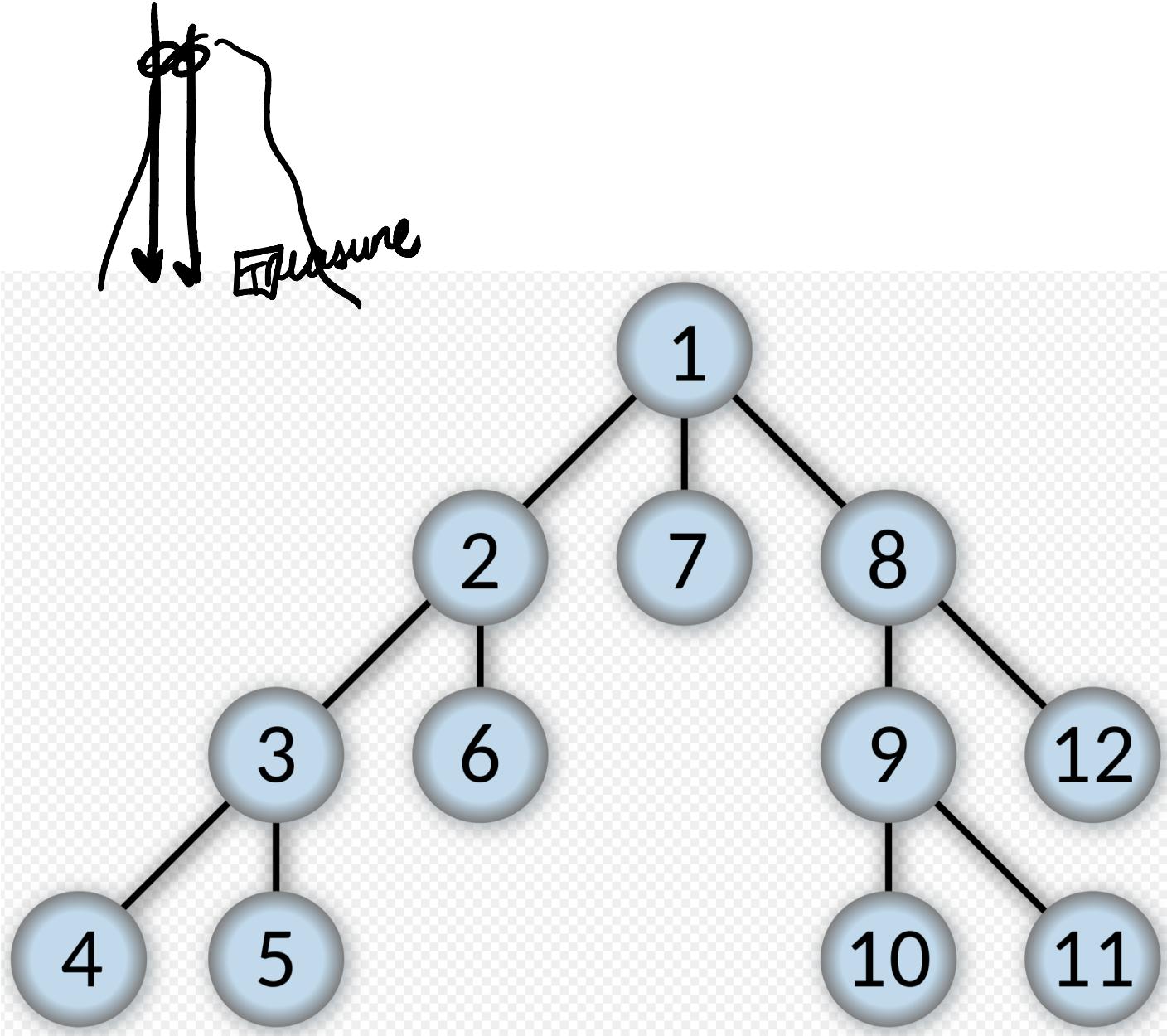


CSCI 3202: Intro to Artificial Intelligence

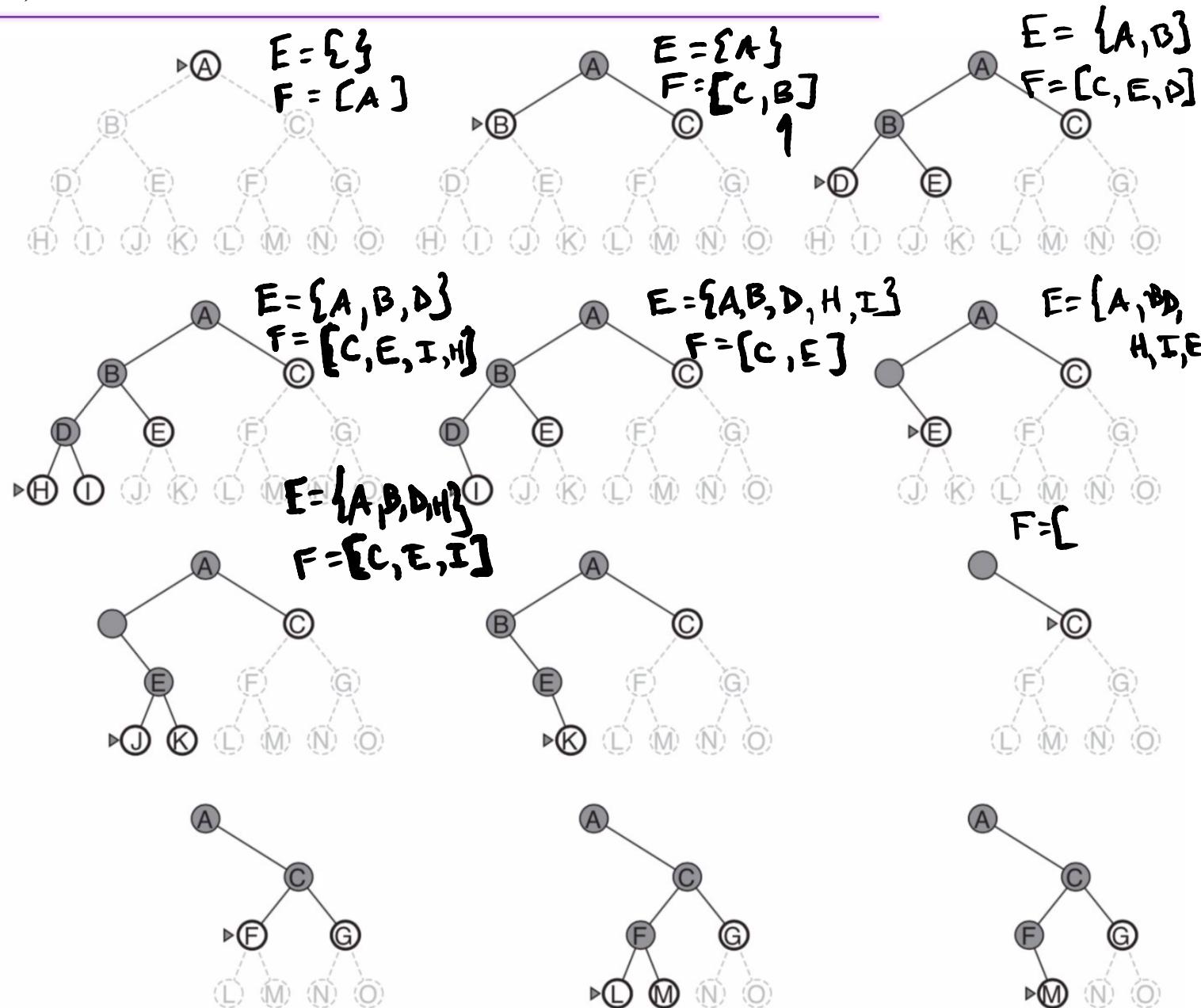
Lecture 5: Depth-First Search (DFS)

Rachel Cox
Department of Computer Science



Depth–First Search (DFS)

- Uninformed
 - Expand deepest node first (LIFO)
 - “Back up” to next-deepest node with unexplored successors



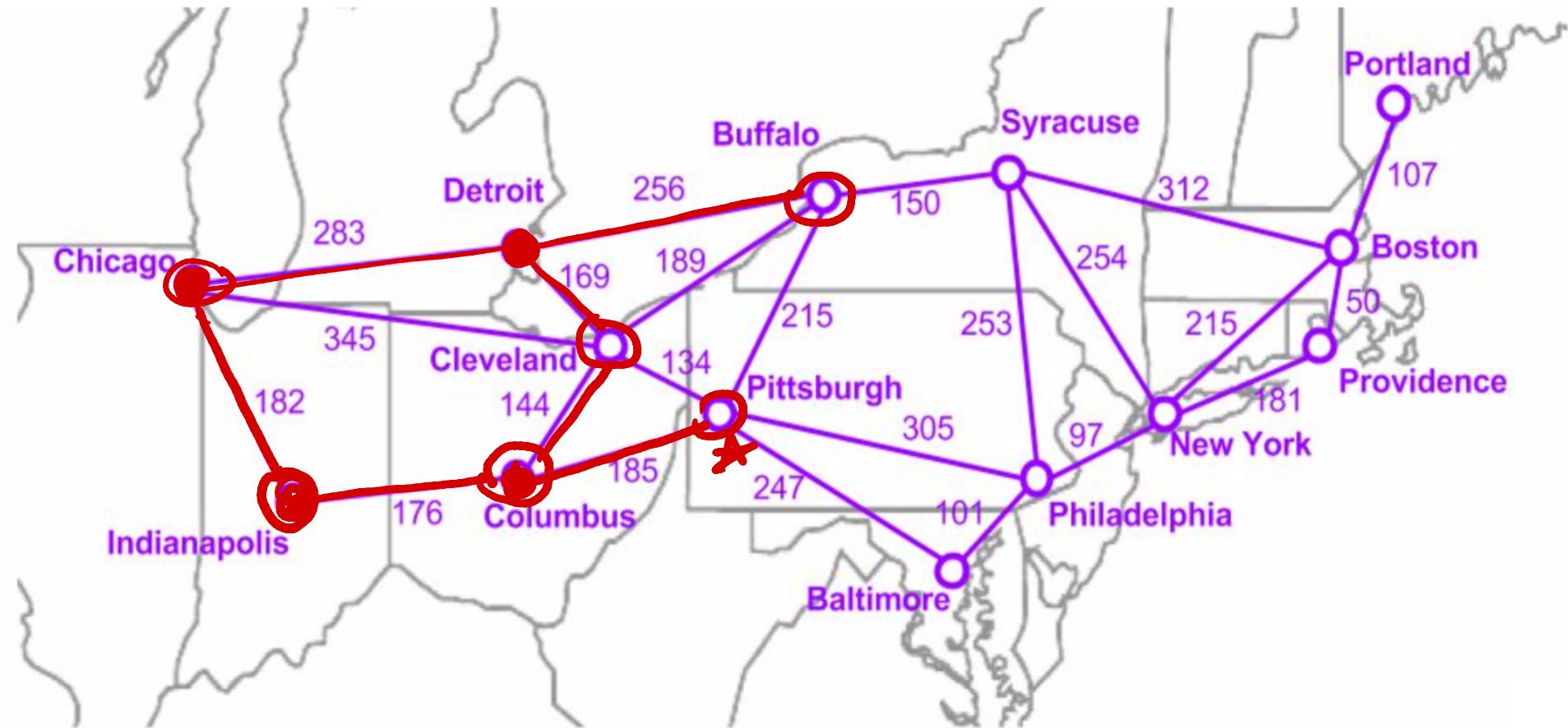
Depth-First Search (DFS)

Example: Traveling in the US northeast

Step costs: miles between cities along major highways

Initial State Goal State
Detroit → Pittsburgh

Detroit → Chicago
→ Indianapolis
→ Columbus
→ Pittsburgh

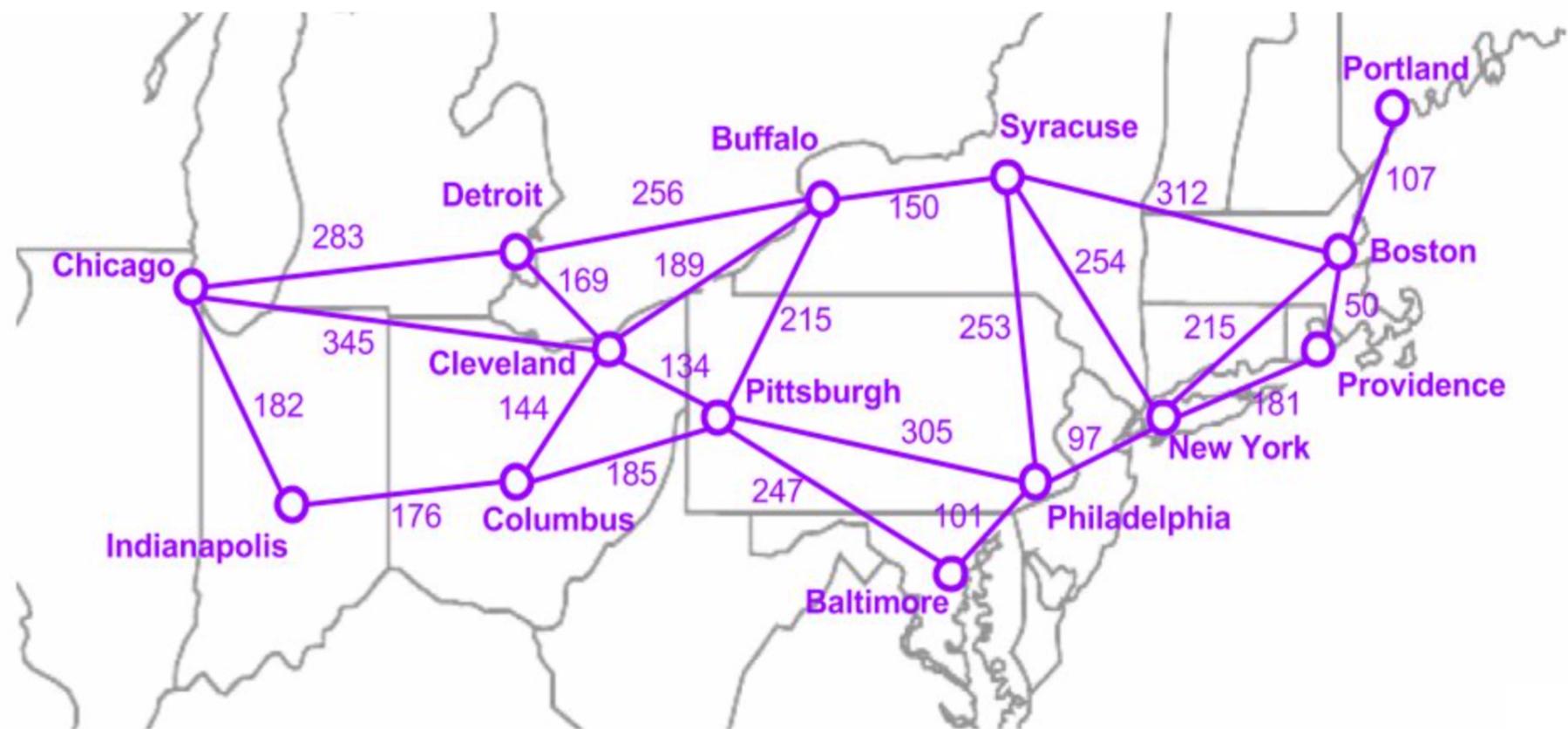


Depth-First Search (DFS)

Example: Traveling in the US northeast. **Question:** Would changing the step cost function change our DFS result?

Step costs: estimated travel time (minutes) along major highways at 5PM east coast time on a Friday

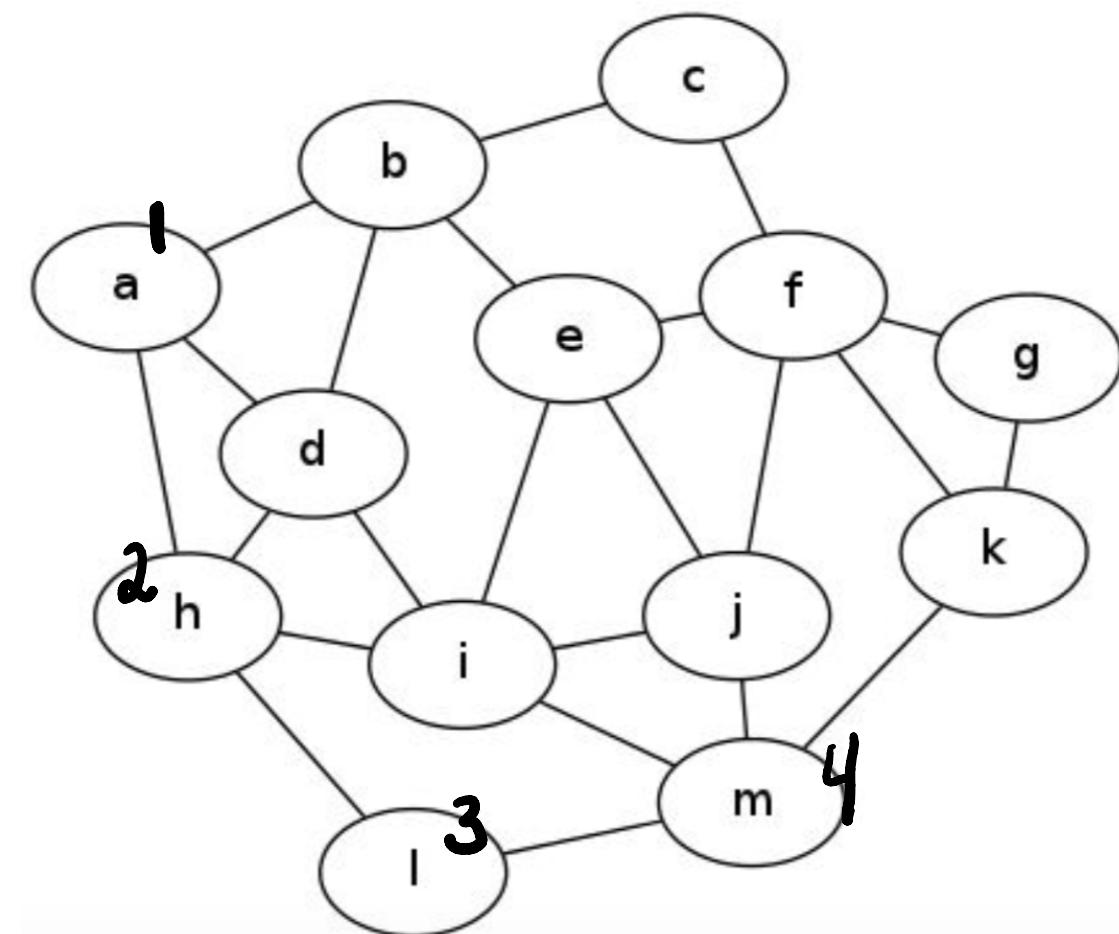
Nope!



Depth-First Search (DFS)

Example: Number the nodes in the search graph according to the order in which they would be expanded using DFS to find a path from a to k . Assume that nodes within a layer are added to the queue by alphabetical order. What is the route that DFS yields, if any?

Explored	Frontier
$\{\}$	$[a]$
$\{a\}$	$[b, d, h]$
$\{a, h\}$	$[b, d, i, l]$
$\{a, h, l\}$	$[b, d, i, m]$
$\{a, h, l, m\}$	$[b, d, i, j, k] \underset{\equiv}{=}$
	$a \rightarrow h \rightarrow l \rightarrow m \rightarrow k$

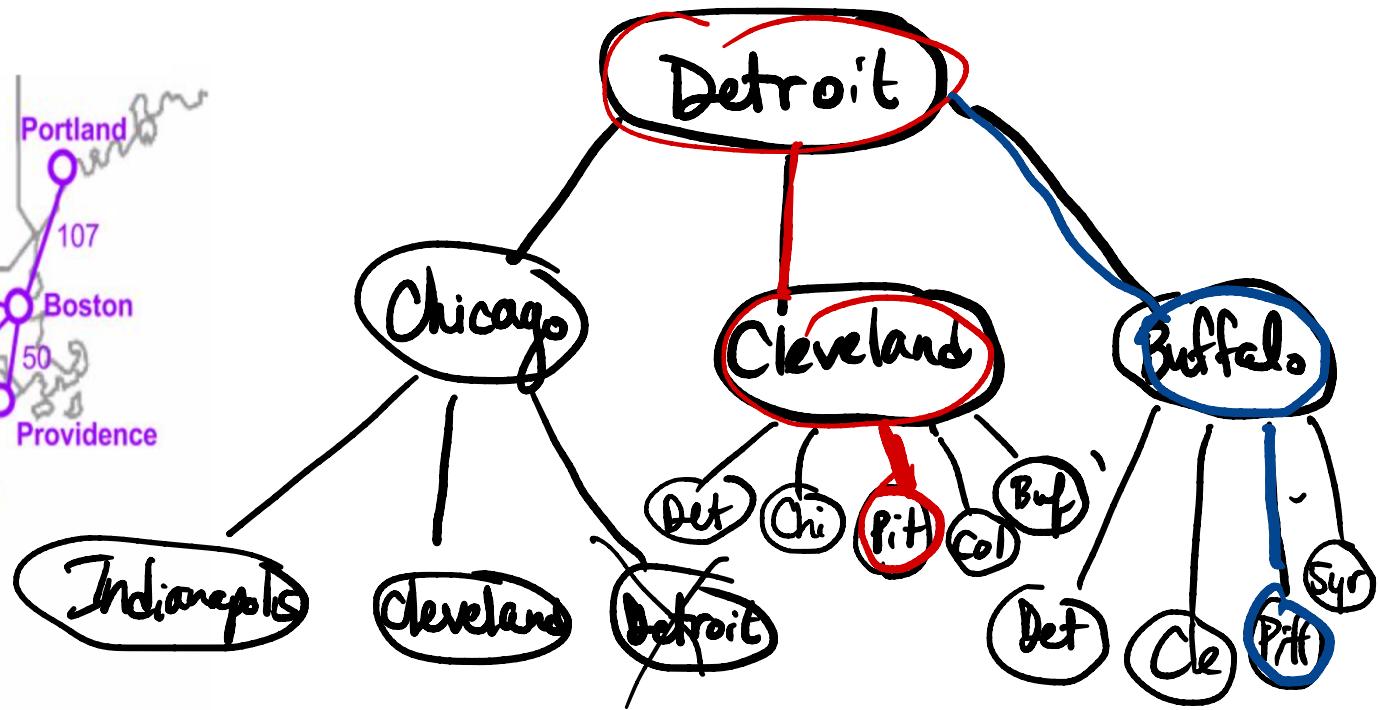
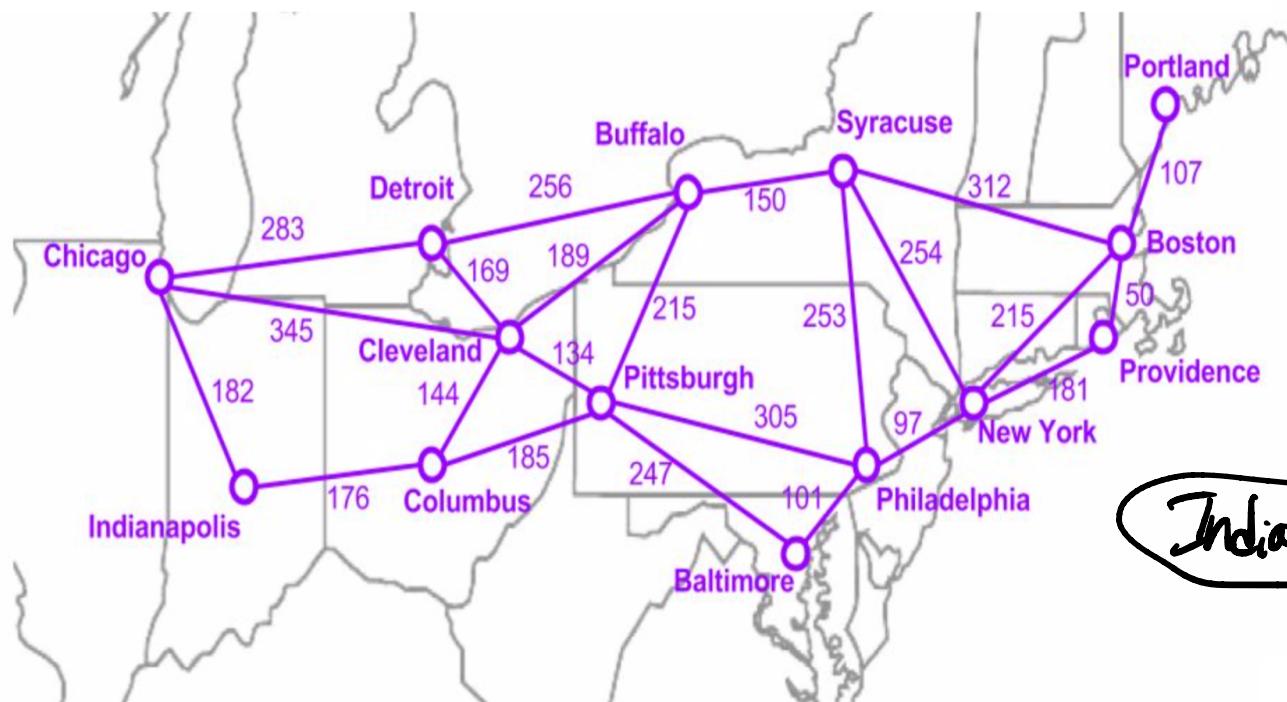


Depth-First Search (DFS)

Start : Detroit

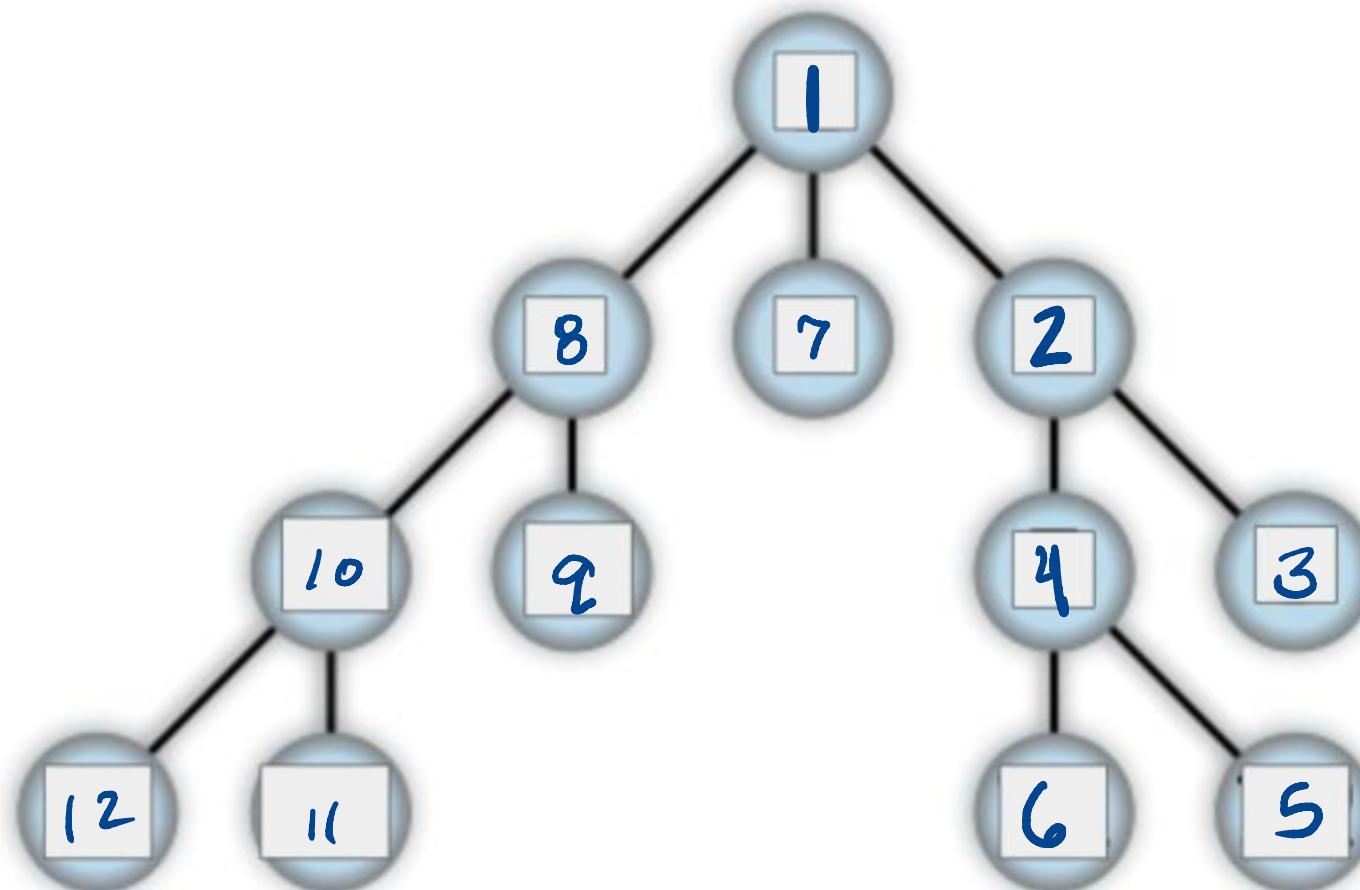
Goal : Pittsburgh

Example: Draw the search tree for the graph below.

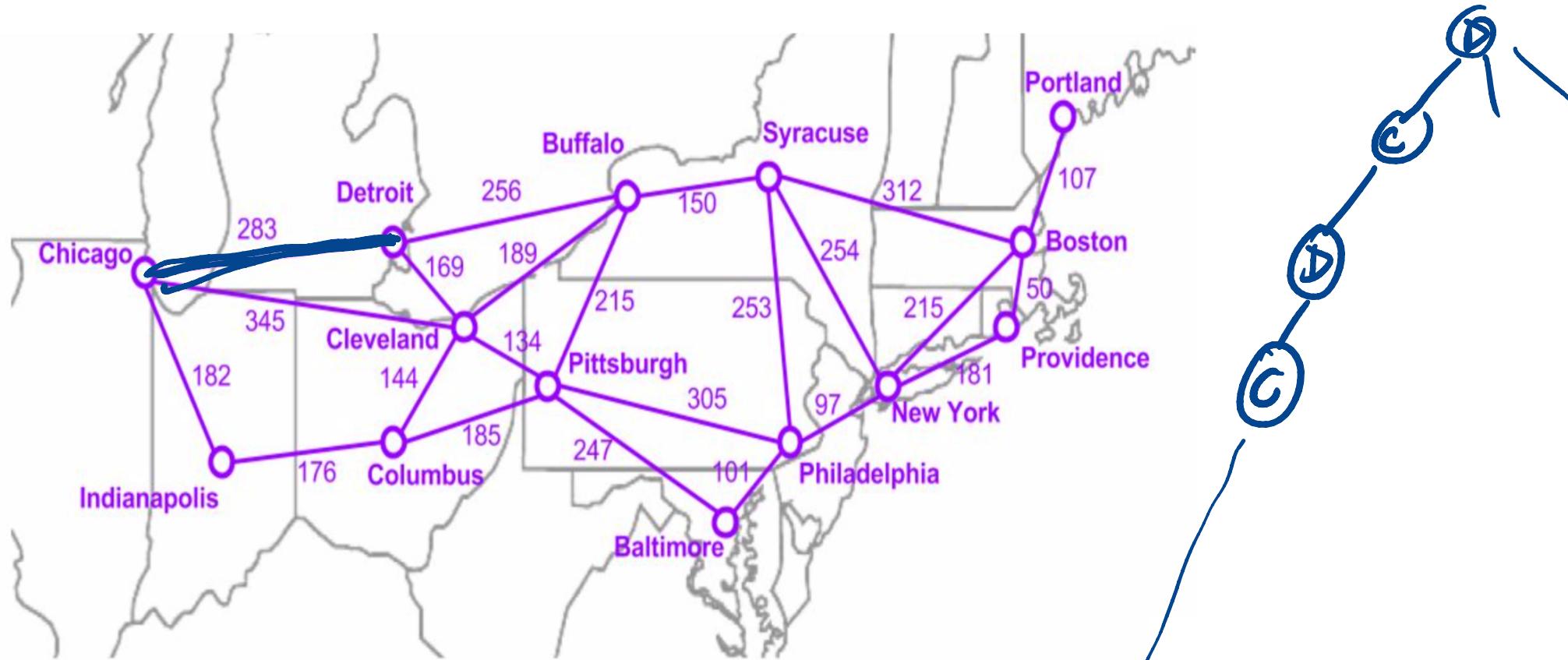


Depth–First Search (DFS)

Example: Number the nodes in the search tree according to the order in which they would be generated using DFS. Assume that the goal is not found, and nodes are added to the frontier stack from left to right.



Depth-First Search (DFS)



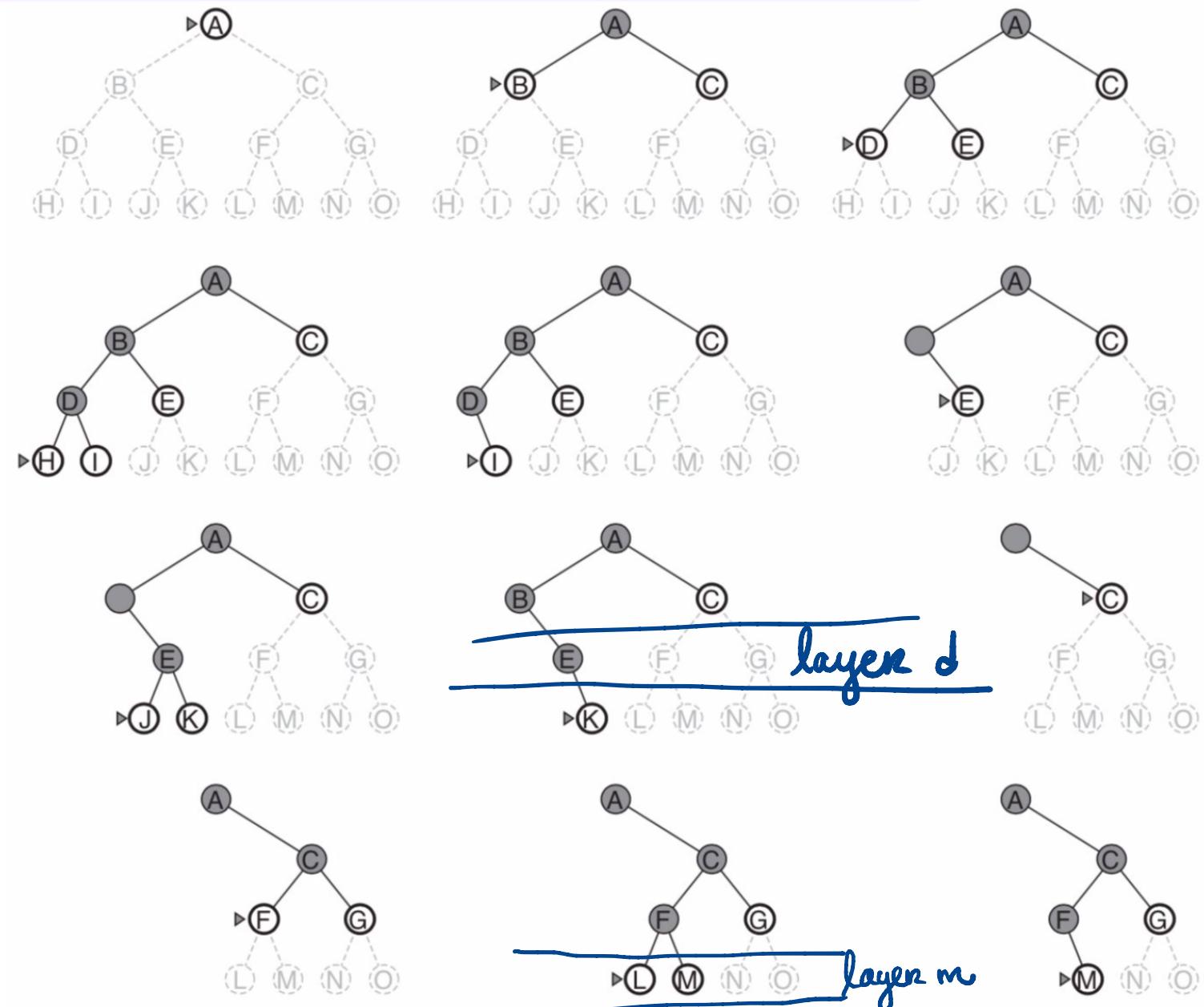
Complete? yes! assuming a finite tree

Optimal? no!

Depth–First Search (DFS)

Time Complexity:

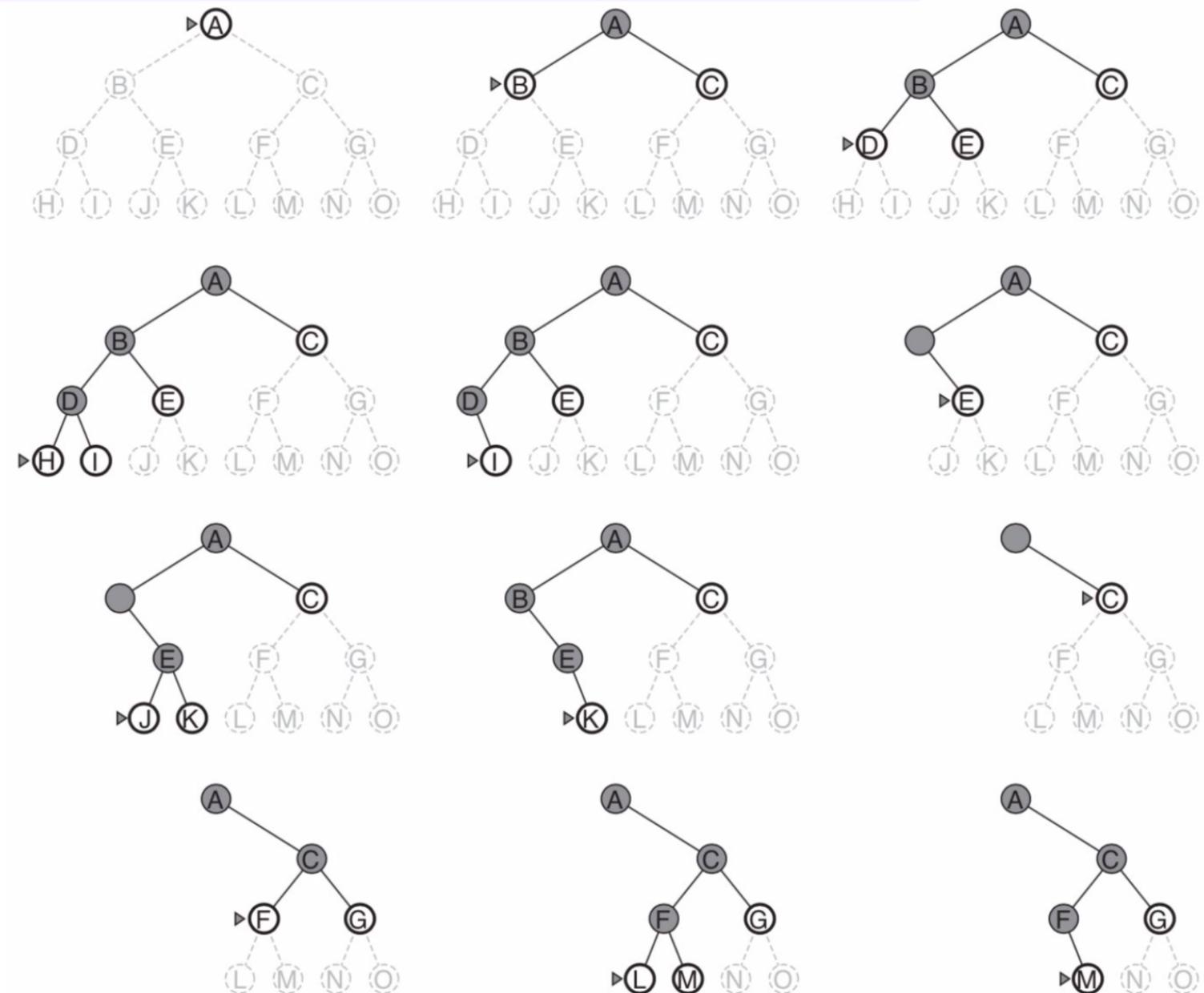
- branching factor b
- maximal depth of m layers
- shallowest goal state in layer d
- might need to generate all b^m states *worst case scenario*
- could be substantially more than just going to shallowest goal state b^d
- total worst case: $\mathcal{O}(b^m)$



Depth–First Search (DFS)

Space Complexity:

- branching factor b
- maximal depth of m layers
- shallowest goal state in layer d
- only need to have one branch expanded at a time: b ... for each of m layers
- total worst case: $\mathcal{O}(mb)$
- Potential failure in infinite state spaces

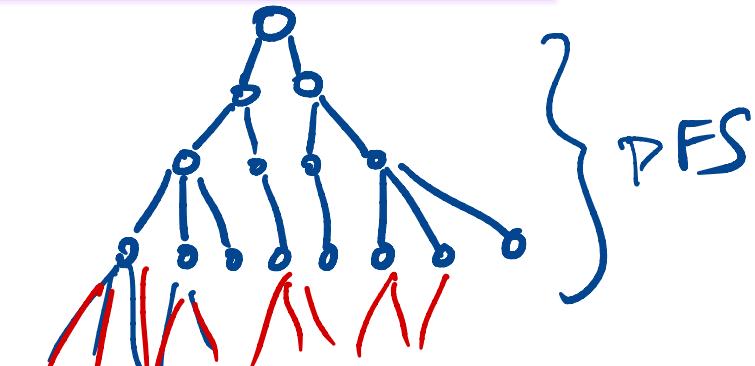


Depth-First Search (DFS)

Offshoots:

Depth limited search

$l = 3$

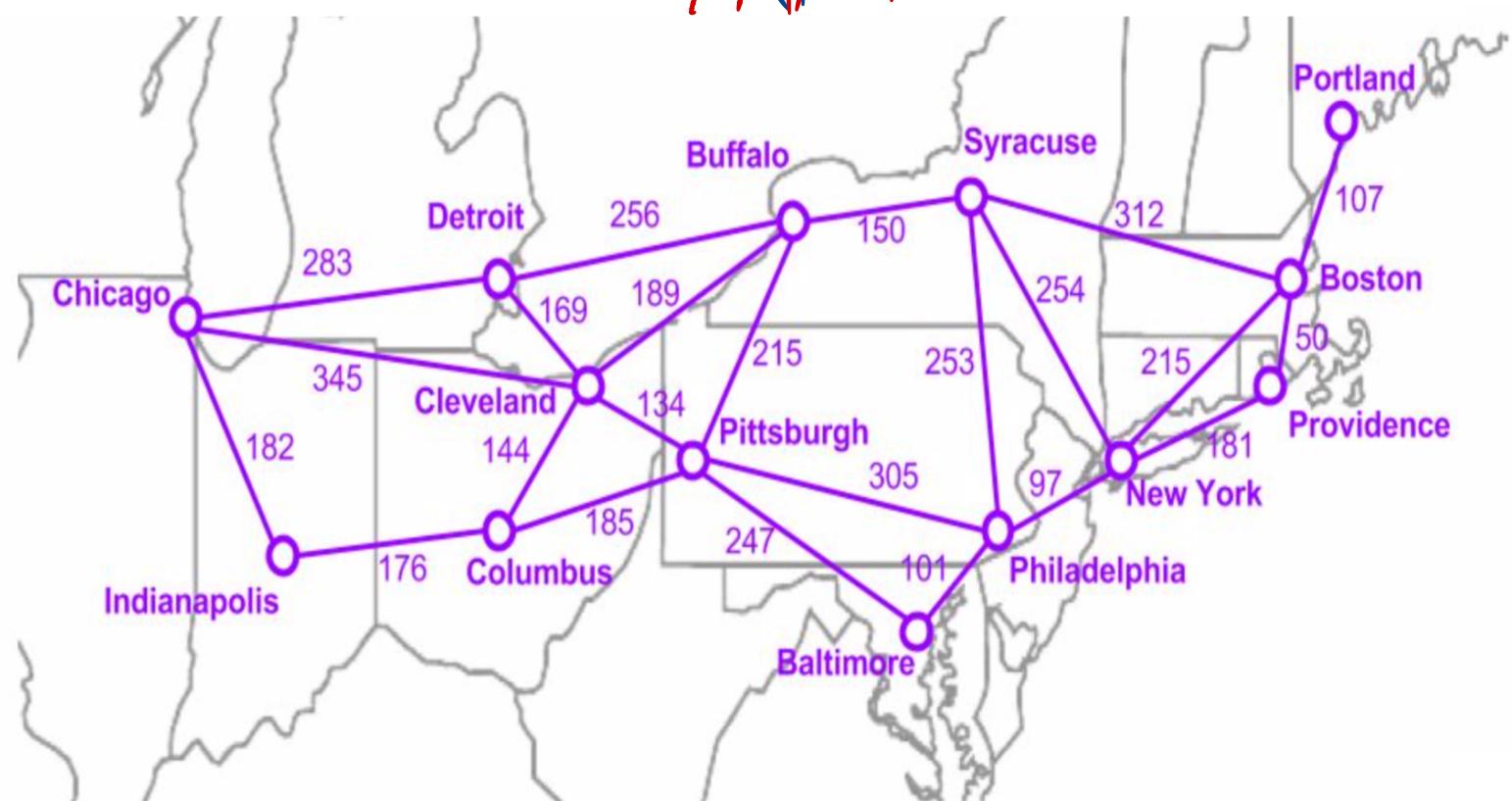


Depth-limited search:

- Search only to maximal depth l

Iterative deepening search:

- Gradually increase l until you find a solution
- Blend of elements of BFS and DFS



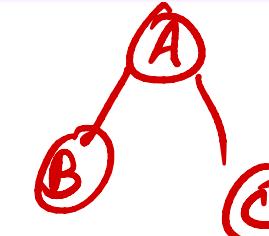
Depth-First Search (DFS)

Iterative deepening search:

- Gradually increase l until you find a solution
- Blend of elements of BFS and DFS

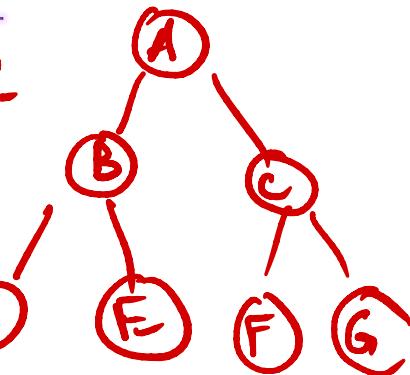
$l=0$ A

$l=1$



$l=2$

DFS

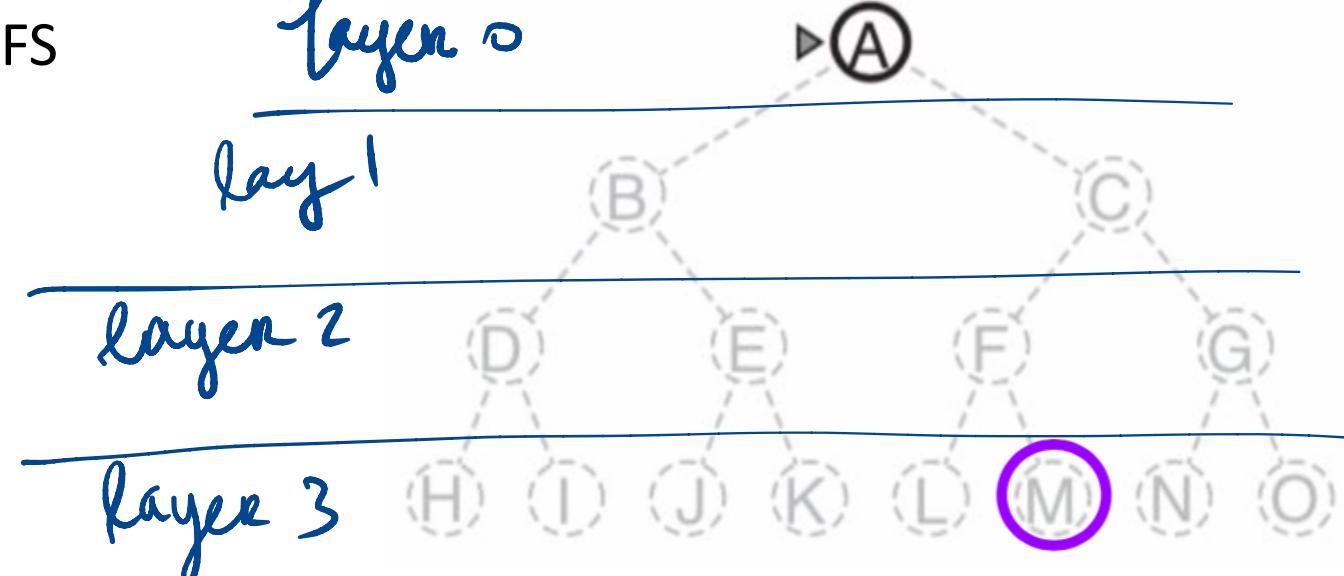


layer 0

layer 1

layer 2

layer 3



Next Time

Uniform-Cost Search (UCS)