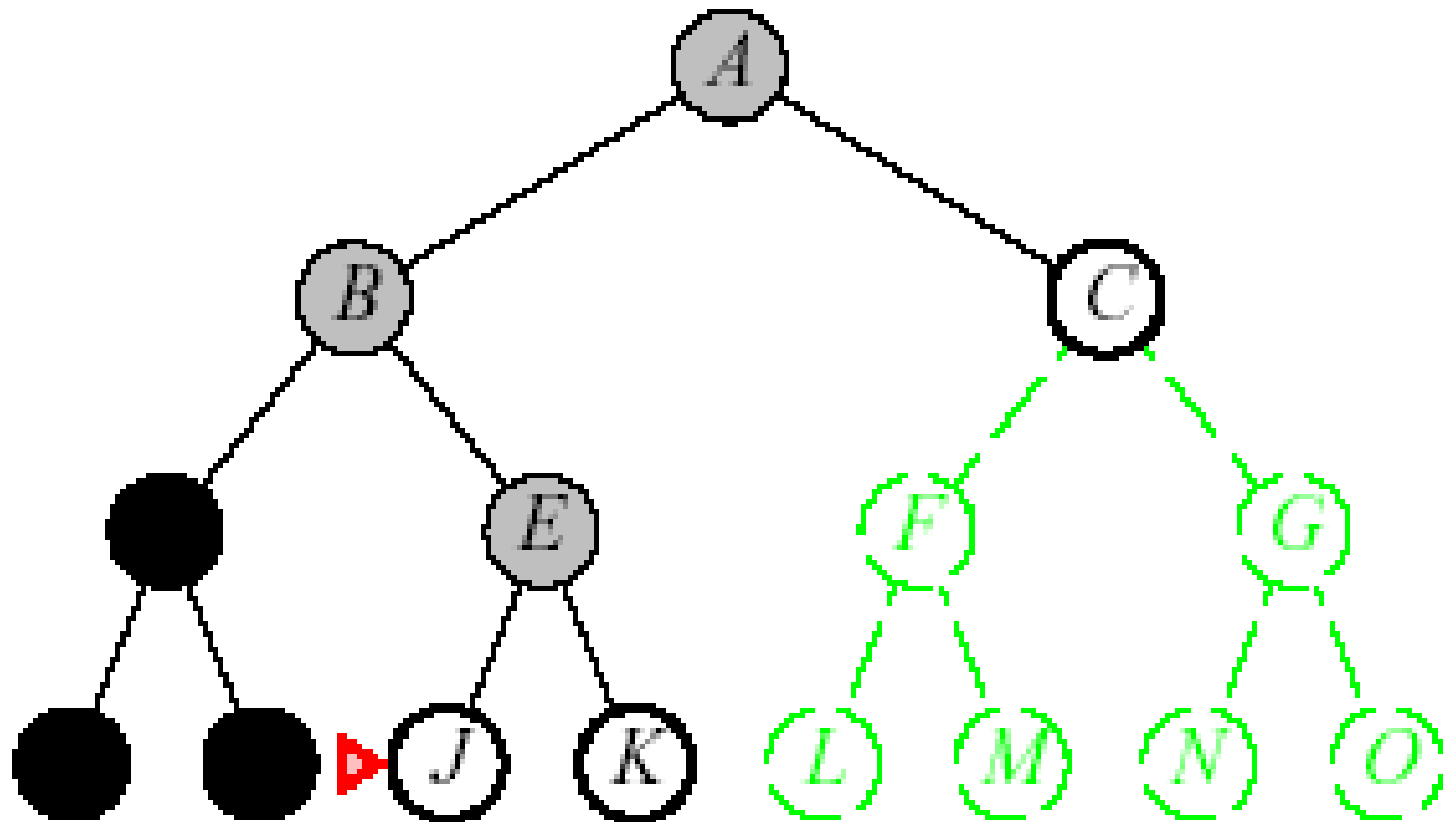


Continue.....



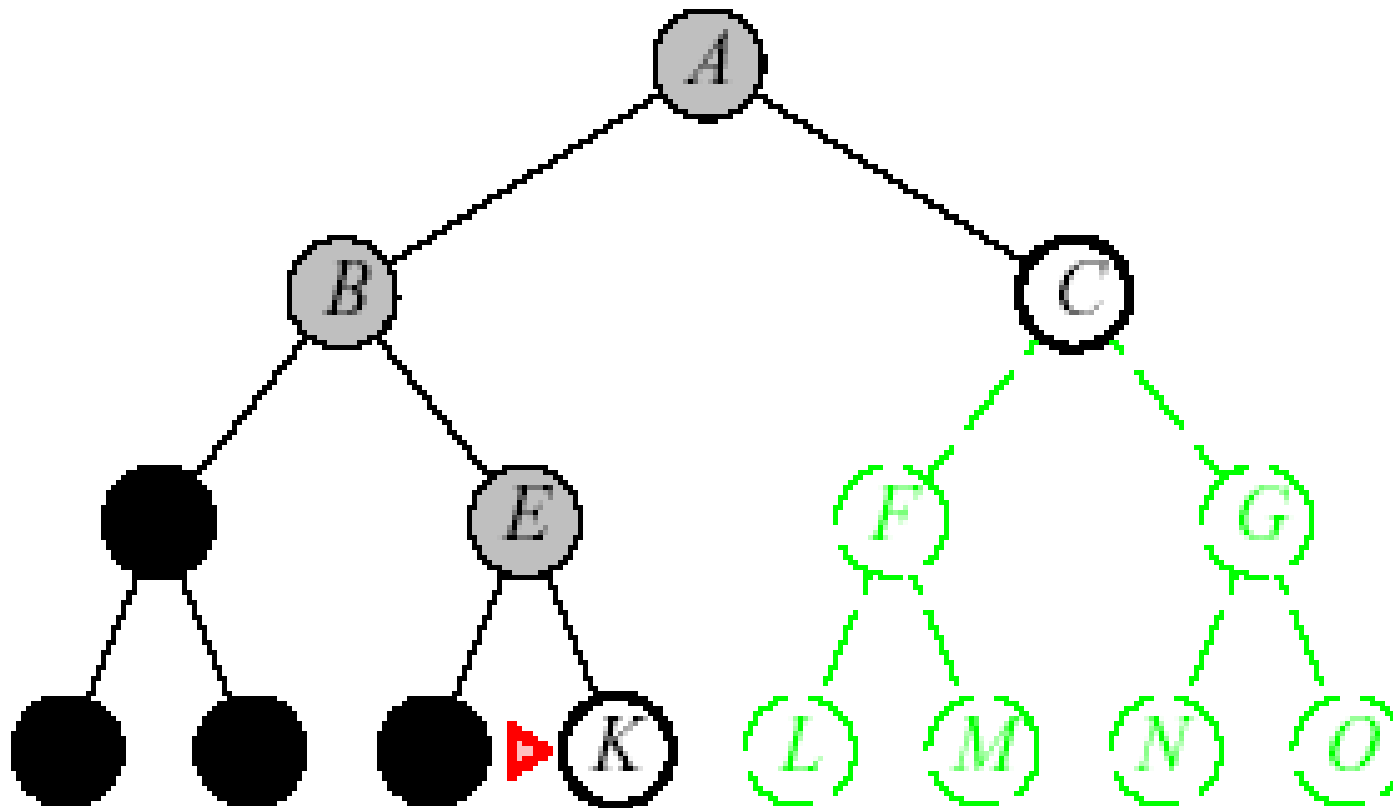


Continue.....



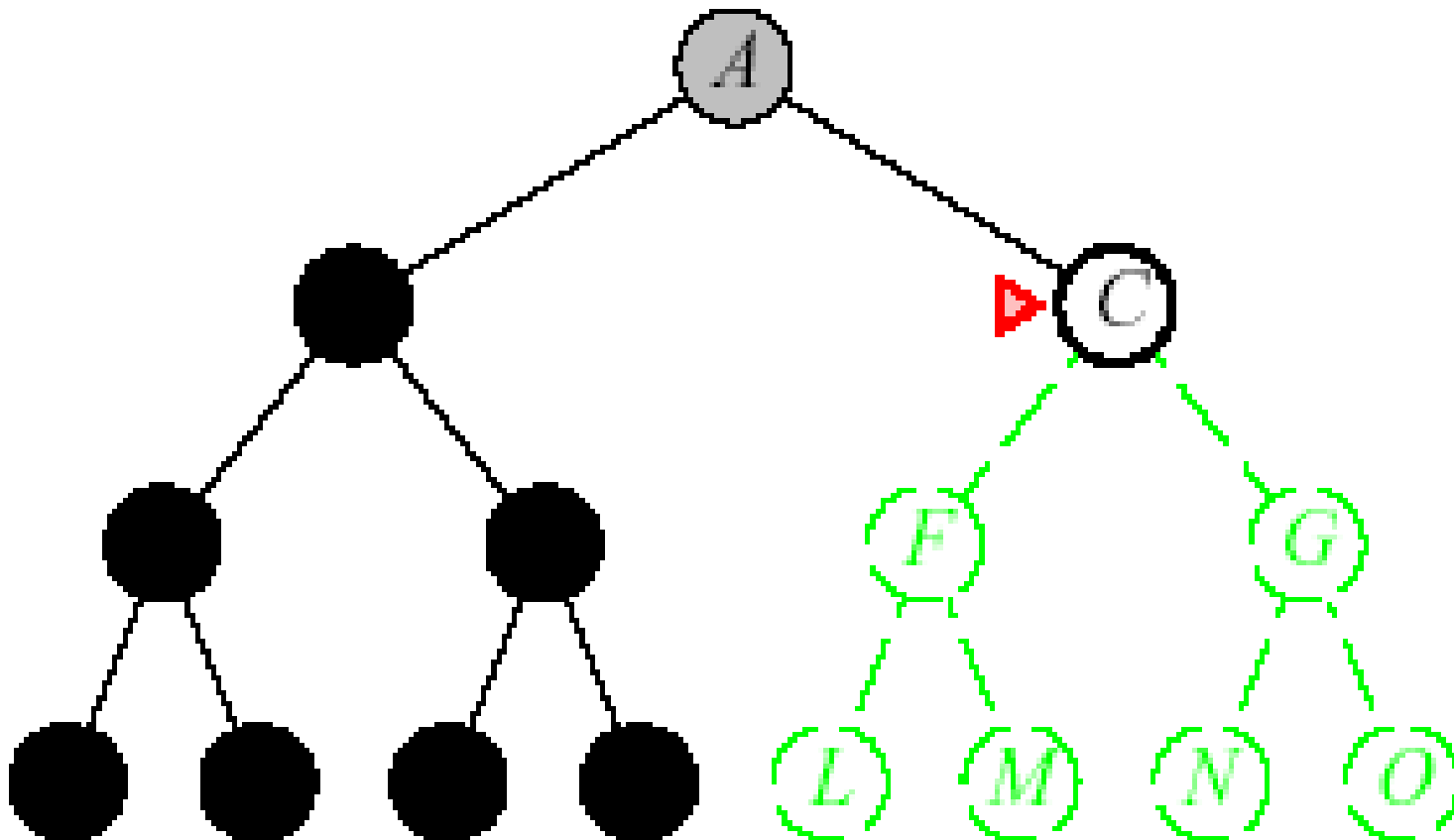


Continue.....



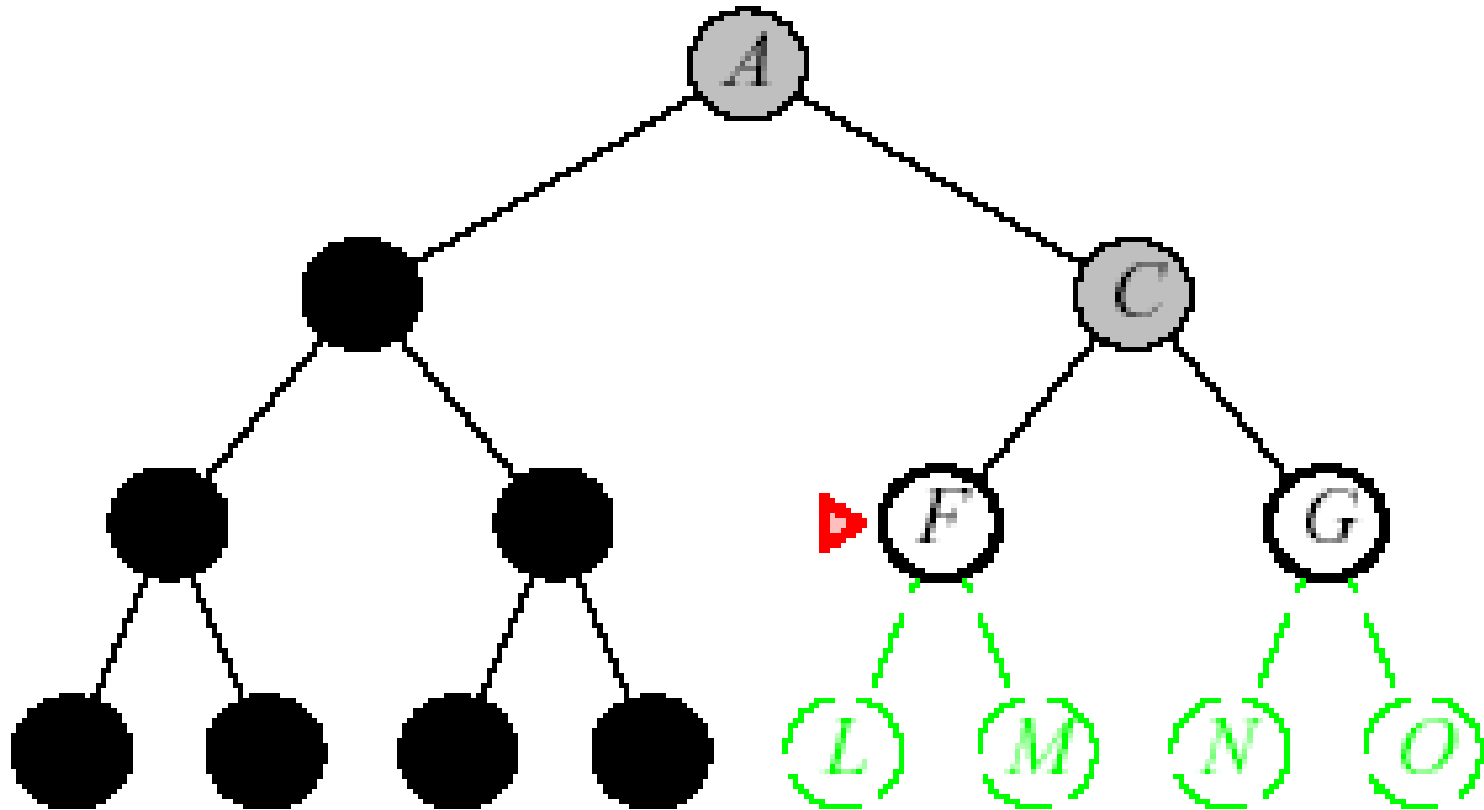


Continue.....



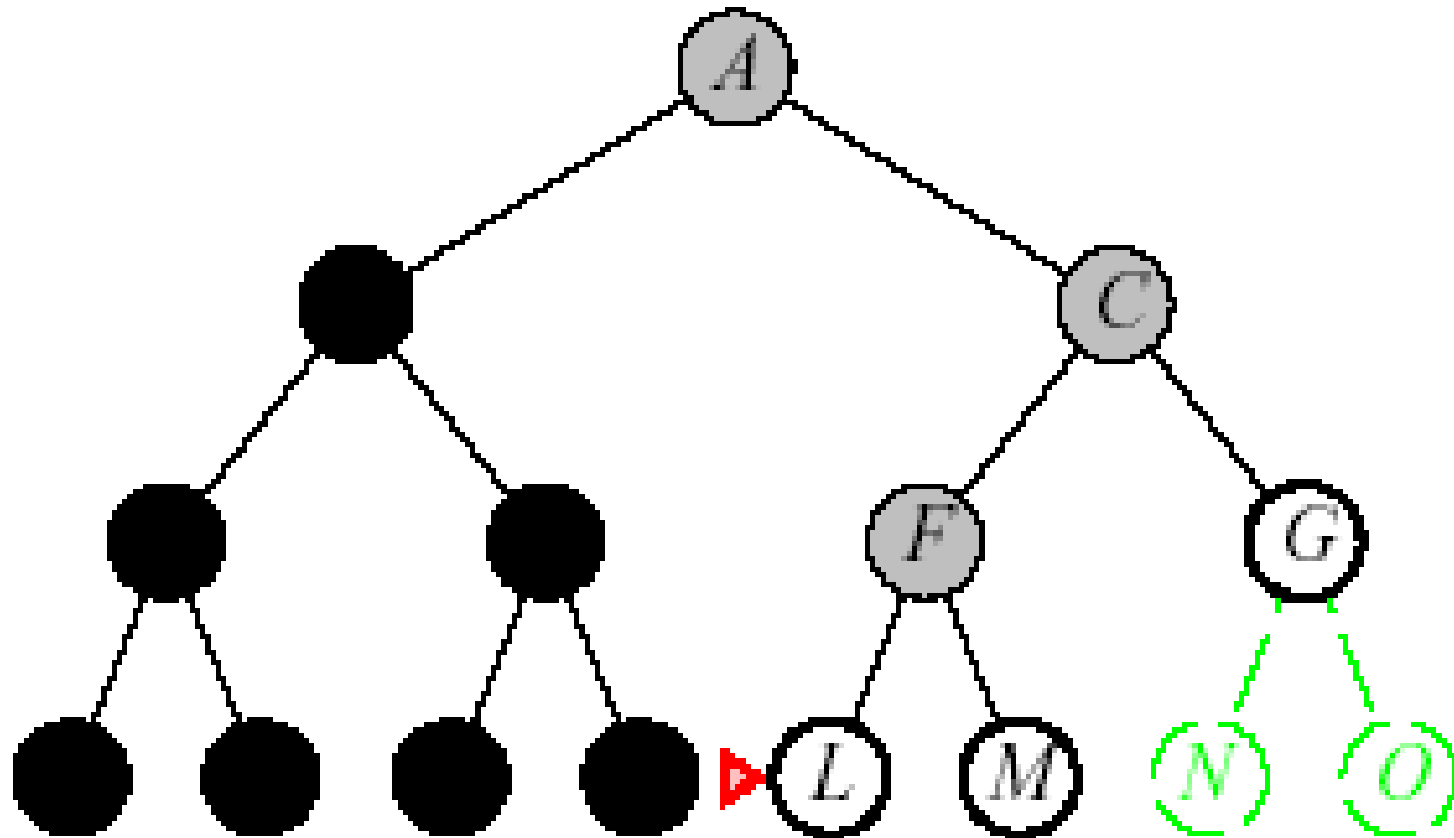


Continue.....



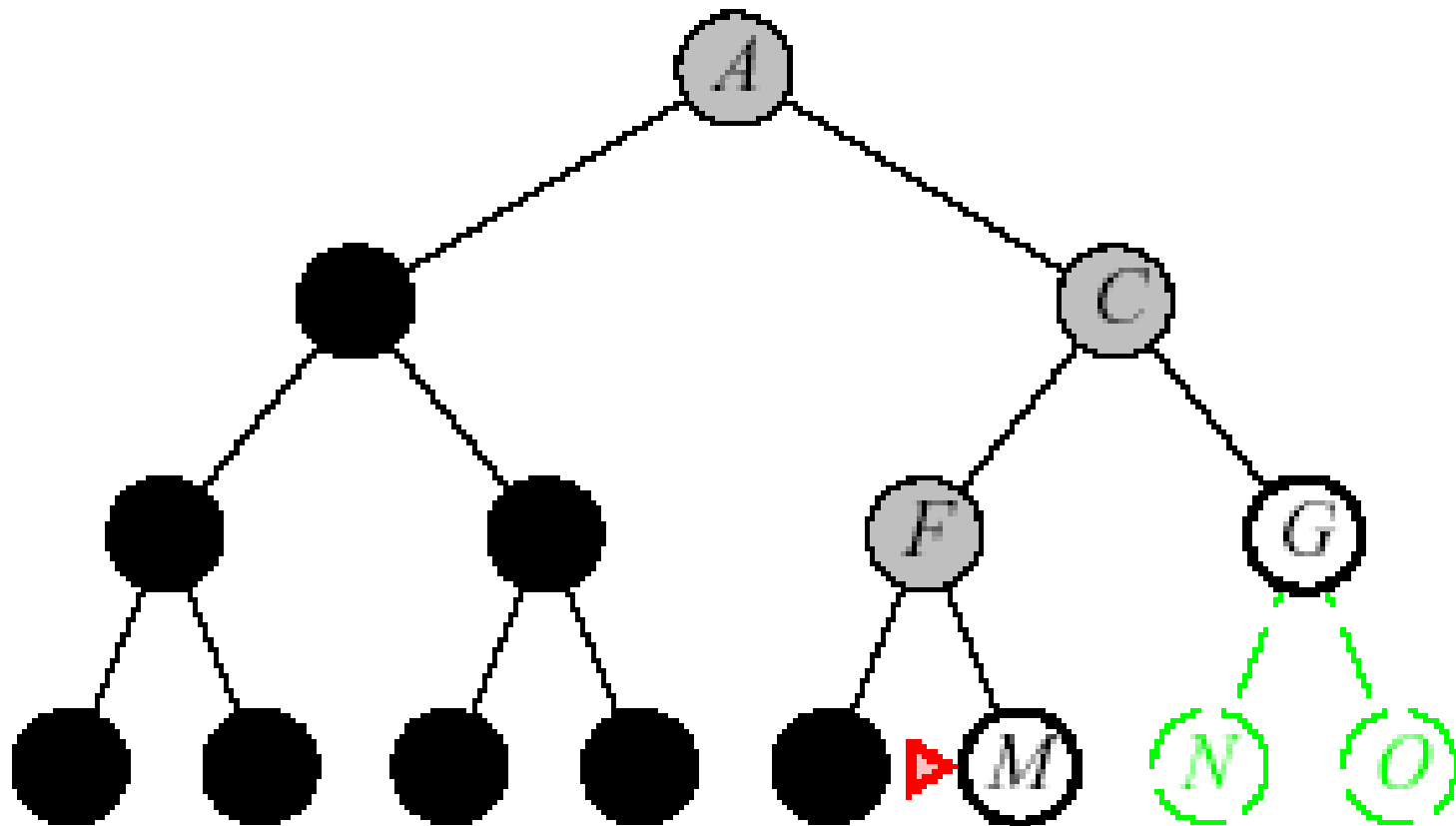


Continue.....





Continue.....





Evaluation

- Evaluate depth-first search strategy based on
 1. Completeness
 2. Optimality
 3. Time Complexity
 4. Space Complexity



Example 1



Figure 1

The 8-puzzle

Move tiles LEFT, RIGHT, UP, DOWN

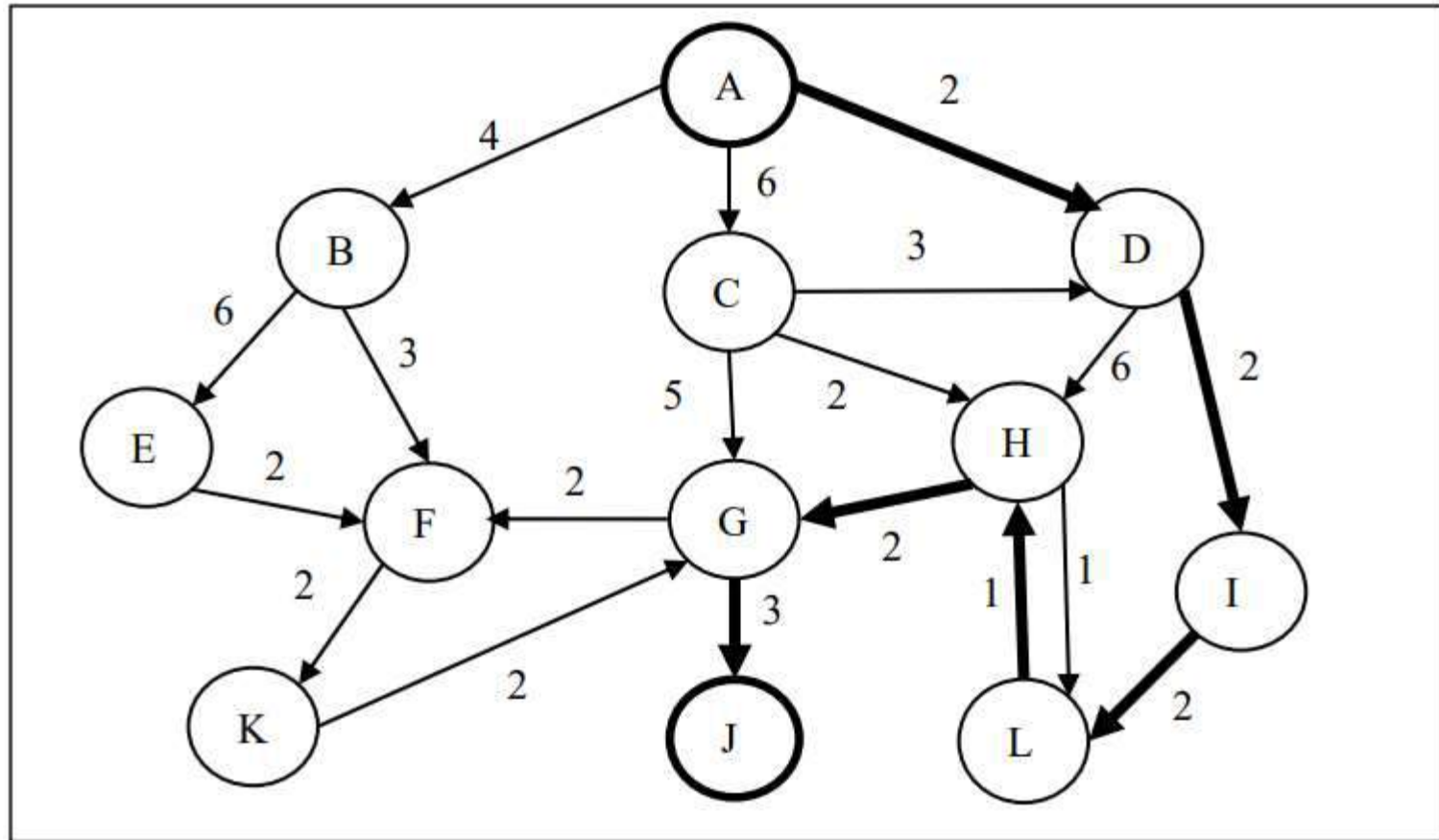


Figure 2



Example 2

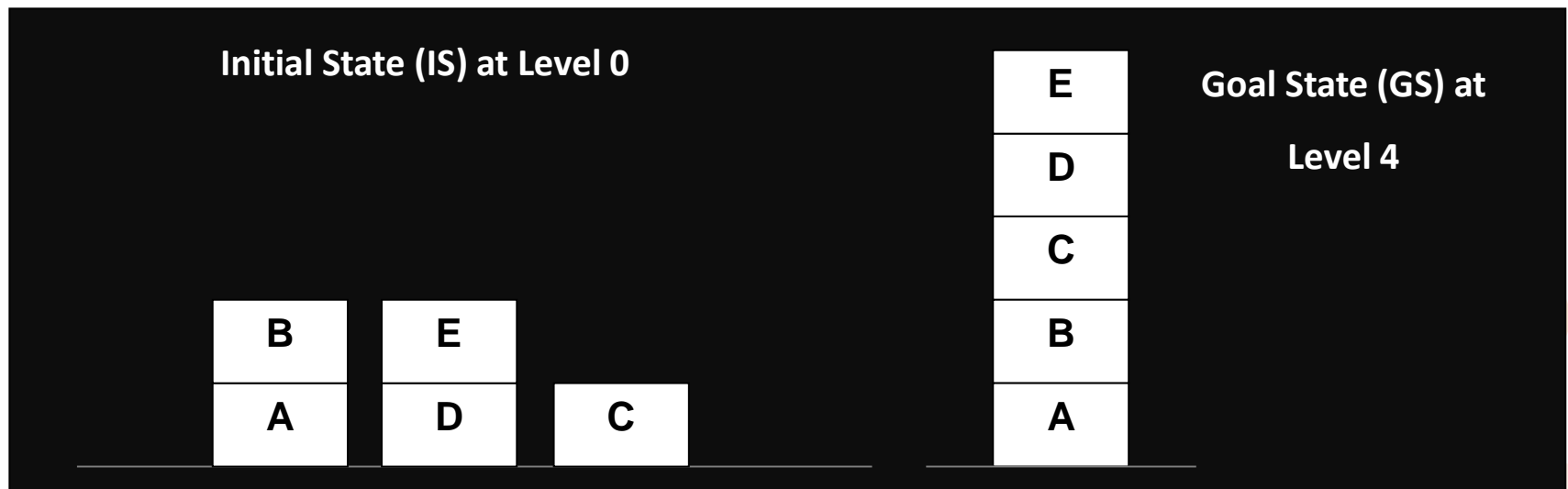
Directed Graph





Example 3

Blocks World Problem



Only one block can be moved at one time



Explain the Differences

BFS

VS

BFS










Depth-Limited Search

- Depth-first search can lead to **infinite path**.
- Solution – Limit the depth of the expansion to avoid uncontrolled infinite path.
- Problem – how to determine the depth limit, l .

Iterative Deepening Depth-First Search (IDS)



- In depth-first search, there is no reliable method to decide a depth limit to include at least one solution.
 - Iterative deepening can solve this by checking the entire tree by gradually increasing the depth limit – first depth 0, then depth 1, then depth 2 and so on.
- 
- 
- 
- 
- 



Bidirectional Search

- Run two simultaneous searches:
 1. One forward from the initial state
 2. One backward from the goal (final state)
- Stop searching when the two searches meet in the middle.

Heuristic search

Next Lecture