

Search techniques (2)

⑥ Uninformed (Blind)

- DFS, BFS
- Depth Limited Search
(Ext. DFS)
- Iterative Deepening
search (Ext. DFS)
- Uniform cost search
- Bi-directional

Def: use only the info. avail.
in the prob. def.

④ Informed (Heuristic)

- Hill Climbing Search
- A*
- Best first
- Greedy

Def: use info. avail.
in the prob. def.

and domain know-
ledge/heuristic
to choose the
best move

Optimization:

In which the search
finds an optimal value
of an objective function

Uninformed

- Search without info
- No knowledge about domain
- Time, Space complexity - exponential
NP (non-poly)
- ~~Always~~ Always gives optimal soln.

Informed

- search with info.
- use knowledge to find steps to solution
- Less complexity
 - [Polynomial]
- optimal or not.

heuristic
↑

Hill Climbing Algo:

Local Search Algo

~~not complete~~
(stops at local maxima, plateaus, ridges)

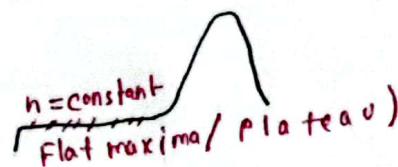
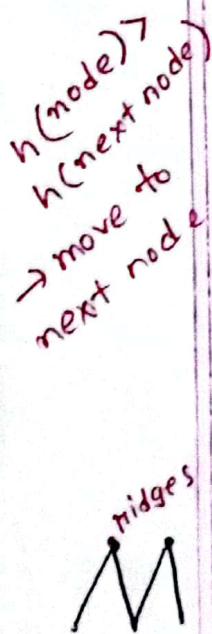
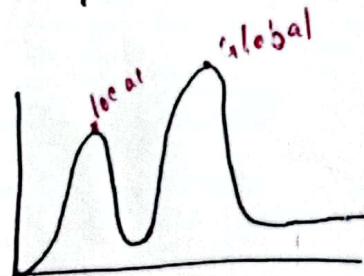
Greedy

No backtrace (not remember prev node on alt. path)

If next state is better, move to next.
otherwise stops. (in local maxima)

- not find global maxima

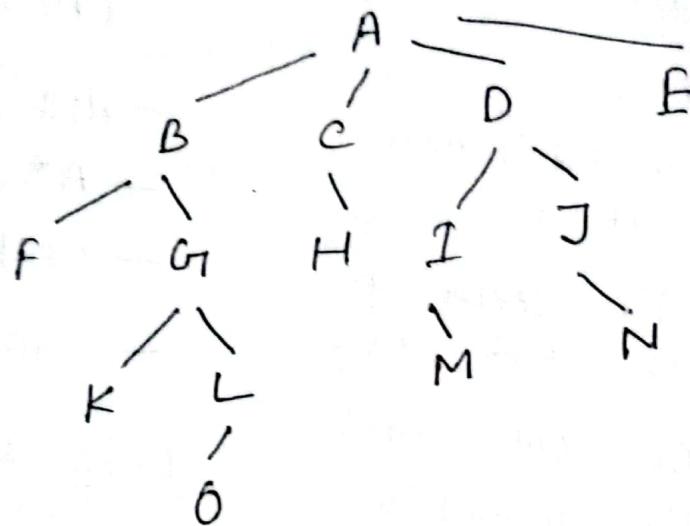
Pros: just go to the best next move, saves time/space for big problems like 8 queens.



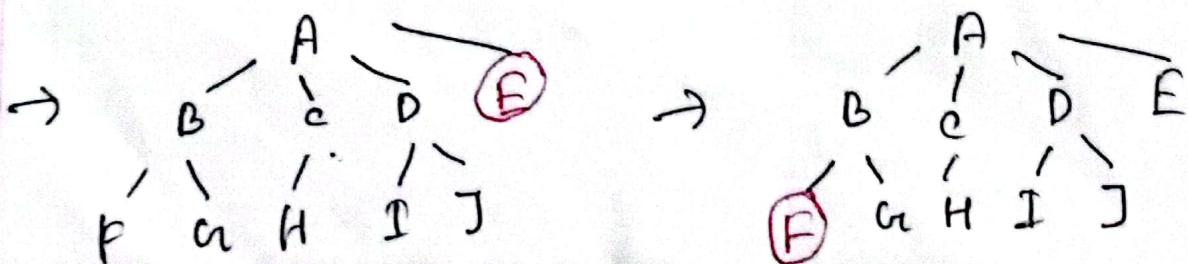
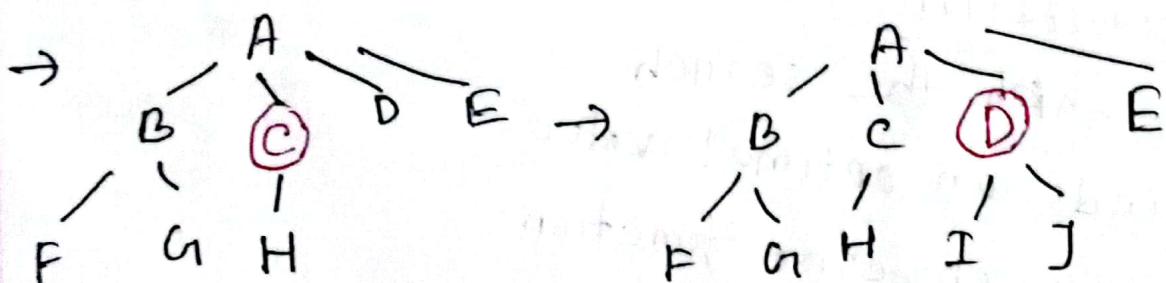
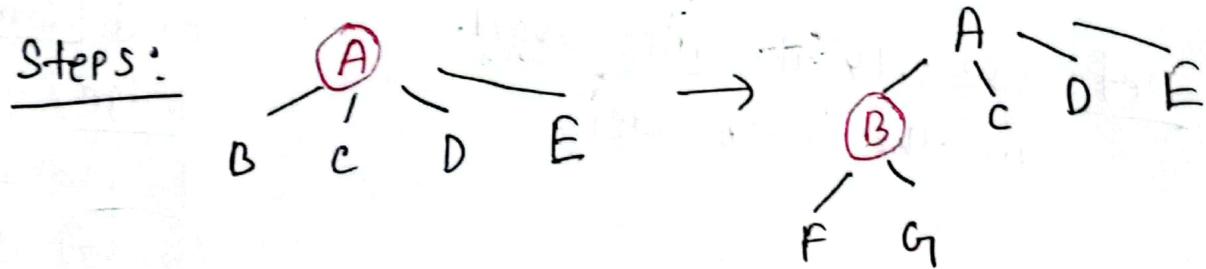
BFS.

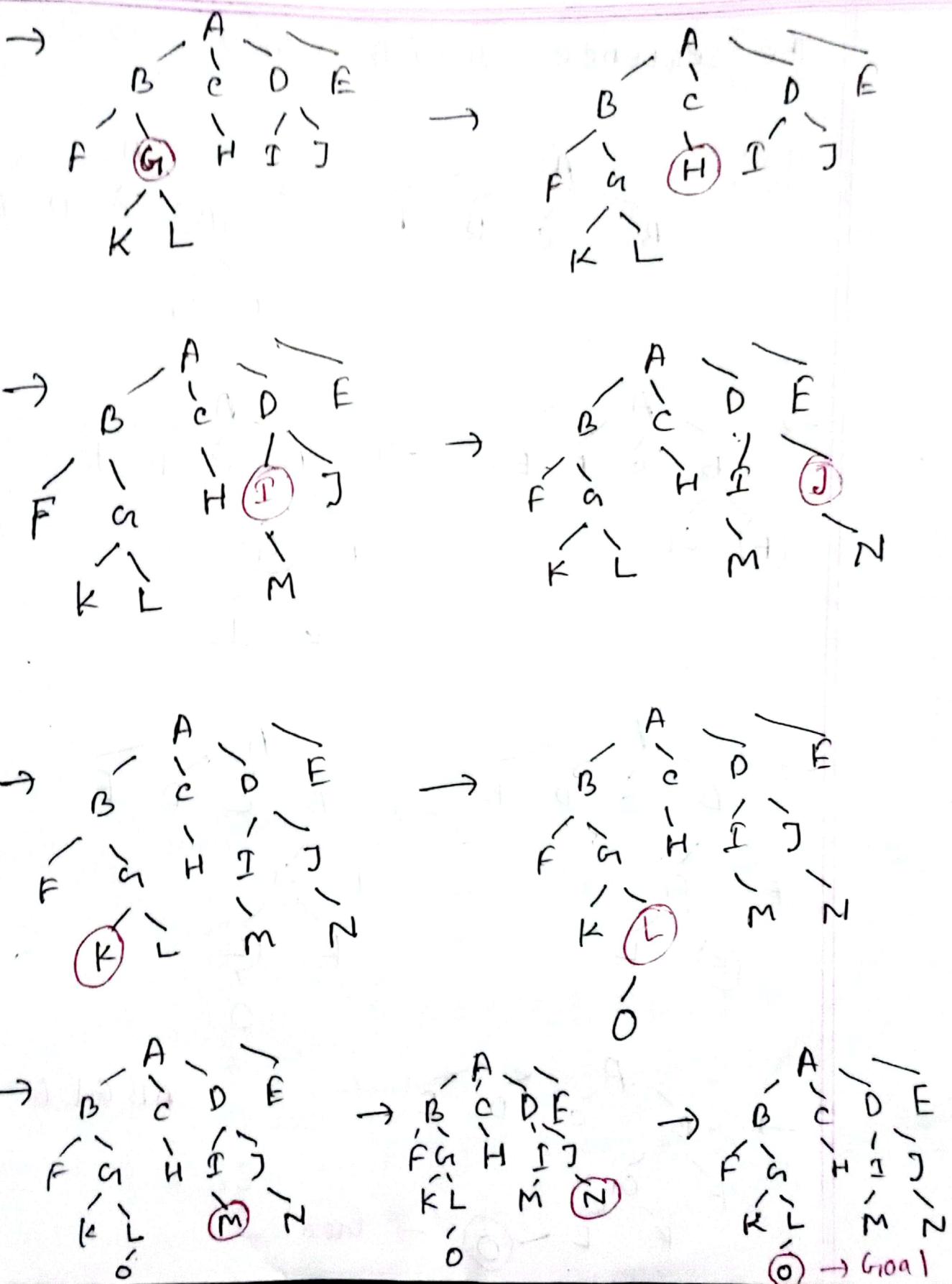
* Given - tree , 0 - goal state
write sequence of visited nodes using

BFS.



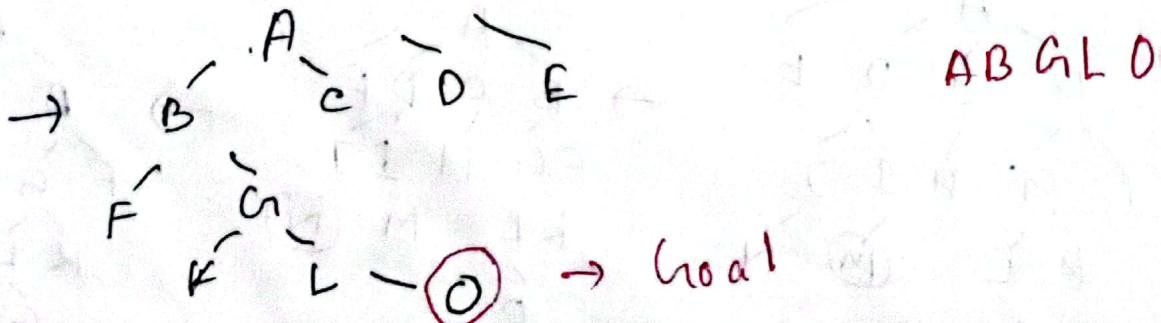
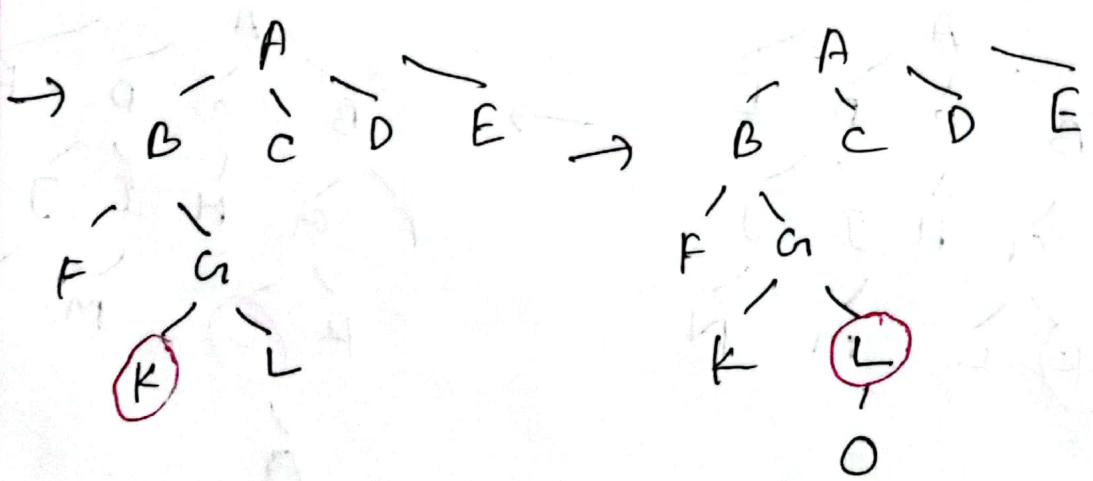
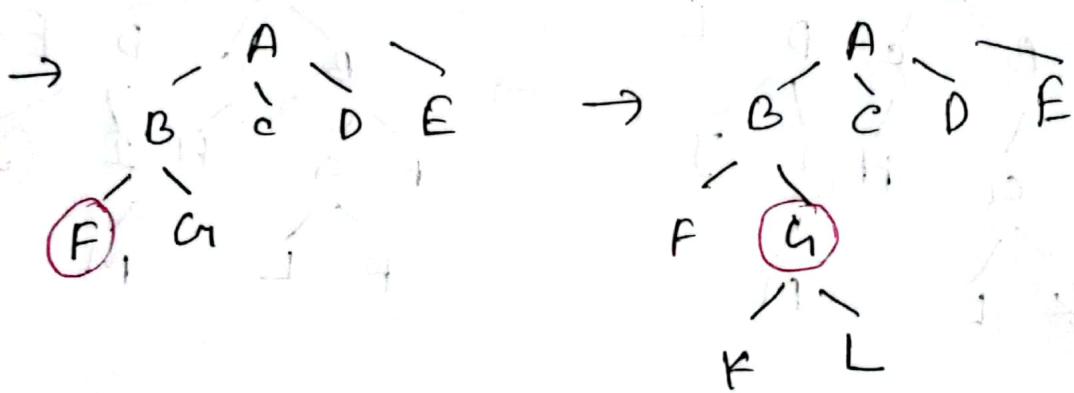
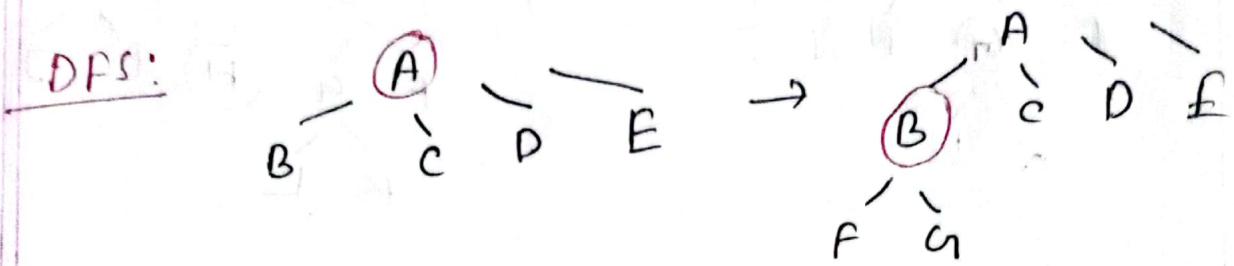
Steps:





The sequence: $A \rightarrow B \rightarrow G \rightarrow L \rightarrow O$

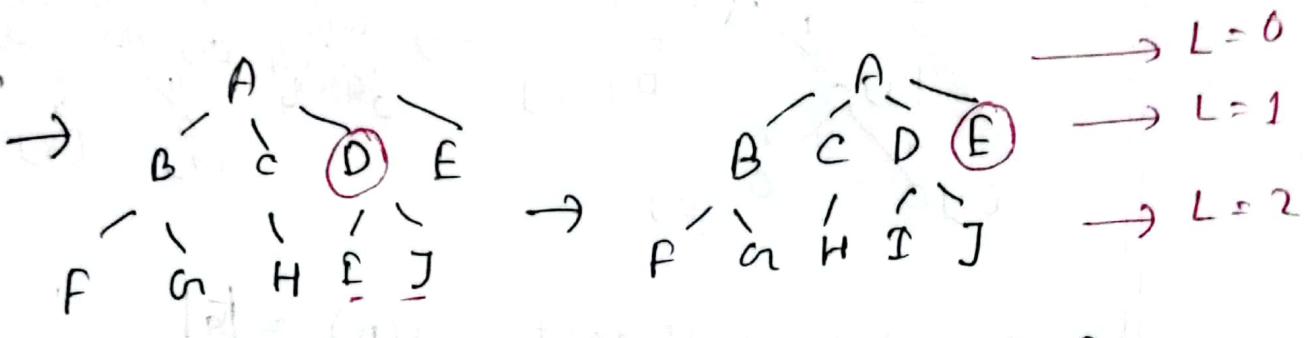
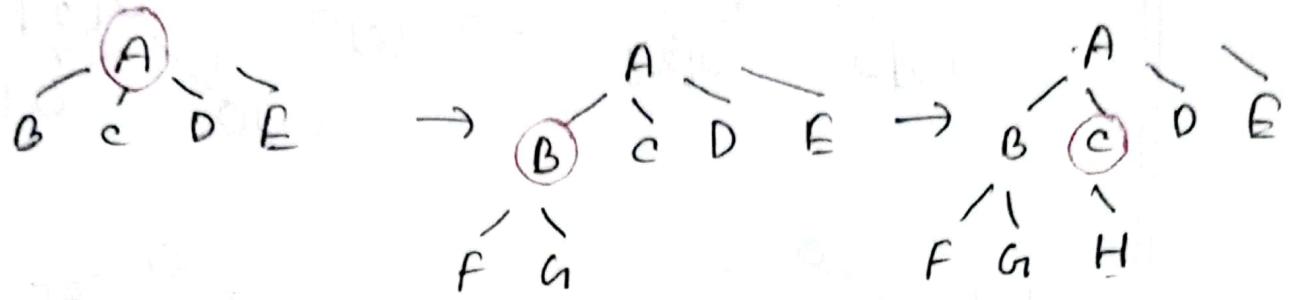
DPS:



DLS and TDS = modified DFS
 ↓
 depth given at first ↳ limit depth = 0, then increase by 1 until goal is found

DLS: Same as DFG, but limit given

Limit = 2



0 not found after traversing level 2.

↳ goal depth = 4, (Given → 2)

TDS: bound / depth limit given 0 at first.

if 0 → A only explore, failed

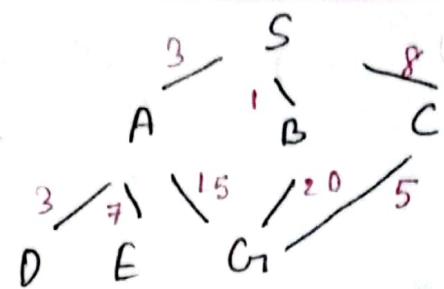
1 → A B C D E " "

2 → A B F G H D I J E

3 → A B C D E F G H I J K L M N E

4 → A B F G K L C H D E M J N E

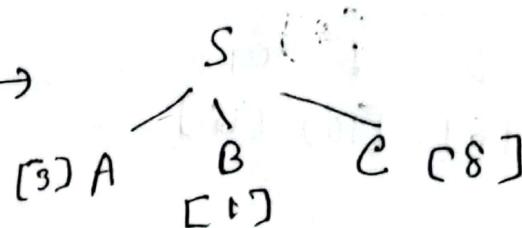
4 → A B F G K L O (goal)



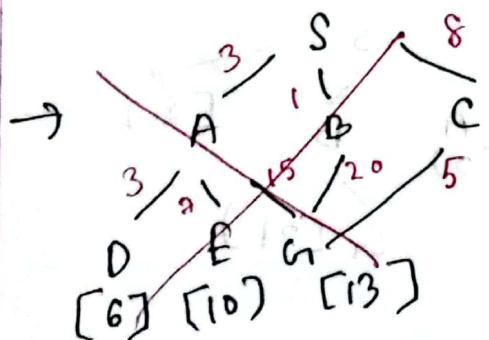
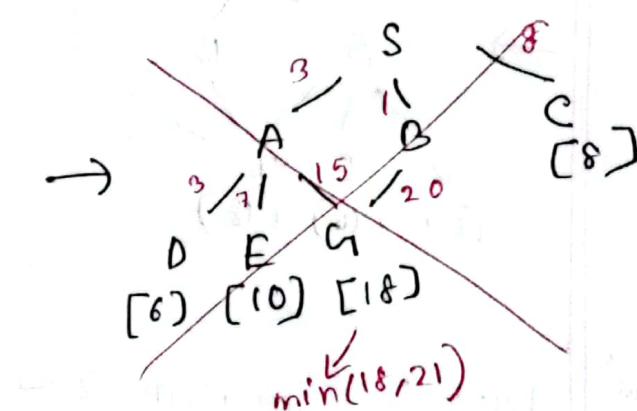
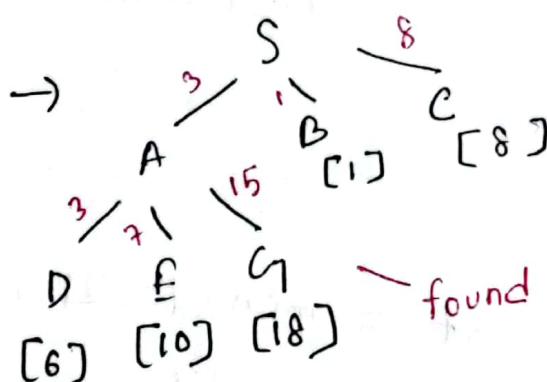
comparisons among BFS, DFS, UCS.
which gives → path cost (min)

DFS:

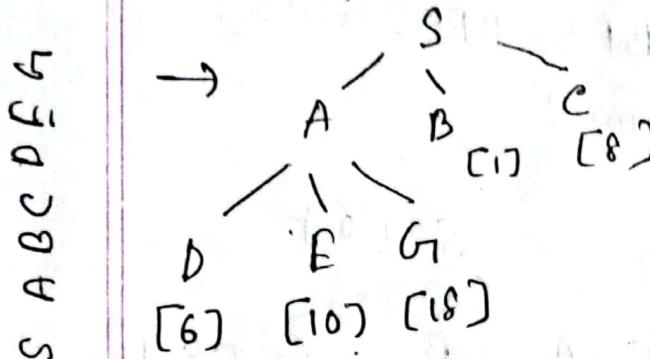
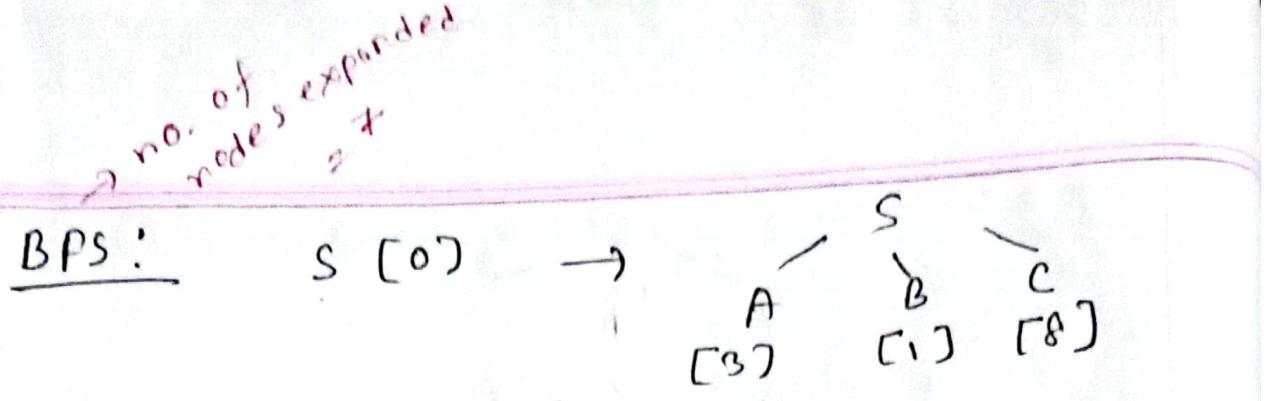
S [0]



S A D E F G
no. of nodes exp. = 5



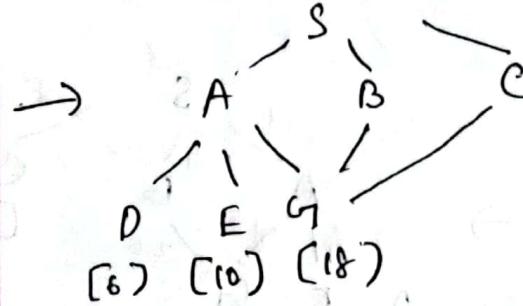
∴ path cost = 18



Ans. 18

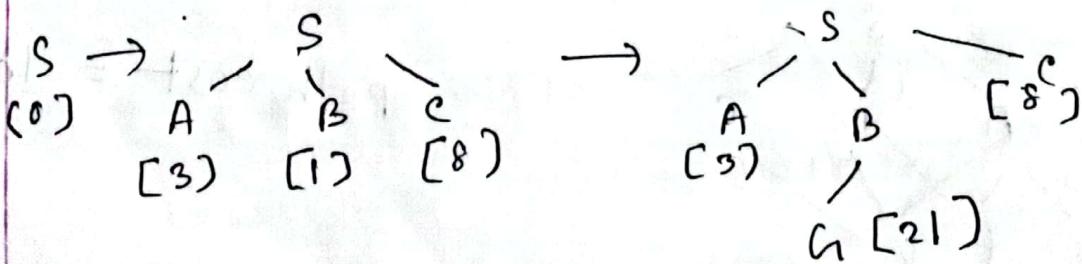
In found already

doesn't change
cost here.

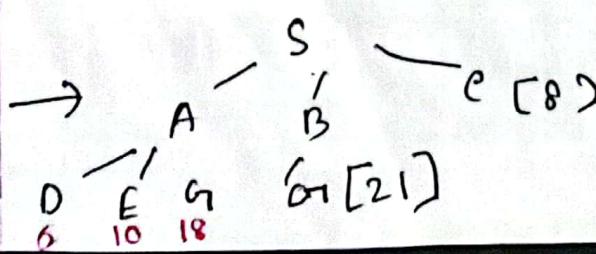


UCS: minimize cost = $\min(18, 21, 13)$
 $= 13$

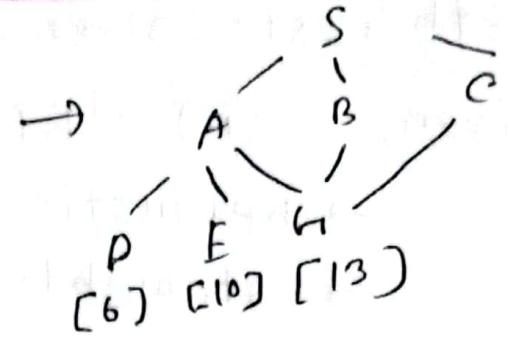
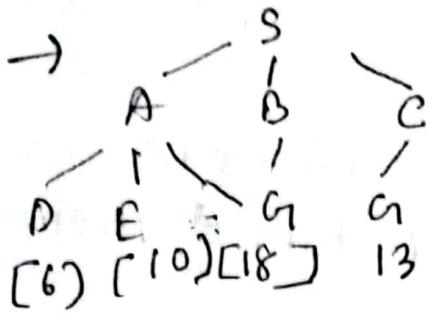
no. nodes



$S B A D$
 $C E G$



explores graph from both start and goal nodes simultaneously



Ans. 13

Bidirectional search:

- 2 simultaneous search from an initial node to goal and backward from goal to initial.
- stops, when 2 meet
- complete in BFS, not in DFS
(may or may not)
- efficient, reduces search space, time
- Time: $O(b^{d/2})$
- worst: $O(b^d)$

$d \rightarrow$ depth
 b - branch factor

(1 min 34) May 19, 2024
at 10:09 AM
with 0:00:00
and 0:00:00

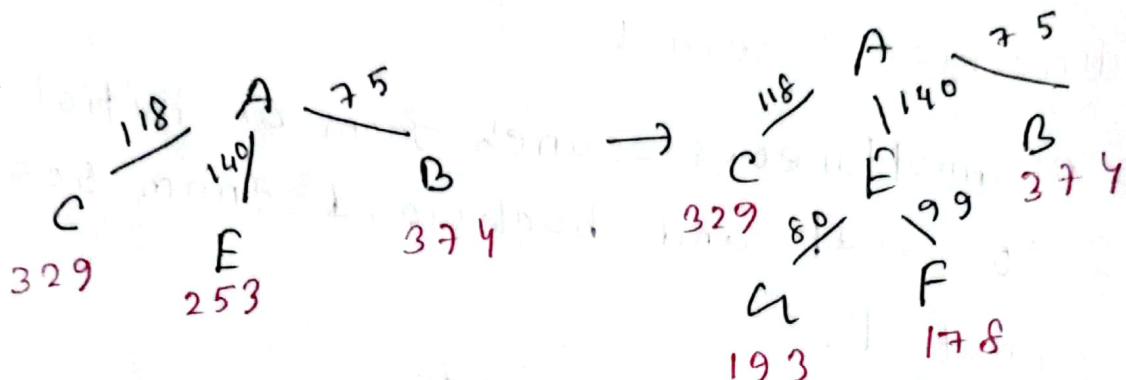
Informed

(unclear) path cost given

given, $f(n) = h(n)$

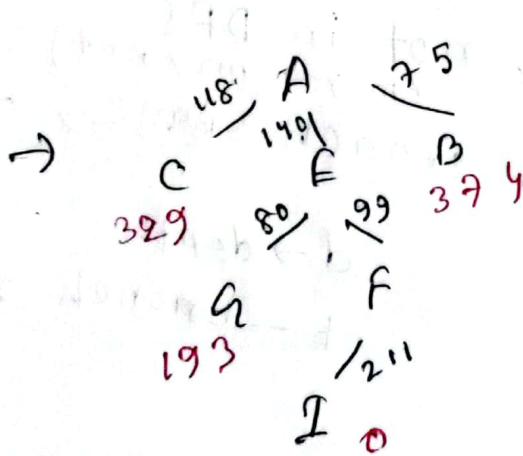
→ heuristic value of node

→ straight line distance from node to goal



seq: A E F I

$$\text{distance} = 140 + 99 + 211 = 450$$



$$\begin{aligned}\text{path cost} \\ &= 253 + 178 + 0 \\ &= 431\end{aligned}$$

not optimal: path (A E G H I)

$$\begin{aligned}\text{distance} &= 140 + 80 + 97 \\ &\quad + 101 = 418\end{aligned}$$

not complete: if not systematically check repeated states.

time: $O(b^m)$

space: $O(b^m)$

* A^*
Admissible

combines greedy search and UCS to get an efficient algo

- complete ✓
- optimal ✓

estimated cost from goal
mode -> node

Greedy:

minimizes $h(n)$

- efficient ✓

- not optimal ✗

- not complete

UCS:

minimizes $g(n)$

- optimal

- complete

- not efficient ✗

exact cost from start to mode

function $f(n) = g(n) + h(n)$

- * optimal if $h(n)$ is not greater than real.
- * complete if node is not repeating (infinite)

Ex:

$$\begin{array}{c}
 & A & 75 \\
 118 & / \quad \backslash & \\
 C & & E & 140 \\
 & 118+320 & & 140+253 \\
 = [447] & = [393] & & 75+279 \\
 & & & = [449]
 \end{array}$$

$h(n)$

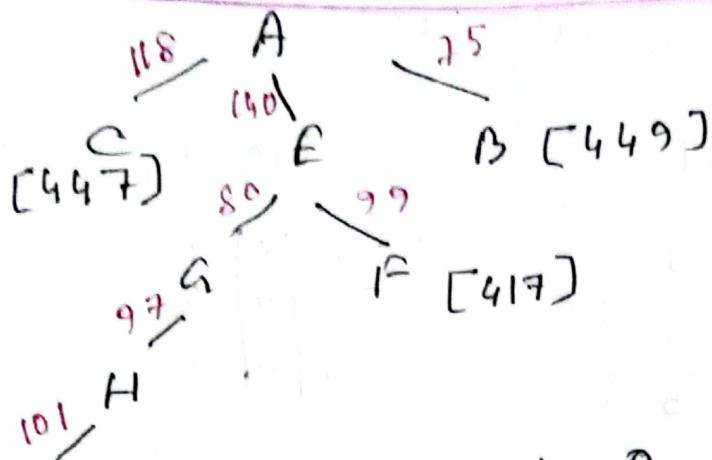
$$\begin{aligned}
 A &- 366 \\
 B &- 374 \\
 C &- 329 \\
 D &- 244
 \end{aligned}$$

$$\begin{array}{c}
 & A & 75 \\
 118 & / \quad \backslash & \\
 C & & E & 140 \\
 & 46 & & \\
 [447] & & & B [449] \\
 & 80 & & \\
 & / \quad \backslash & & \\
 G & & F & 99 \\
 & 293+80+140 & & 178+99+140 \\
 = [413] & = [417] & &
 \end{array}$$

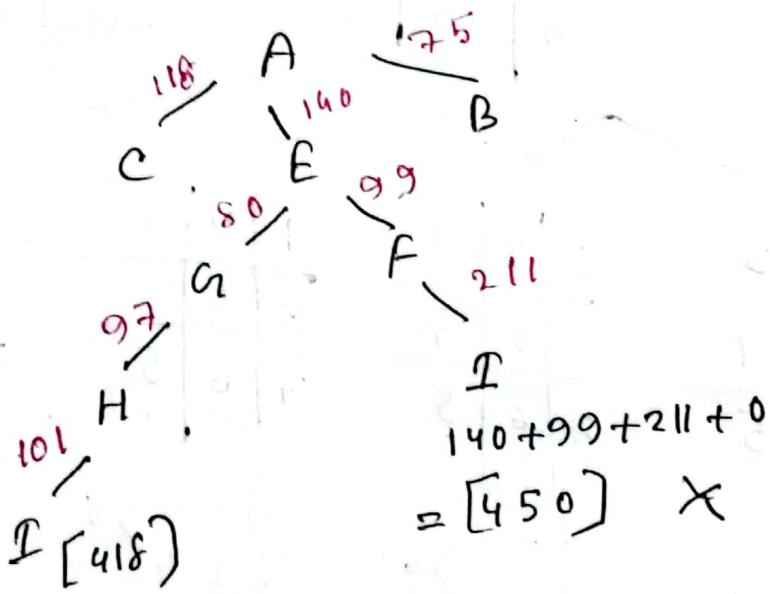
$$\begin{aligned}
 E &- 253 \\
 F &- 178 \\
 G &- 193 \\
 H &- 98 \\
 I &- 0
 \end{aligned}$$

$$\begin{array}{c}
 & A & 75 \\
 118 & / \quad \backslash & \\
 C & & E & 140 \\
 & 46 & & \\
 [447] & & & B [449] \\
 & 80 & & \\
 & / \quad \backslash & & \\
 G & & F & 99 \\
 & 97 & & \\
 & / \quad \backslash & & \\
 H & & F [417] &
 \end{array}$$

$$\begin{aligned}
 & 140+80+97+98 \\
 = [915]
 \end{aligned}$$



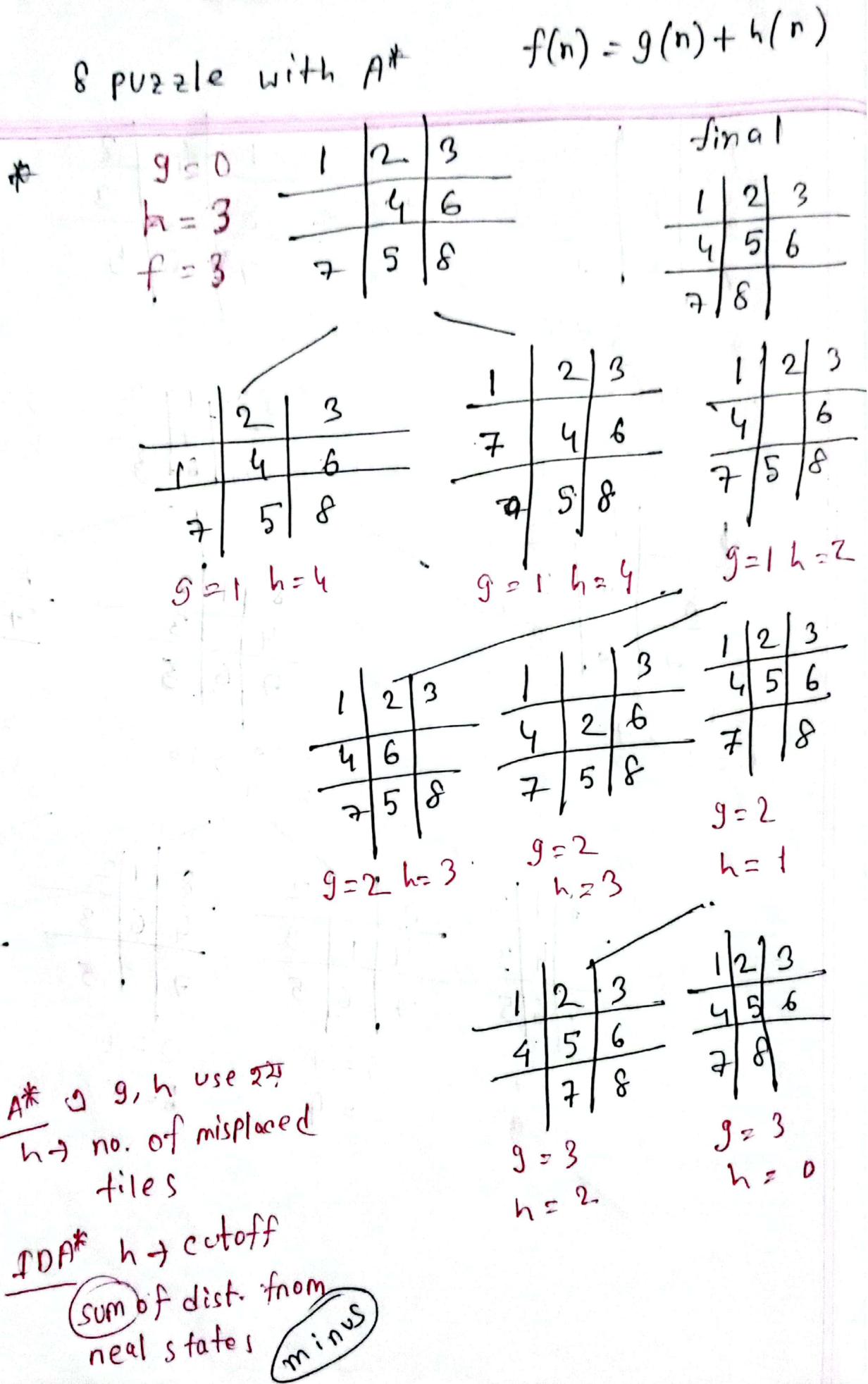
$$\text{I} \quad 140 + 80 + 97 + 101 + 0 \\ = [418]$$



Ans: 418

$$\text{I} \quad 140 + 99 + 211 + 0 \\ = [450] \times$$

with BFS - blind search
— every ans. explore



$h_1, h_2 \rightarrow$ no. of steps to solve n-puzzle

h_1 - no. of misplaced tiles

h_2 - total manhattan distance (sum)

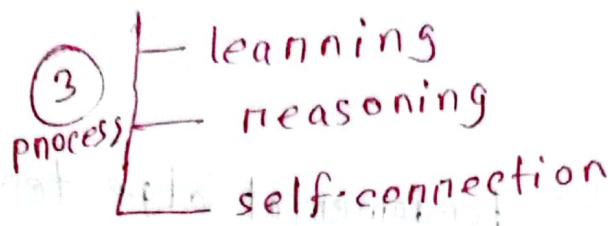
SUCCESSION -

state space search: fundamental algo technique to solve problems in which the solution is a sequence of states, with each state representing a possible configuration of the problem.

Intelligence:
Capacity to acquire
and apply
knowledge

Slide - 1

AI: simulation of human intelligence
processed by computer systems.



Goals: 1) Represent and store knowledge

(5) 2) Retrieve and reason about "

3) Behave intelligently in complex environments

4) Develop interesting and useful apps

5) Interact with people, agent and environment.

4 goals

why AI:

(3)

- 1) Engineering
- 2) cognitive science
- 3) philosophy

foundations of AI

(8)

- | | |
|----------------------|----------------|
| 1) math | 5) Linguistics |
| 2) Eco | 6) Biology |
| 3) psychology | 7) Philosophy |
| 4) cognitive science | 8) CSE |

system that

think like

1) human
2) rationally
act like

3) human
4) rationally

Strong AI

- matches/surpasses human intelligence
- can reason, plan, learn, communicate and integrate skills to solve problems.

Weak AI

- not matches full range of human cognitive skills
- use s/w to solve a specific prob.

Slide-2

Agent: perceives and acts in an environment, has an archt. and is implemented by an agent program.

- Properties:
 - 1) autonomous
 - 2) reactive to environment
 - 3) goal-directed
 - 4) interact with others

Ideal agent: always chooses the action which maximizes its expected performance.

Autonomous agent: uses own experience rather than built-in knowledge of the environment by the designer.

types:

- 1) Reflex agents. → respond immediately to percepts
- 2) Goal-based " → act to achieve goal
- 3) Utility " " → max. utility func.

Rationality: measures how well a task is achieved

Rational agent: max its expected performance
measure based on
1) percept sequence
2) built-in and acquired knowledge

Performance
measures:

- false positive - speed - effect on environment
- " neg - neg. resources "

Agents with memory: keep track of past states

Agents with goals: has

- state info
- goal "

Logical reasoning: process of deriving new
sentence from old sen.

such that new \rightarrow true
if old \rightarrow true

model based agent: whose action is derived directly from an internal model of the current world state that's updated over time.

7	2	4
5		6
8	3	1

1	2
3	5
6	8

7 is 3 sqrt \rightarrow (3)

2	"	1	sqrt.	\rightarrow	(1)
5	\rightarrow	2		\rightarrow	(2)
4	\rightarrow	2		\rightarrow	(2)
6	\rightarrow	3		\rightarrow	(3)

$$3 + 1 + 2 + 2 + 3 + 2 + 2 + 3 \rightarrow 18 = h2$$

Given,

<u>state</u>	<u>$h(n)$</u>	<u>state</u>	<u>$h(n)$</u>
Anad	366	Mehadia	241
Bucharest	0	Neamt	234
Craiova	160	Onadea	380
Dobreta	242	Pitesti	100
Efonie	161	Rimnicu Vilcea	193
Fagaras	176	Sibiu	253
Giurgiu	77	Timisoara	329
Hinsonava	151	Urziceni	80
Iasi	226	Vaslui	199
Lugoj	244	Zenind	374

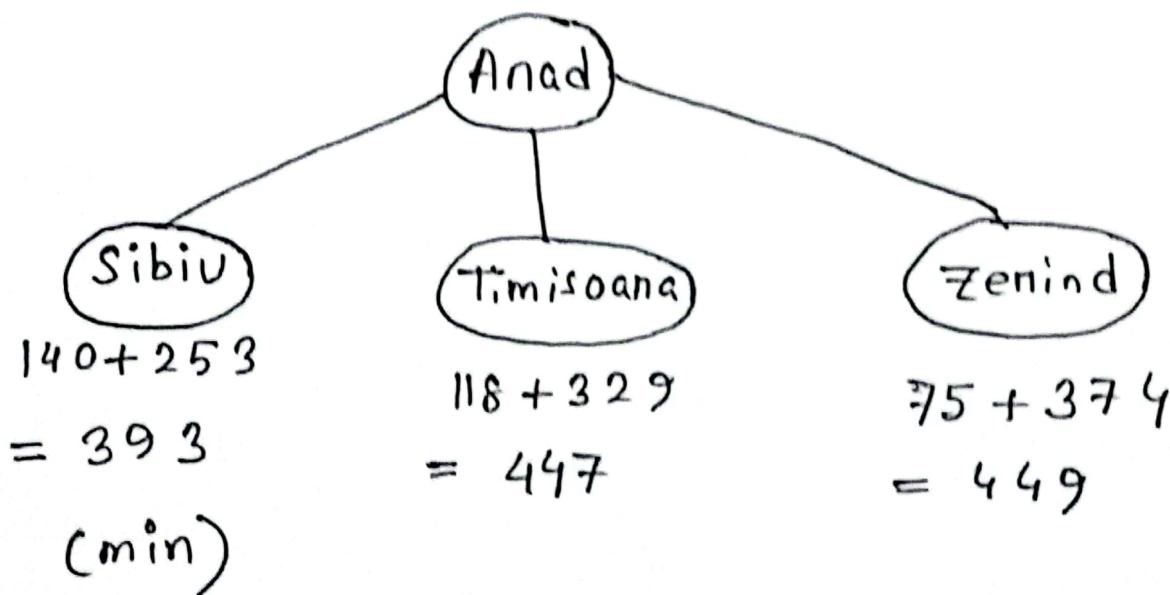
Steps to go Bucharest from Arad:

Step 1:

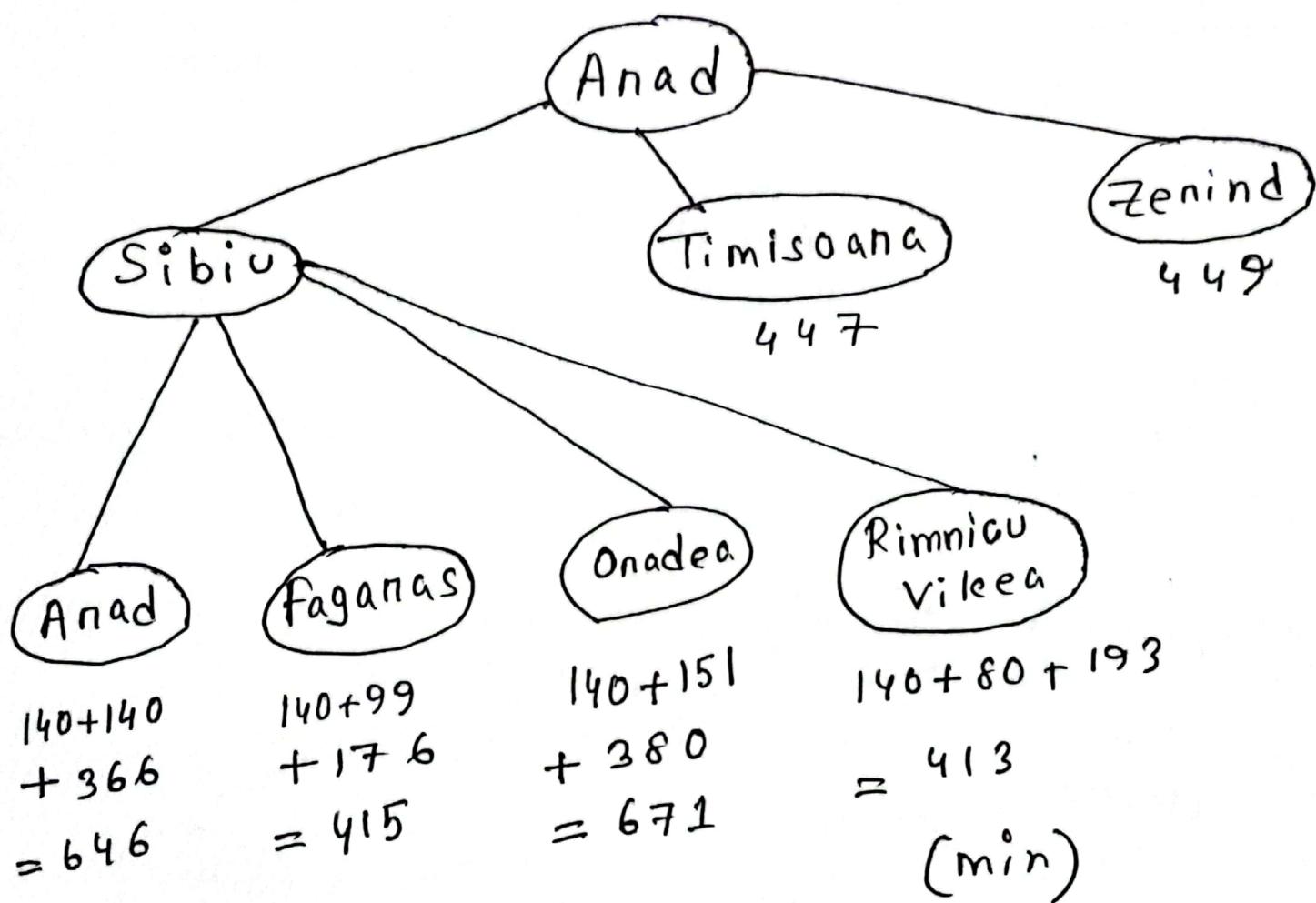
Anad

$$0 + 366 = 366$$

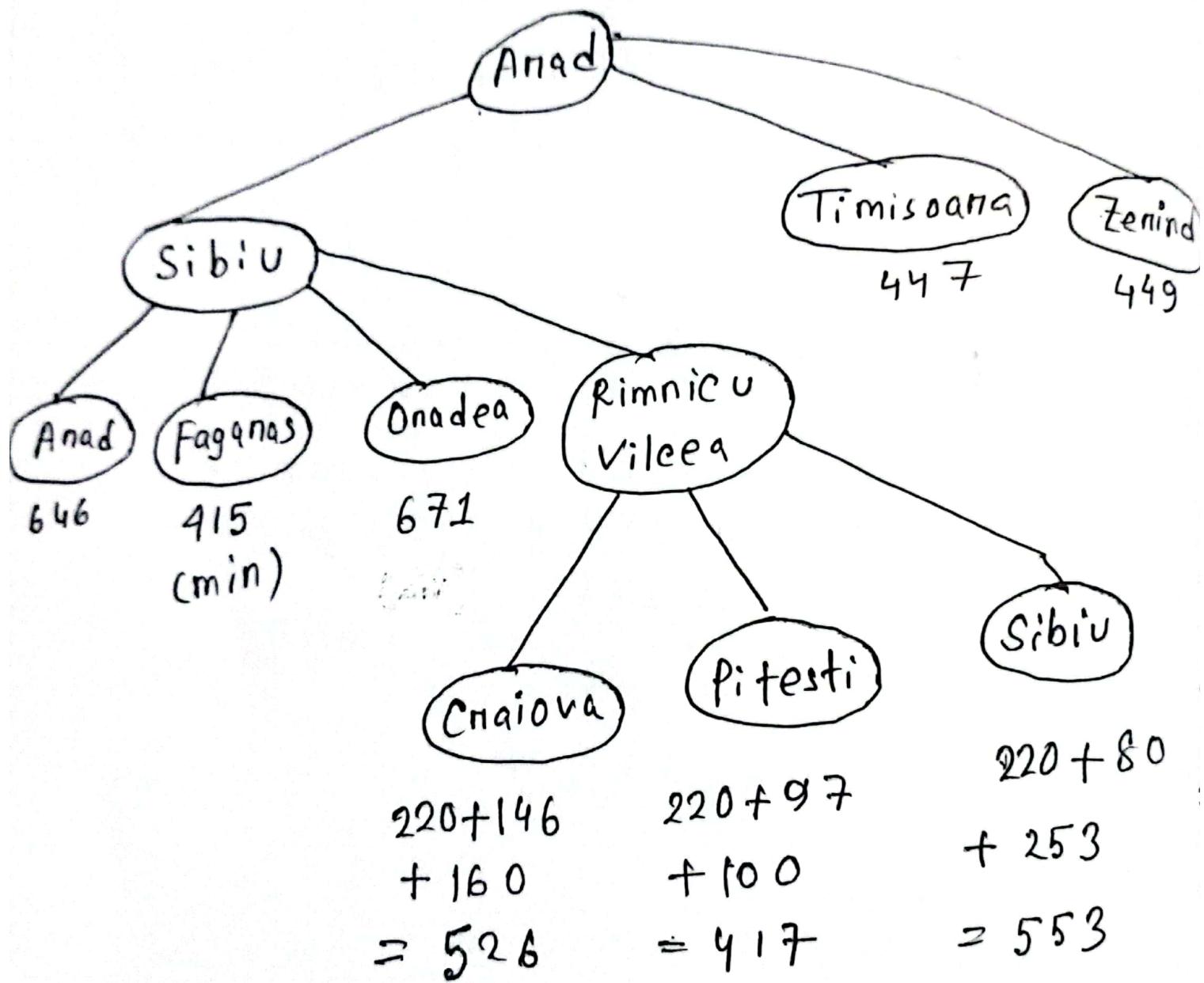
Step 2: Expanding Arad, we get:



Step 3: Expanding Sibiu, we get,



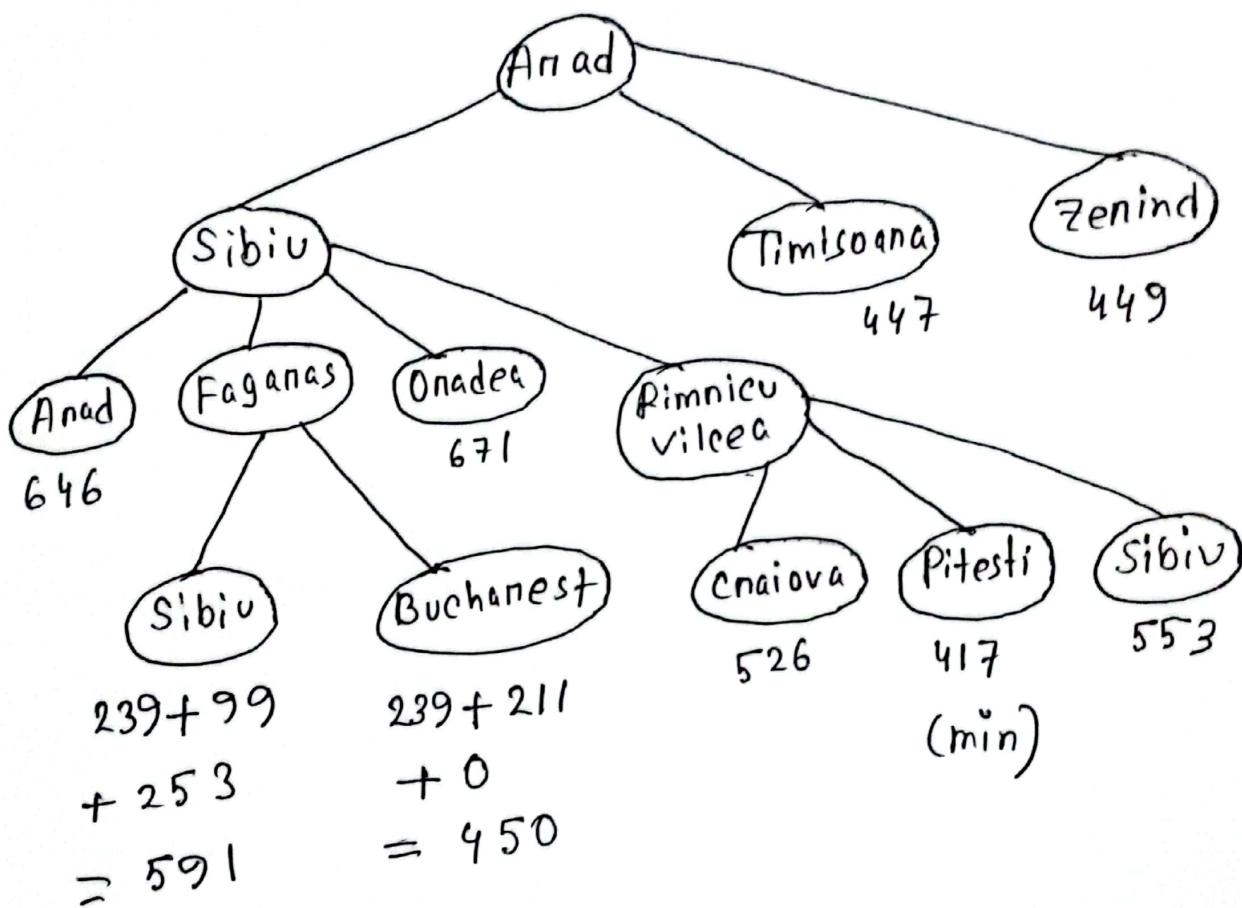
Step 4: Expanding Rimnicu Vilcea, we get





মুজিব
মাজবুর 100
MUJIB
MABUR 100

Step 5: Expanding faganas, we get,





Step 6: Expanding Pitesti, we get.

