

COP 3503 - 3/3/2022

① BFS Applications

(a) 8 Puzzle

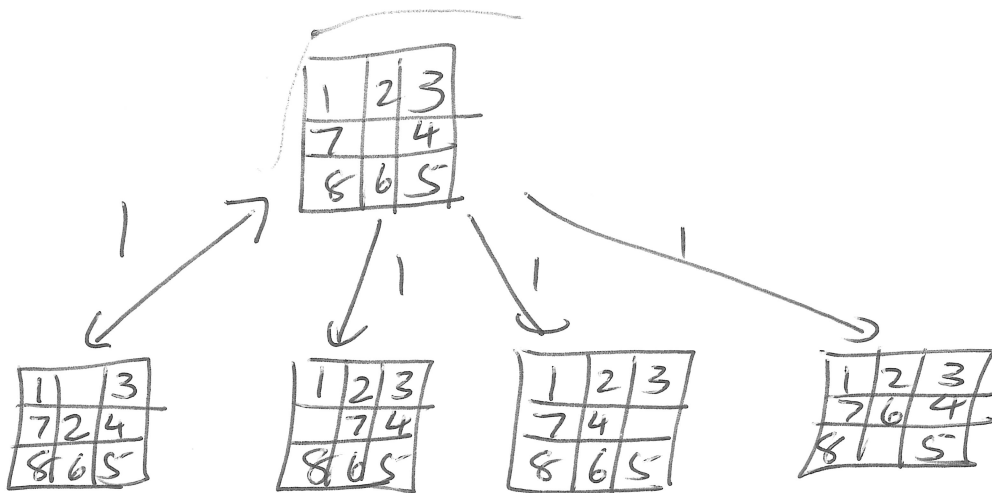
(b) Rush Hour

~~(c) Great Wall Problem~~

② Top Sort

(a) Algorithm - Iterative

(b) Dueling Philosophers



Reg BFS $O(V+E) \sim 700,000$ ops

$$V = \frac{9!}{2} \quad E \sim 4\left(\frac{9!}{2}\right)$$

Can't repeat 10,000 times

Observation #1: ALL MOVES ARE BIDIRECTIONAL

Reverse the direction START BFS

1	2	3
4	5	6
7	8	

Run it once marking ALL distances

Why is this harder to implement?

ANS: Orig BFS vertices labeled 0, 1, 2, ... n-1
getting to vertices easy storing distances
easy.

What is the vertex ID for

1	2	3
7		4
8	6	5

?

1) Bitwise Magic: Store each item as a 4-bit mask
0001 0010 0011 0111 0000 0100 1000 0110 0101
HashMap < Long, Integer > → DIST

2) String → Int map "123704865"

3) Int → Int 123704865 (9 digits ⇒ Int)

4) HashMap < Int & Int, Integer > - a little slower

I did #3 in my posted soln

NO MATTER WHAT STORAGE YOU CHOOSE
FOR THE MAP, THESE ARE THE KEY
STEPS

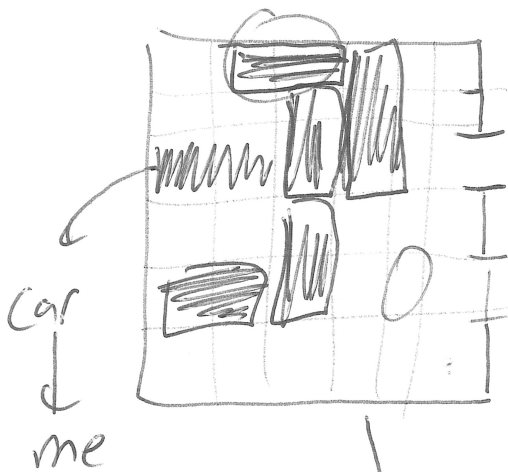
(1) Be consistent in your storage.

(2) Make sure storage is unique.

(3) Write a function to convert Board → Storage
Storage → Board.

↳ If you DON'T do this you have to move directly
in your storage.

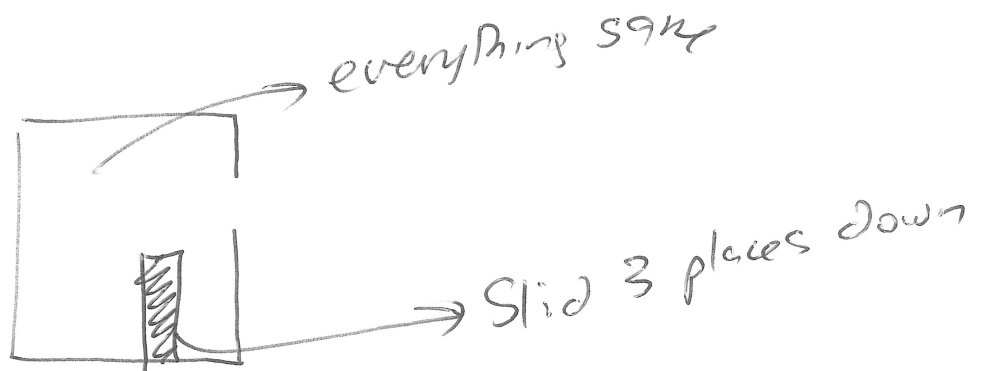
Rush Hour



Goal: Get me out
all cars slide either $\updownarrow$ or



Board pos



How to represent?

My representation

for each piece

top left corner	(0,1)	→ 6 bits
orientation	0	0=H, 1=V
size	2/3	⇒ 0/1

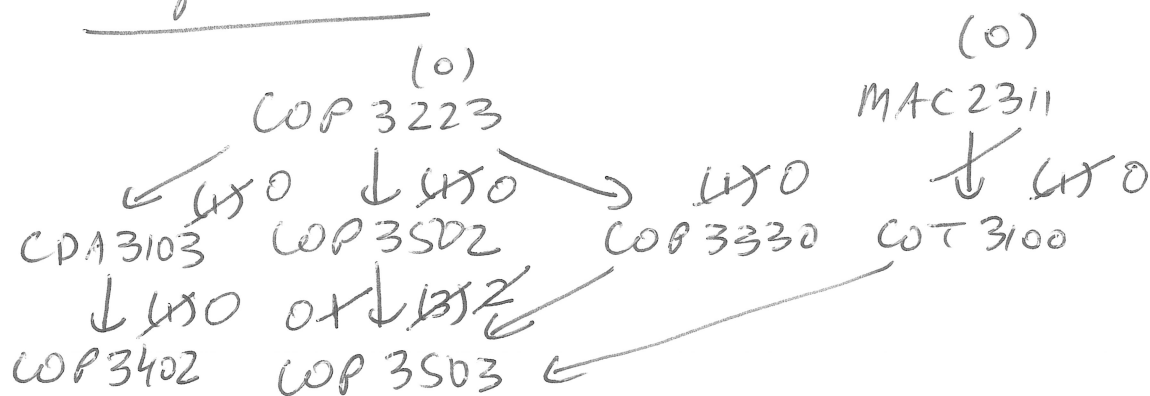
8 bits per piece {LONG}

↓
8 pieces max.

Another idea: each piece # 1, 2, 3, 4,

001000
222 ... } MASK or String

Top Sort



Is there an ordering of these tasks that satisfies the constraints?

In degree of vertex is # of edges coming into it in a directed graph

Store graph and all in degrees

Maintain a queue of vertices w/in degree 0.

Q: COP 3223, MAC 2311

while (q isn't empty) {

1) take the next item from the queue.

2) Place it ^(v) in our ordering next.

3) For edge leaving v:

remove it from the graph
(subtract 1 from vertex x that the edge goes to.)

if (degree(x) == 0)

add x to queue

Q: ~~MAC2311~~, ~~LOP3223~~, ~~LOT3100~~, ~~CDA3103~~,
~~LOP3502~~, ~~LOP3330~~, ~~LOP3402~~, ~~LOP3503~~

Schedule: MAC2311, LOP3223, LOT3100, CDA3103, LOP3502
LOP3330, LOP3402, LOP3503

Fail case: queue is empty (finishes)
but schedule is NOT full!