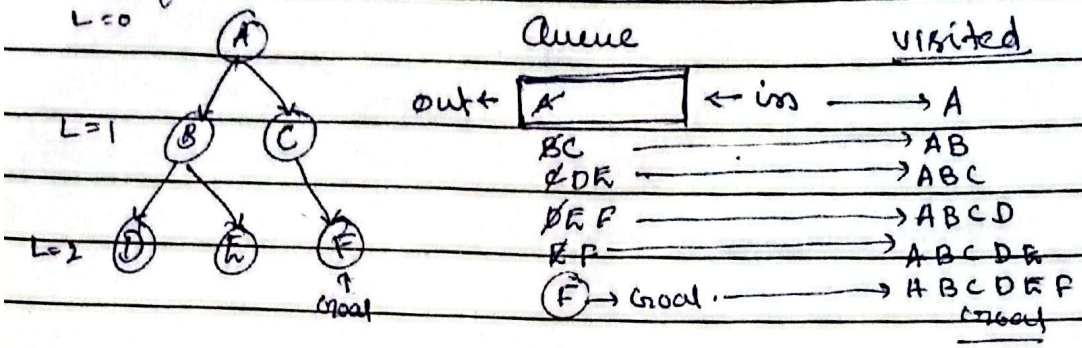
Day: 7-Feb Date:

AI Self Study: →

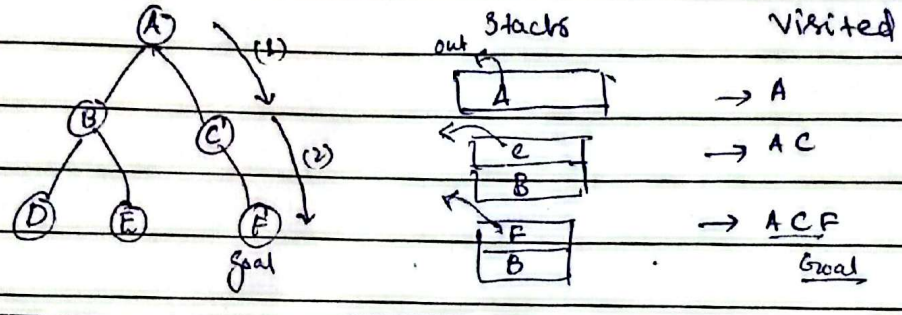
Breadth-First Search (Queue) [FIFO]

↳ Example:



Depth-First Search (Stack) [LIFO]

Example:



~~Bidirectional Search: $O(b^{d/2})$~~ →

↳ Two simultaneous searches:

↳ one from start node
↳ one from goal node — ~~Both meet at (mid)~~

This is Bid Search.

↳ BFS → can give us complete search under $2(b^{d/2})$
↳ DFS → ~~Both can be used.~~
↳ ~~This can give us (b^d)~~ → here time comp increased.
↳ when paths parallel.

BFS preferred.

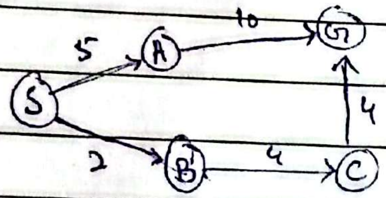
Day: Date:

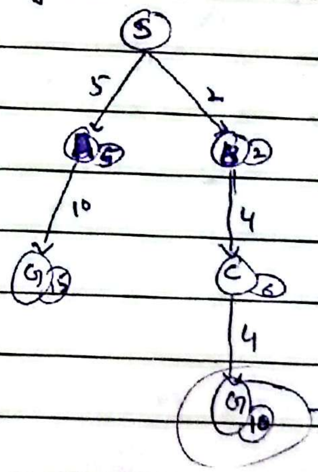
A key can be foreign key
↳ can be primary key.

Cascade → change in parent will remain in child.

Views → Such table which is created at runtime.
↳ not stores somewhere just created in ~~ram at runtime.~~
↳ purpose:
↳ same data can be seen by different users in different ways.

* Uniform Cost Search (Priority Queue → Sasta Pasta)





→ visited
SBAGCG

→ path
S→B→C→G
10

Queue (Priority)

out ← | ~~S~~ [redacted] | ← in

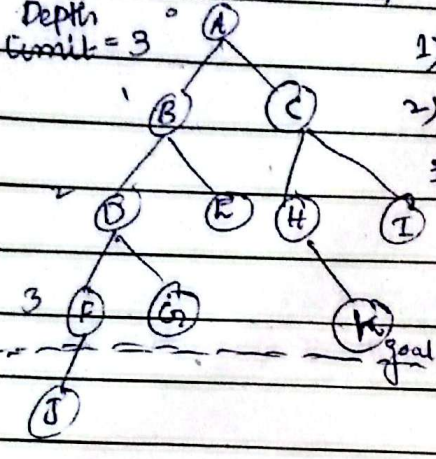
~~B~~ A
~~A~~ C
~~C~~ G15
~~G~~ G10 (15)

G10 ✓

Day: Date:

Problems in DFS
↳ infinite loop → going in depth
↳ no receiving goal → Search vast
Space

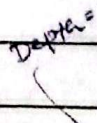
Depth-limited Search:



→ paths:

1) A → B → D → F → G → E → C → H → K
2) A → C → I → H → K
3) A → C → H → K → Shortest.

Depth = 3

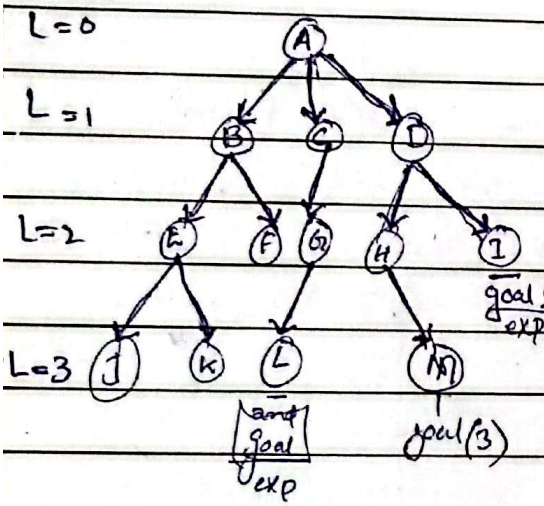


* DFS-limited still incomplete
If solution exist beyond limit

* TC = $O(b^L)$ → limit
↳ branch factor

Space comp $O(L)$

Iterative deepening: (uses Both DFS & BFS)



I=0 A

I=1 A → B → C → D

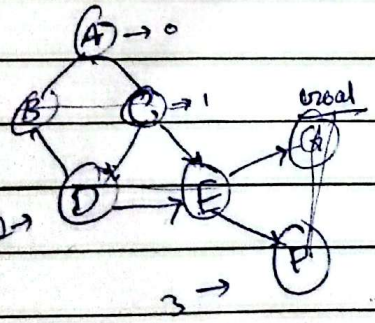
goal=1 for exp

I=2 A → B → E → F → C → G → D → H → I

~~I=3 A → B → E → F → C → G → D →~~

I=3 = A → B → E → J → K → F → C → G → L → goal.

I=3 = A → B → E → J → K → F → C → G → L → D → H → M.
goal found.



L = 0 = A

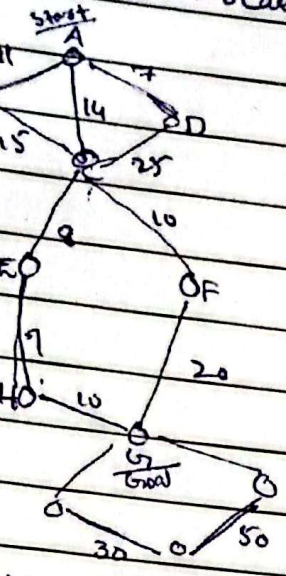
L = 1 A → B → C

L = 2 A → B → D → C → E → G.

Day: _____ Date: _____

* Informed Search Technique:

* ↳ Best-First-Search: [here heuristic val: is (Straight line Dist)]



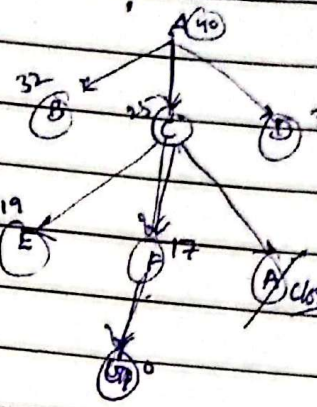
A to G

queue: ~~A B D E~~ A

~~E~~ B D

~~F~~ E

G



Path:

A → C → F → G

↳ good path ↳ 34 cost

optimal also

T.C = (bd)

(Uses → priority-queue)

Heuristic values

straight line dist:

A → G = 40

B → G = 32

C → G = 25

~~D → G = 35~~

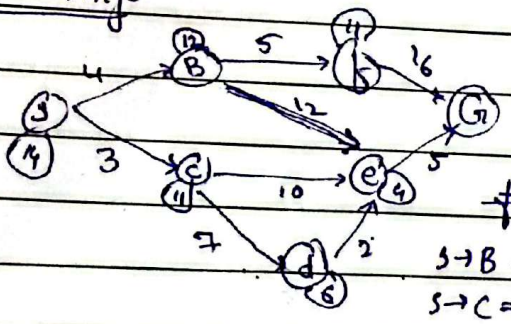
E → G = 19

F → G = 17

~~A~~ → G = 10

G → G = 0

A* → Algo



$f(N) = g(N) + h(N)$

actual cost from start to (n) ; estimated cost from (n) to goal

~~f(G) = f(B) +~~

S → B = 4+12 = (16)

S → C = 3+11 = 14

SC → e = 3+10+4 = 17

SC → d = 3+7+6 = (16)

Scd → e = 3+7+2+4 = 16

Scde → G = 3+7+2+5 = 17

S → n → Goal

g(n) h(n)

SB → f SB → e

= 4+5+11 = (20) = 4+12+4 (20)

T.C = O(bd)

path is Scde G