

★ What is Problem Formulation?

problem formulation means clearly defining the problem so a computer (or AI agent) can solve it.

★ 5 key parts of Problem Formulation:

- i) Initial State start position
- ii) Action moves (Left, Right, up down etc)
- iii) Transition Model what happens after each action
- iv) Goal state exit point
- v) Path Cost Each step cost

if your search space is:

Too big → the AI will be slow
Too small → the best answer may be missed

★ What is search space

The search space is the complete set of possible solutions that AI can explore to solve the problem.

Guess a 3-letter word (All possible combinations of AAA, AAB ... ZZZ)

$$[26 \times 26 \times 26 = 17,576 \text{ possibilities}]$$

Uninformed vs Informed Search

Uninformed Search

- Searching without Information
- No knowledge
- Time Consuming
- More Complexity (Time, Space)
- DFS, BFS etc.

Informed Search

- searching with information
- Use knowledge to find steps to approach the ultimate goal

Quick solution

Less complexity

A*, Heuristic DFS, Best First Search

Information = Heuristic

NP \rightarrow Hard Problem

Travelling Salesperson Problem

$(n-1)!$

یہ تکنیکیں Buffer
 کے ساتھ ہوتی ہیں۔

Brute Force Technique

Blind Technique

Uninformed Searching

- Lacks any kind of domain specific

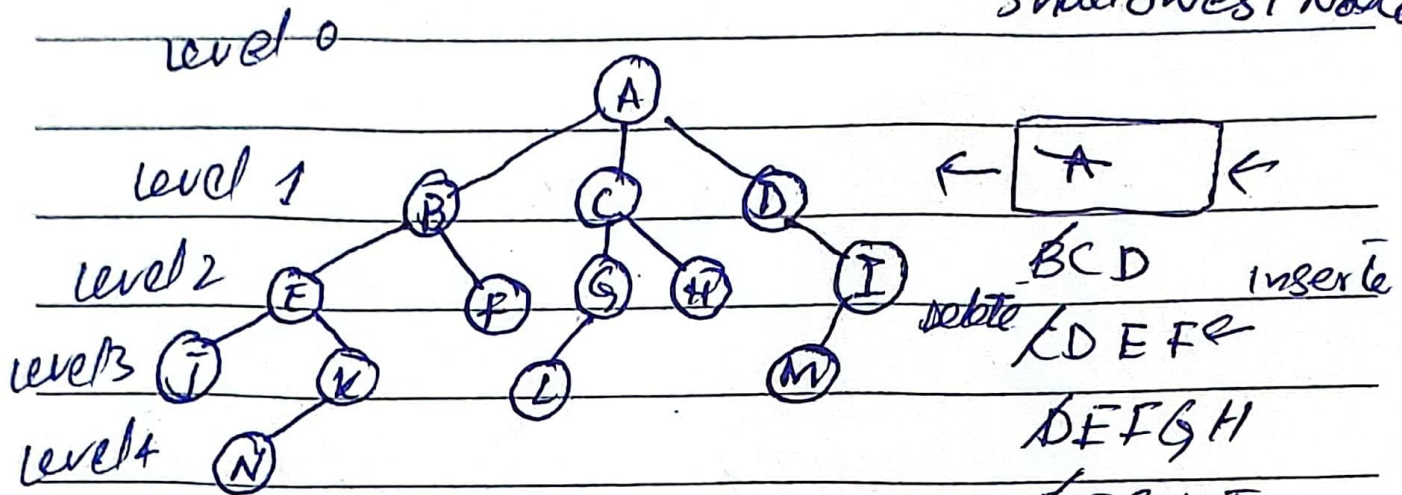
data structure

BFS

(Level Search Technique)

FIFO (Queue)

Shallowest node



Level by level
 movement

A B C D

E F G H

I J K L

M N

It is complete

that mean it

will provide you
 the answer.

Optimal & shortest Result

Time Complexity $O(V + E)$

$b =$ branch factor, $d =$ depth $O(b^d)$ in AI

$b=3$
 $d=2$
 $(3)^2 = 9$

Depth First Search

- Uninformed Searching

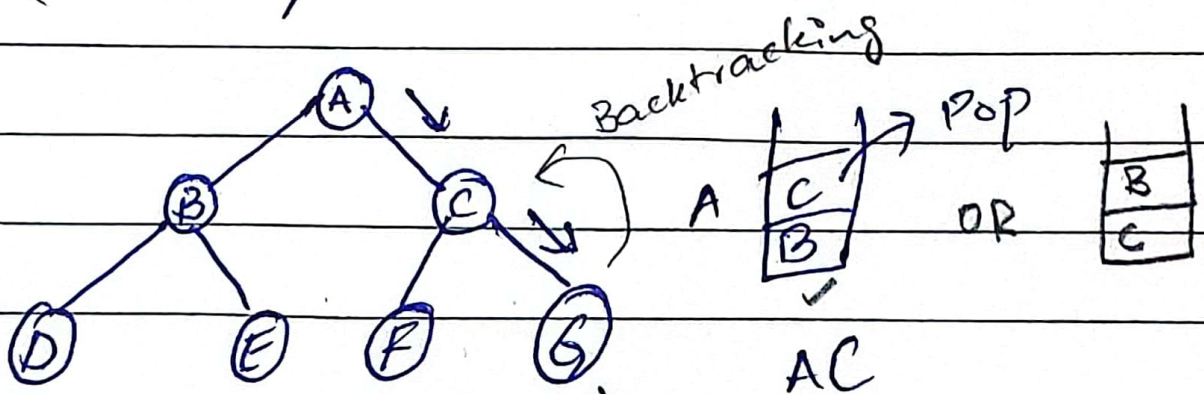
- Stack (LIFO)

- Deepest Node

result of DFS is - Incomplete

- Not Optimal

(Graph Cycles) - Time complexity $O(V+E)$
(Loop Trap)

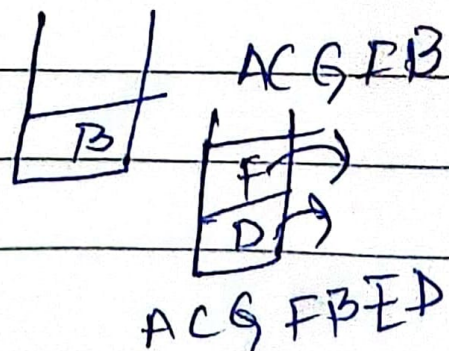
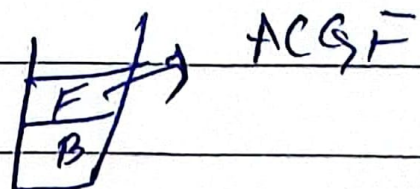


Time complexity

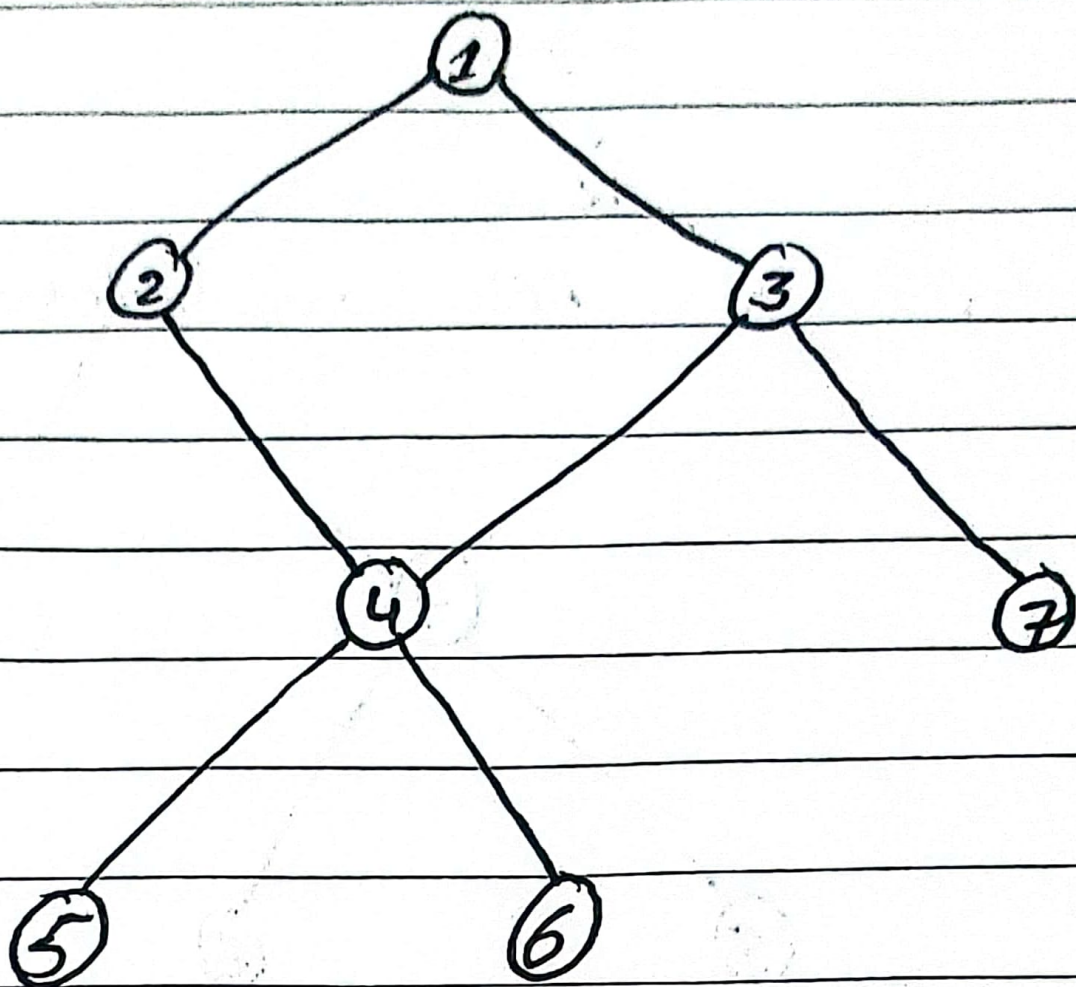
in AI

$$O(b^d)$$

or Branches ke jikar

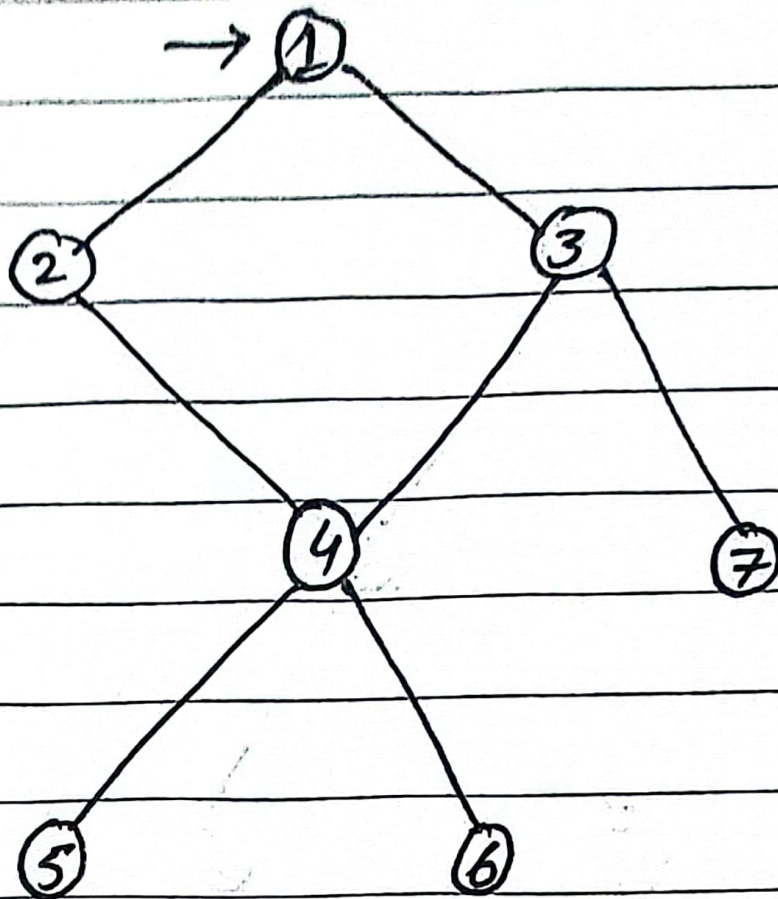


BFS



1 2 3 4 7 5 6

DFS



3
4 4 4
2
1

backtracking

1 2 4 5 6 3 7

$$O(V+E)$$

BFS

web crawler

Google Map

social network analysis

Maze solving

Stack Data Structure

Q. Why do we study so many data structures?

QA. Having so much knowledge of many data structures help us to solve different problems efficiently.

Goal:

Choice of DS \rightarrow Efficient Problem Solving

Stack

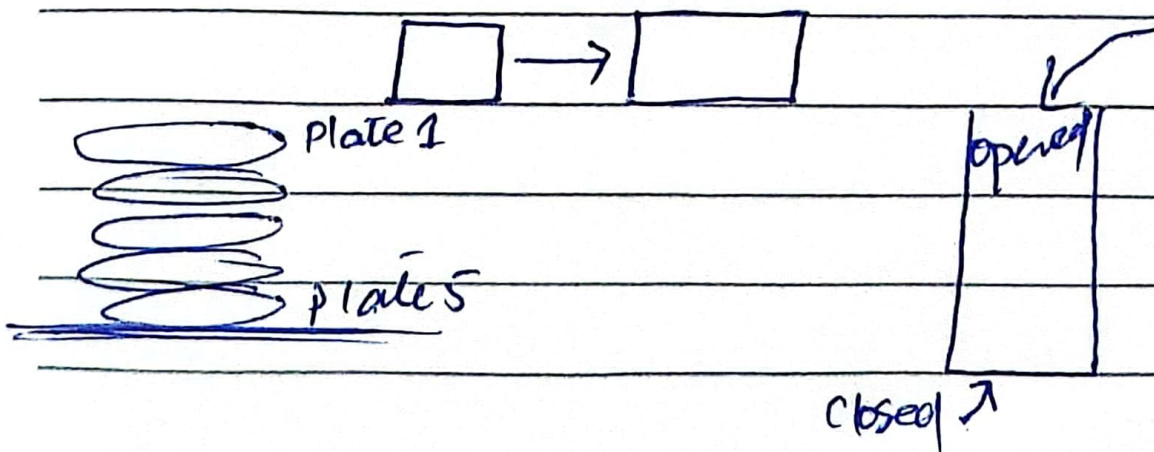
\rightarrow Linear data structure

e.g; i) stack of plates

ii) stack of books

iii) stack of chairs

iv) Pringle can



LIFO Principle

Tower of Hanoi

→ LIFO or FILO

→ Element which is put up last will be the first one out.

Abstract Data Type

Abstract data type is a type for objects whose behaviour is defined by set of operations.

→ Operations restriction

Industry Examples

→ Undo / Redo Operation

→ web browser history and back button

→ Expression Evaluation

↳ Prefix

↳ postfix

↳ Infix

search concepts

State space Search

$$S': \{ s, A, \text{Action}(s), \text{Result}(s, a), \text{cost}(s, a) \}$$