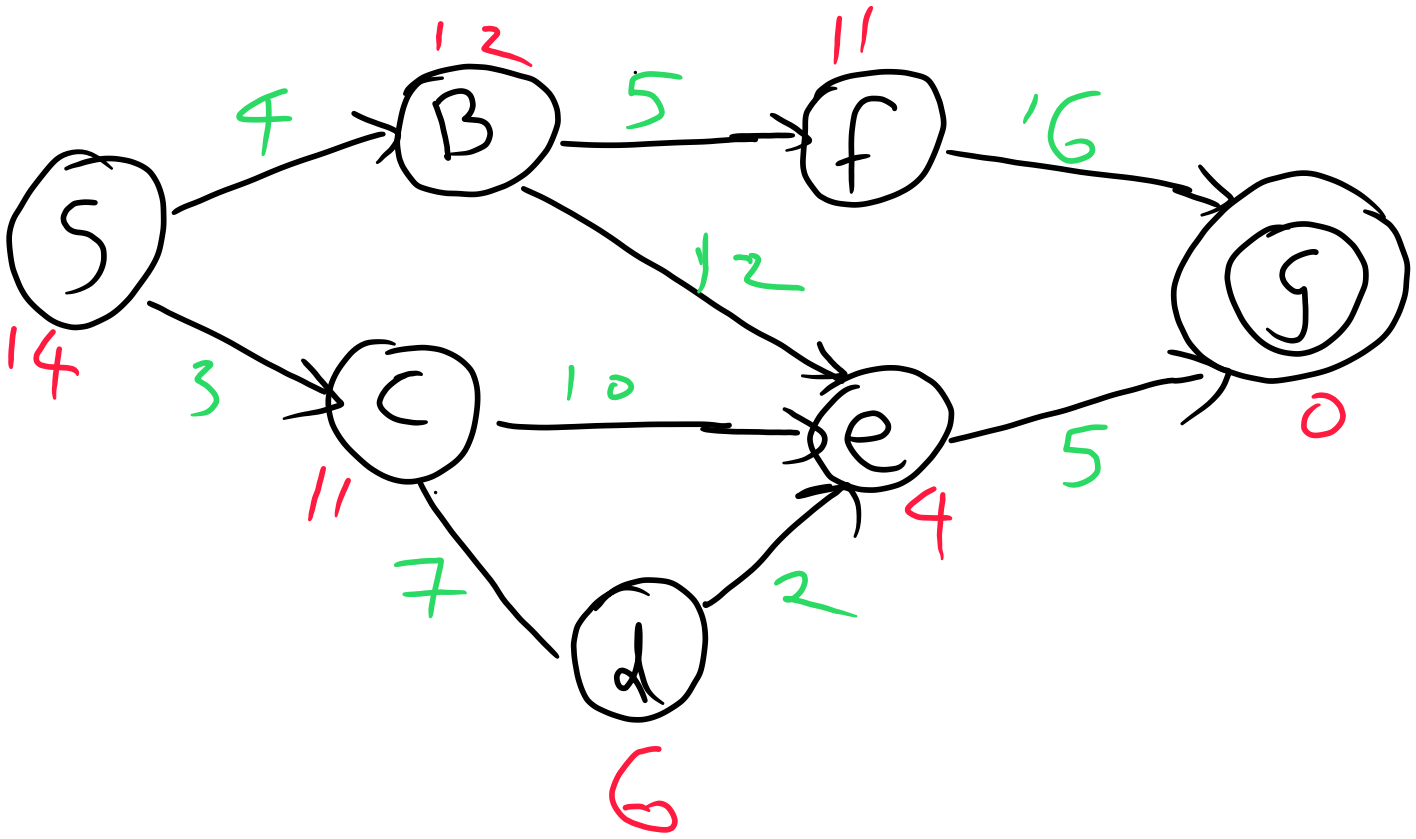


A*

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

GBFS

$$f(n) = h(n)$$

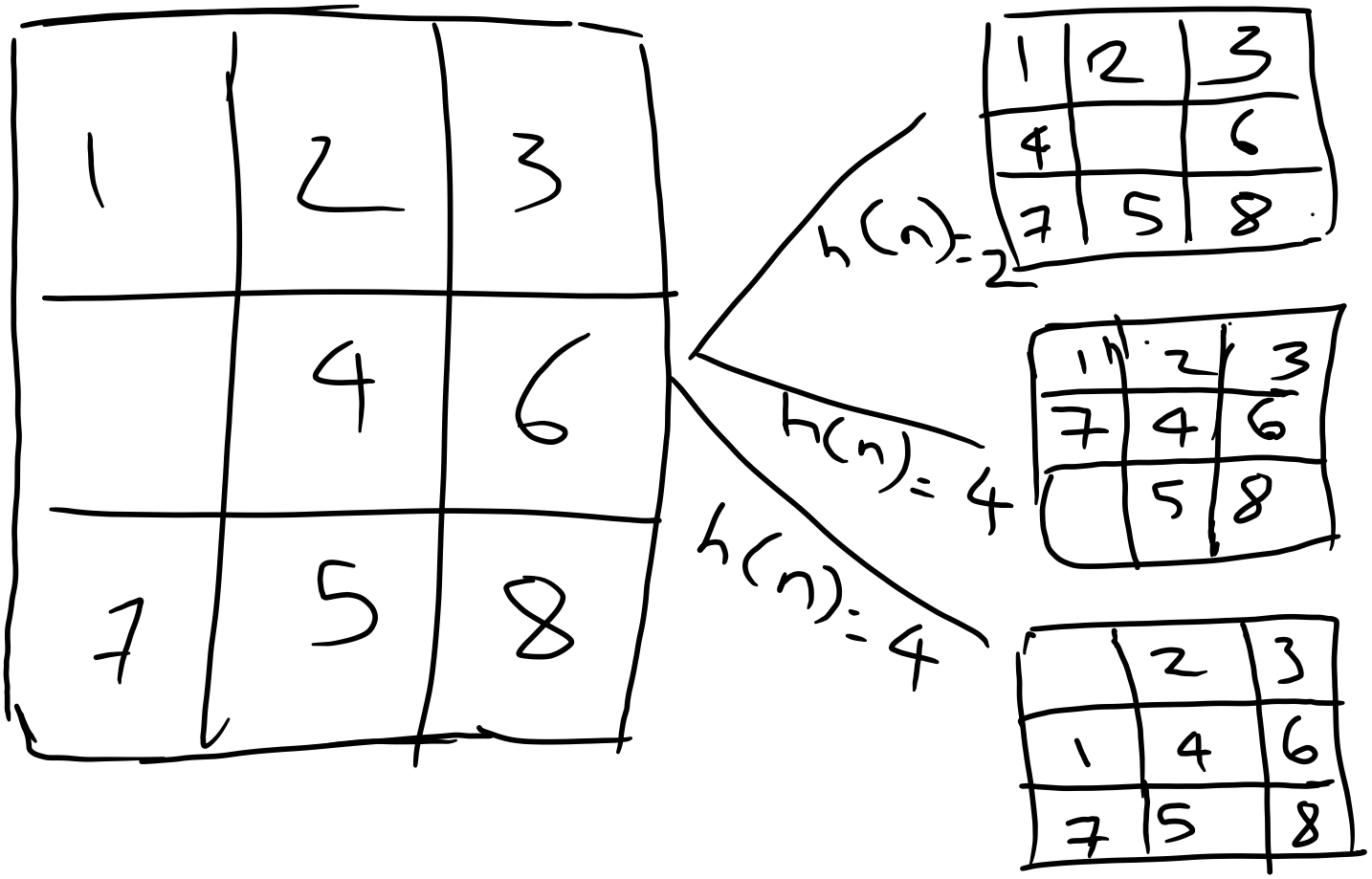
Q Solve
8 tiles/8 puzzles problem using GBFS & A* search

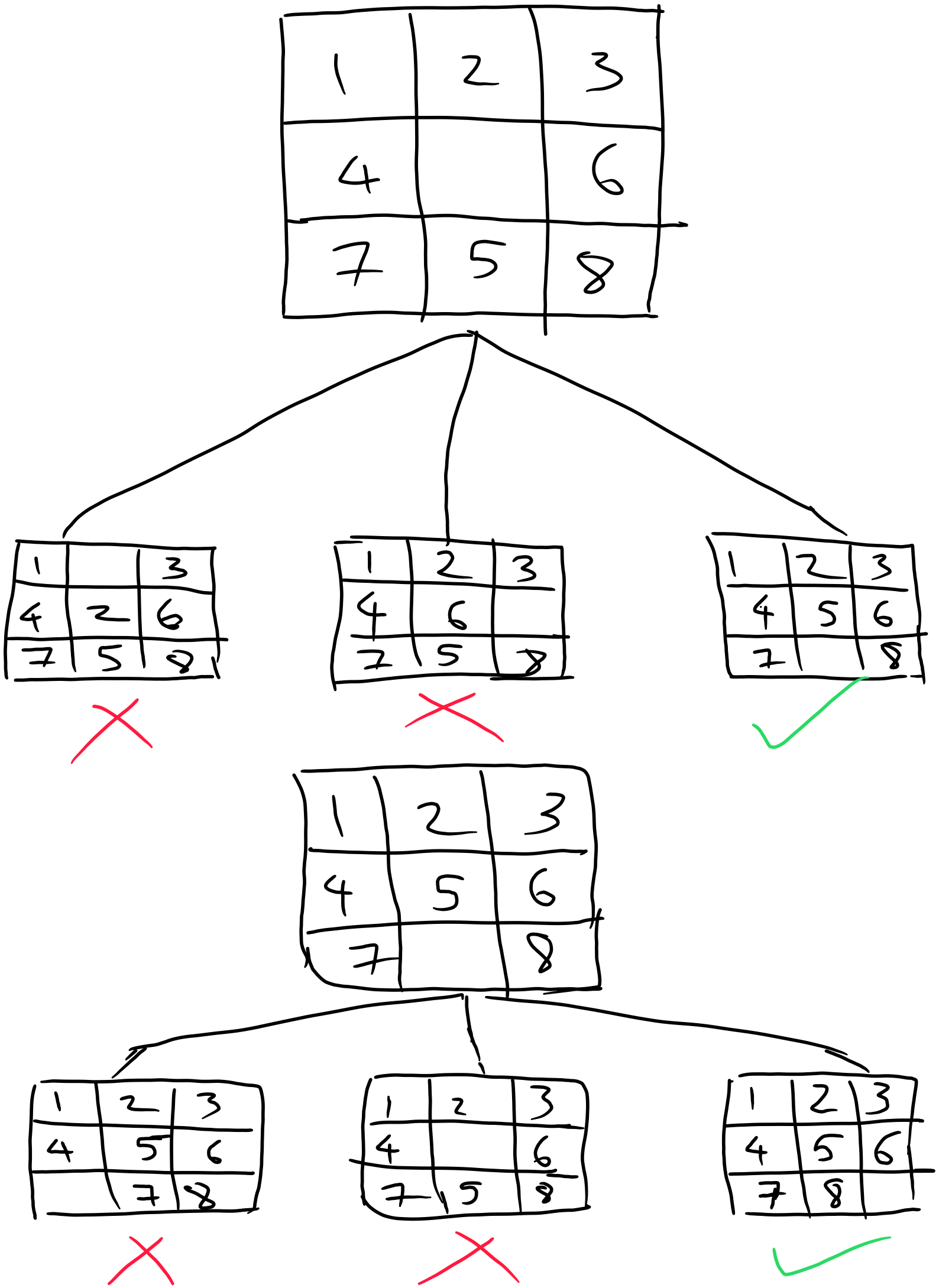
1	2	3
	4	6
7	5	8

Start State

1	2	3
4	5	6
7	8	

Goal State




1 2 3
4 6
7 5 8
1 3
4 2 6
7 5 8
1 2 3
4 6
7 5 8
1 2 3
4 5 6
7 8
1 2 3
4 5 6
7 8
1 2 3
4 6
7 5 8
1 2 3
4 5 6
7 8

h(n) → Manhattan Distance

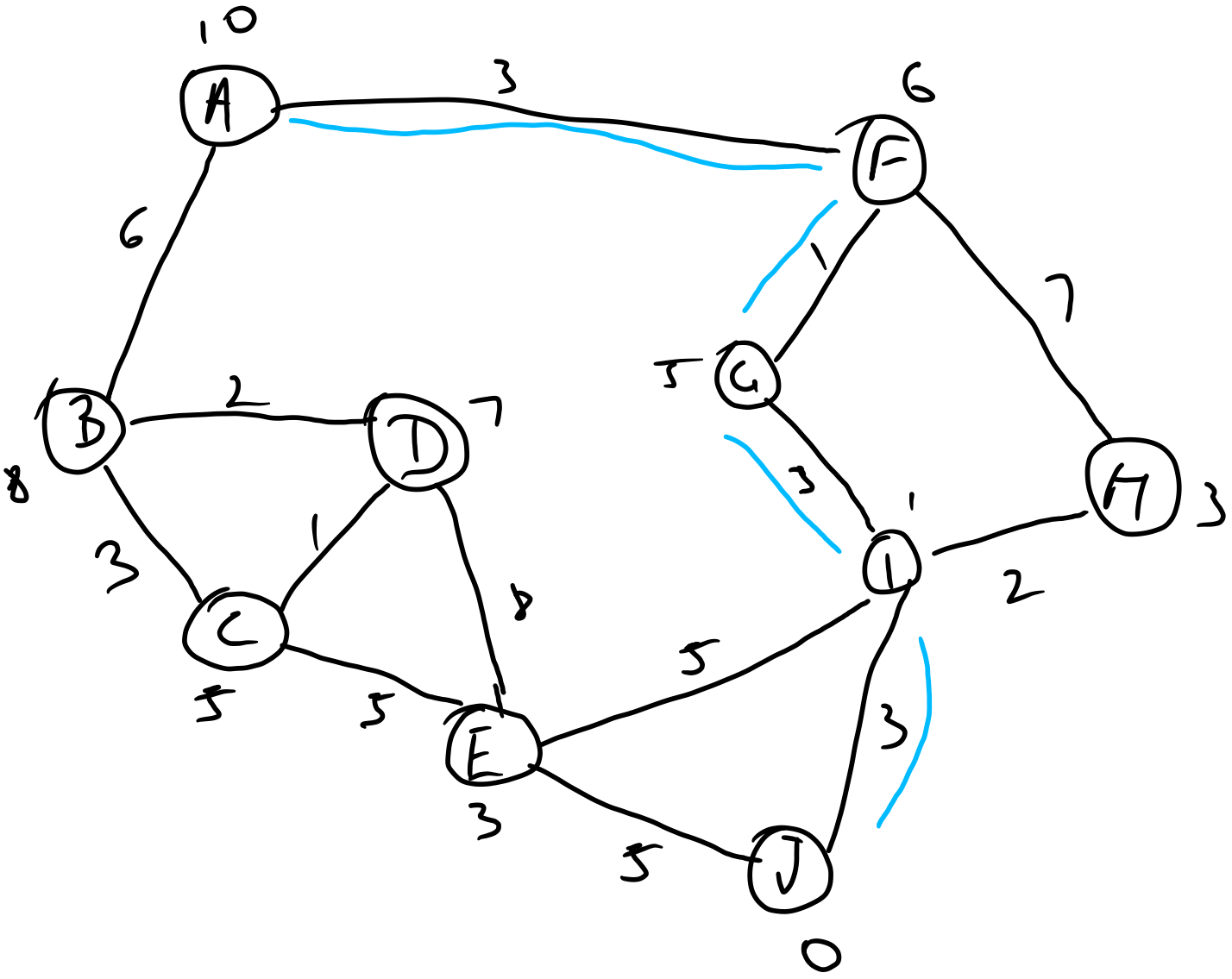
Step 1:

1	2	3
4		6
7	5	8

0 → already in place

n → no. of moves required to move to the final place

h(n) = 0 + 0 + 0 + 0 + 1 + 0 + 0 + 1



1

Source ⇒ A
Destination ⇒ J

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

A → F : 9 ✓
A → B = 14

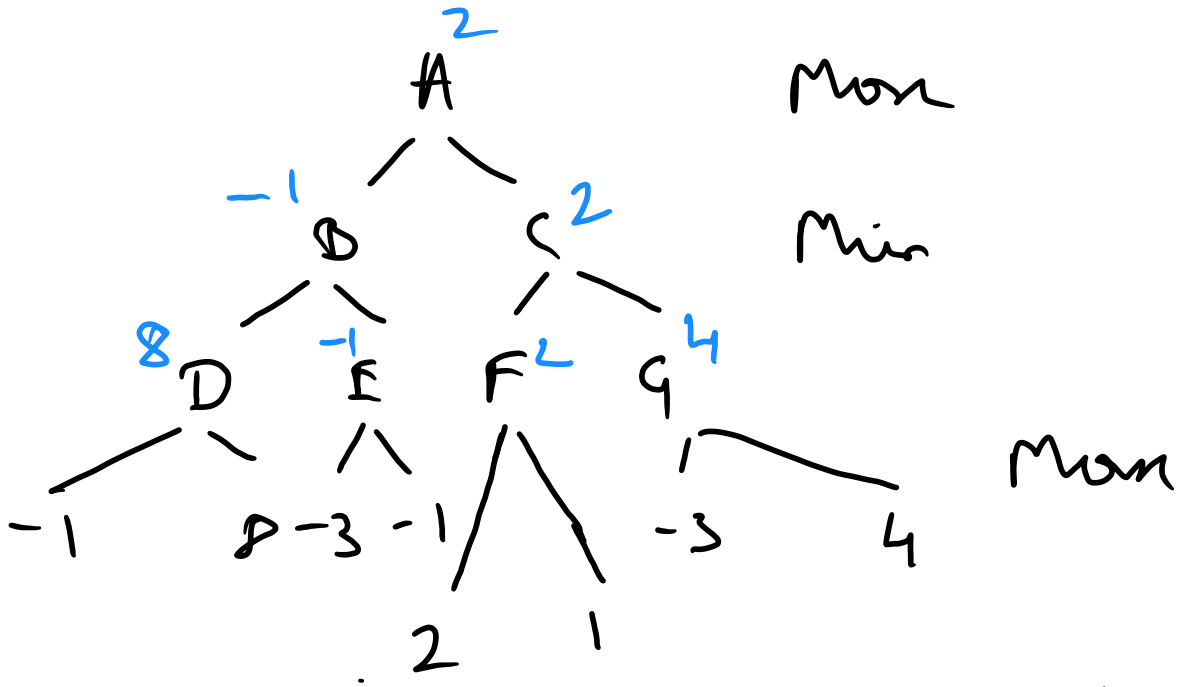
A → F → G → I → J

F → G = 6

Total cost = 3 + 1 + 3 + 3 = 10

* Min max algorithm -

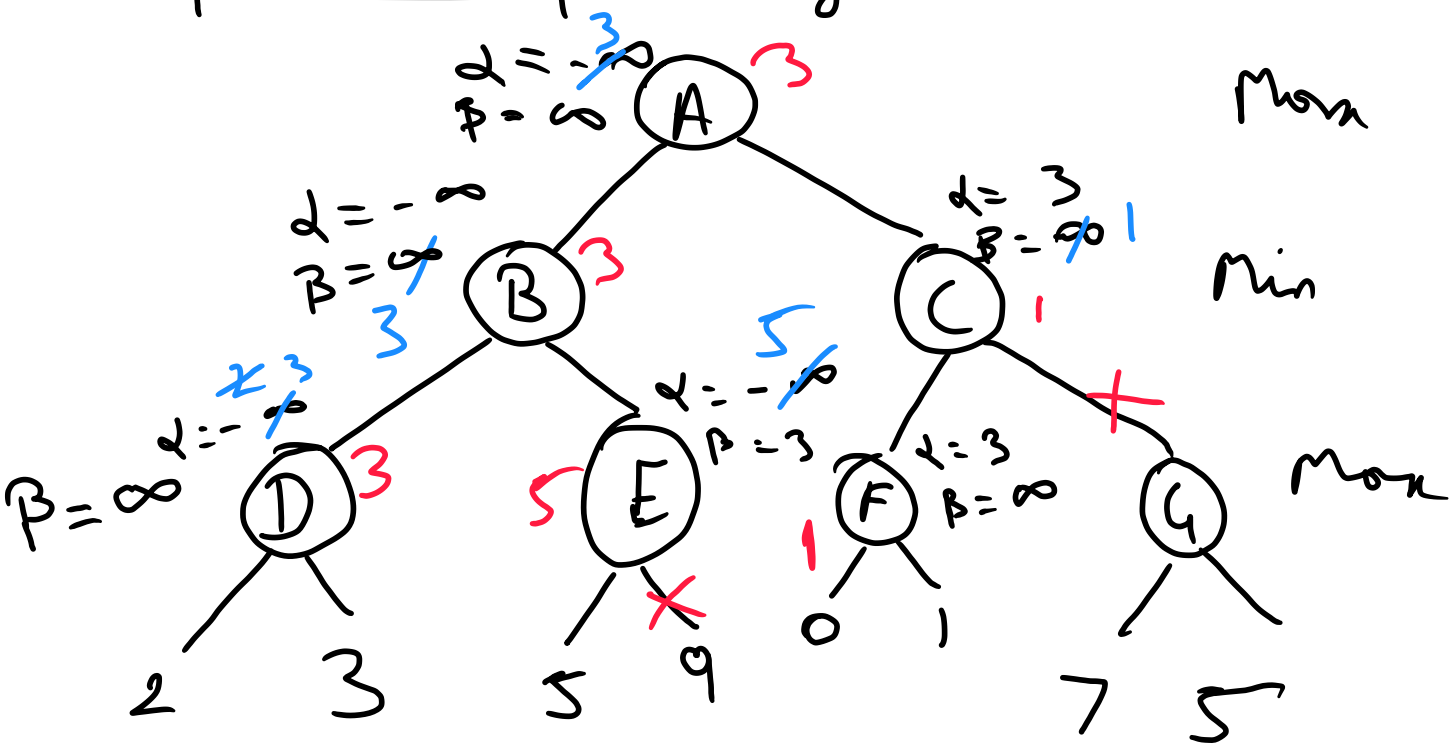
- Backtracking
- best move strategy
- Max will try to maximise its utility
- min will try to minimize its utility
- Mostly used to implement the 2 players games



Time complexity: $O(b^d)$

b = branching factor
d = depth

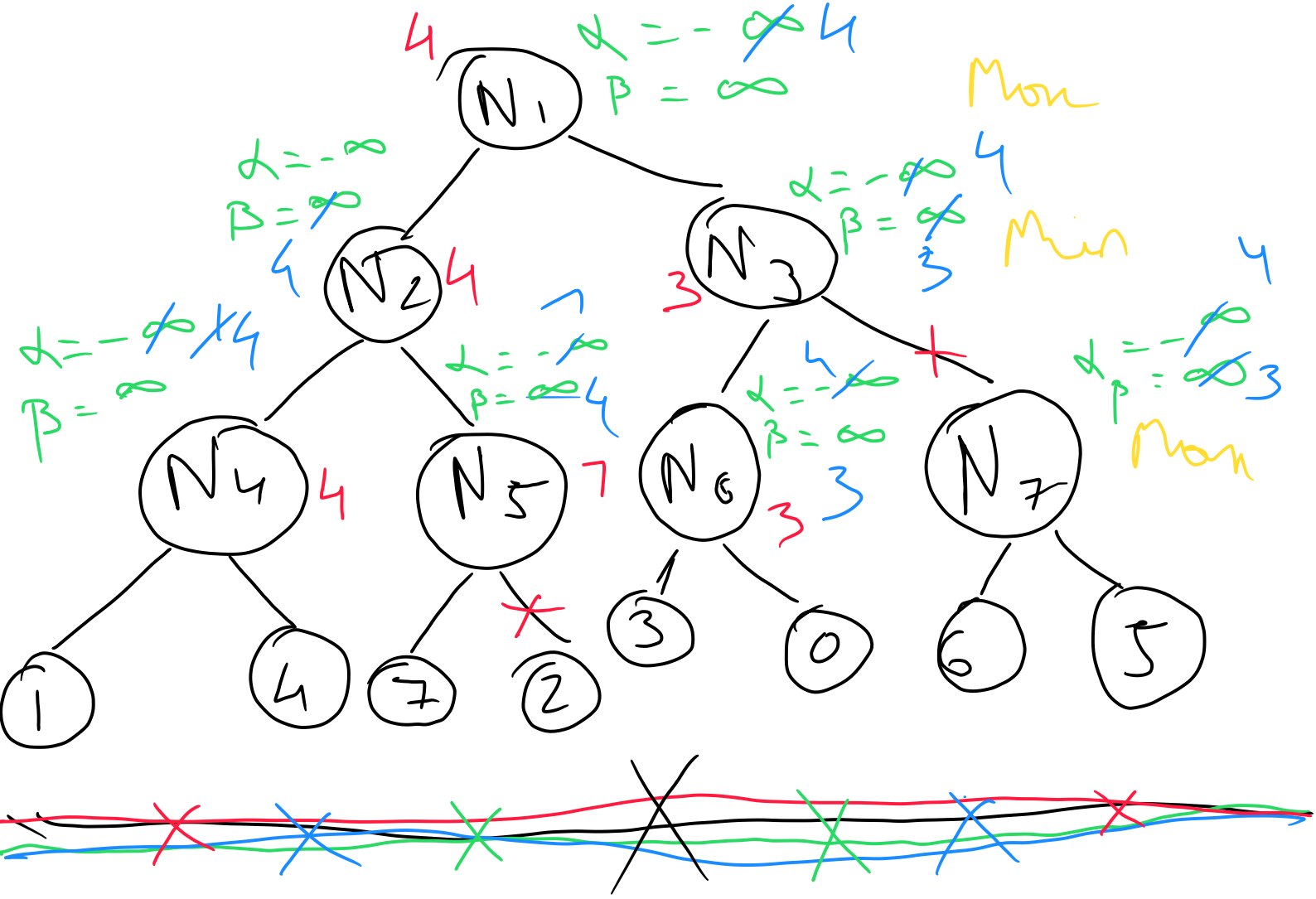
* Alpha Beta pruning -



[if not given, put max 1st]

Pruning condition

$\alpha \geq \beta$



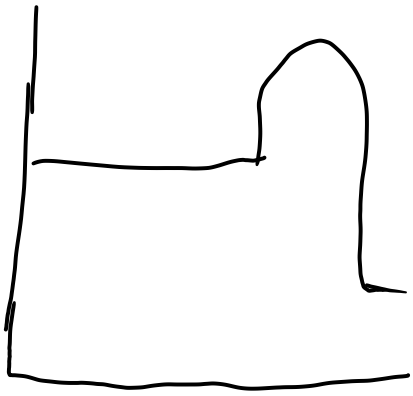
* Hill Climb Algo

1. Evaluate the initial state
2. Loop until a solution is found or there are no operators left
   a) select and apply new operator
   b) evaluate the new state
   c) if goal reach then quit

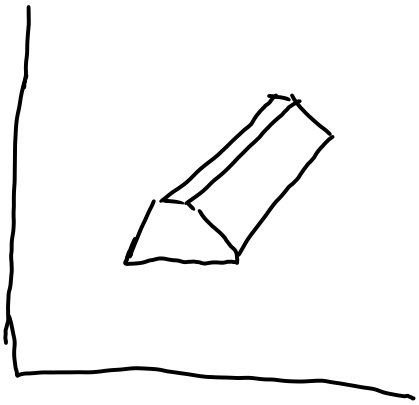
* If a new state is better than the current state, then that is the new current state

- limitations -

local maxima



Plateau



Ridge