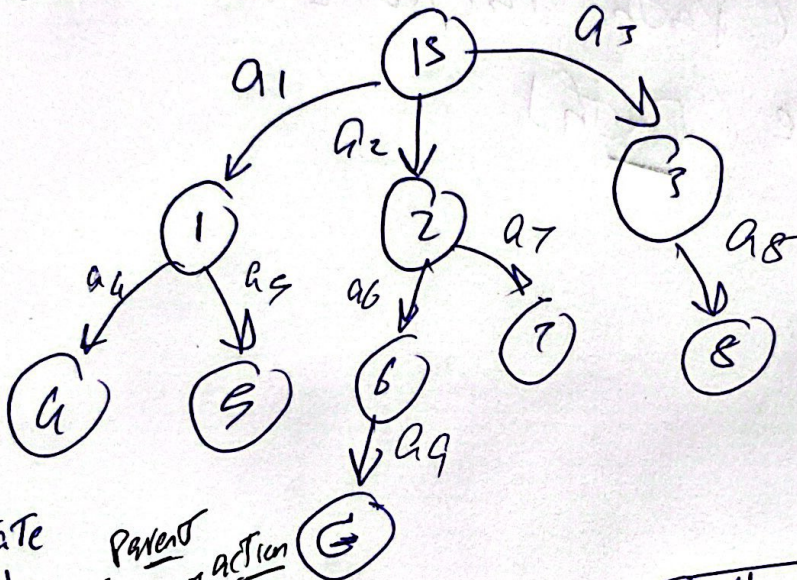


BFS Goal Path Tracker

لو وملت ل goal بتبدأ تعرفه ال Path
 . انت ما مله map ، جواب (action, state)
 وينقل نرجع لحد ما نوجد ال initial state

for each node, define its parent



مراجعة
8/11/23

State Node	Parent	Action
1	(15, a1)	
2	(15, a2)	
3	(15, a3)	
4	(1, a4)	
5	(1, a5)	
6	(2, a6)	
7	(2, a7)	
8	(3, a8)	
9	(6, a9)	

15 1 2 3 ...

Path to the action
 (a9 a6 a2) this is the
 reversed sequence of actions

the initial sequence is
 (a2 a6 a9) #

Path Tracker
 Dict
 14

BFS with Path Tracking Algo (problem, initial state)

1. state = initial state
2. if Problem. is goal (stat)
 1. return { }
3. Frontier Queue, initialize it with the state
4. Explored map or set, to contain the previous visited states
5. Path Tracker dictionary $\langle \text{state}, \text{Pair}(\text{Parent}_{\text{state}}, \text{Action}) \rangle$
6. In Queue map or set, to mark the states inside the Queue.
7. While (Frontier Contains states)
 1. Extract the front state from frontier \rightarrow newState
 2. mark it out of the InQueue Map
 3. mark it as explored.
 4. actions = Problem.get-actions (newState)
 5. iterate over each action in actions.
 1. child = Problem.get-successor (newState, action)
 2. If this child was not in explored or inQueue.
 1. If Problem. is goal (child)
 1. return solution (pathTracker, child, act.)
 2. insert the child in the frontier
 3. mark it as inQueue map
 4. Path Tracker [child] = (Parent, action)
 \downarrow
new state
8. return None # No Found 4

Solution (Path Tracker, state, action)

1. Path = Empty list.
2. Path.add (Path Tracker (state).action)
3. While (state != IS)
 1. node = Path Tracker (state).parent
 2. Path.add (Path Tracker (state).action)
4. return reverse (Path)