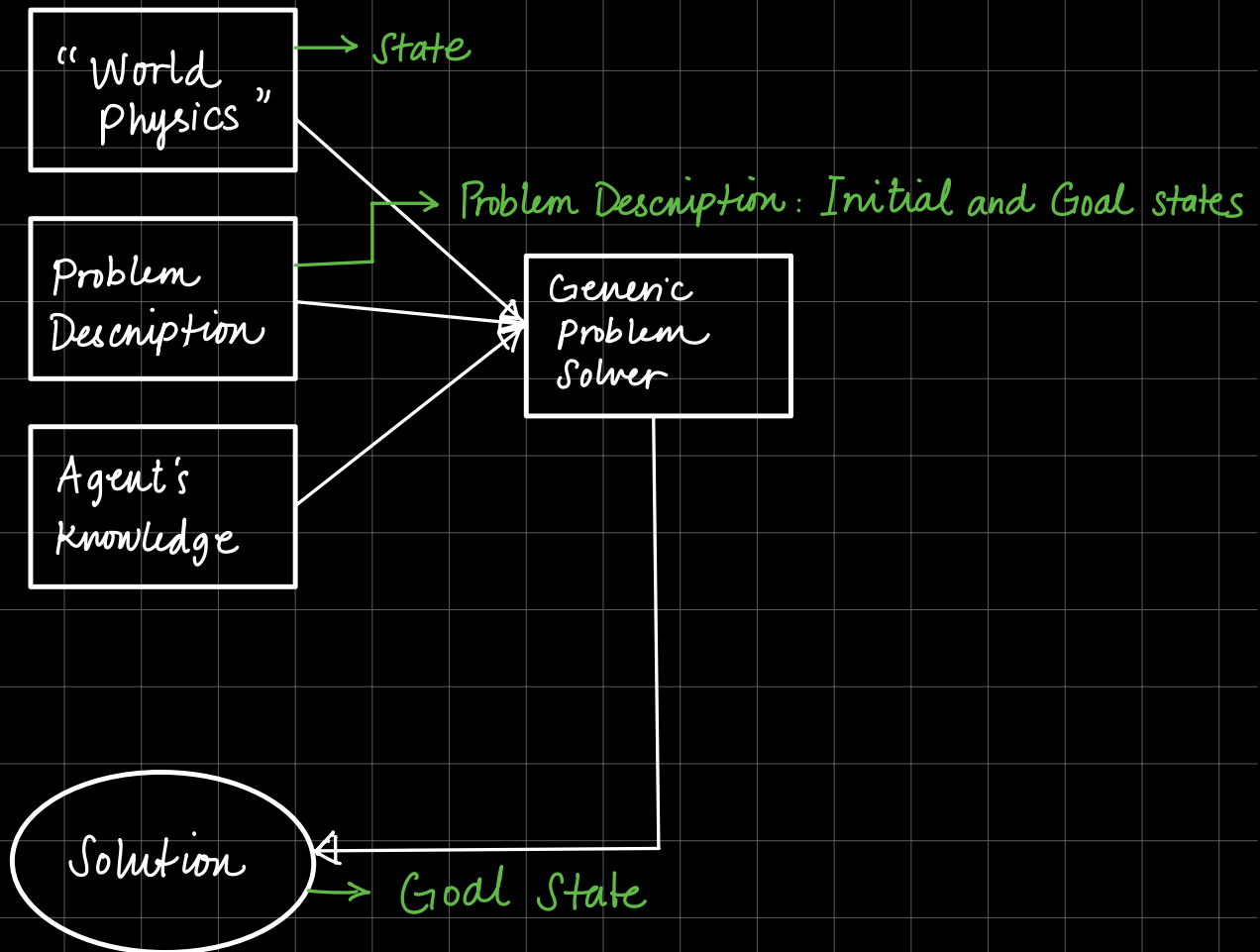


# Generic Problem-Solving Engines



# Eight Puzzle as Graph Search

## Problem Statement:

Find a sequence of operators that will produce a goal state if executed in the initial state.

1	2	3
4		5
7	8	6

Initial State



1	2	3
4	5	6
7	8	

Goal State

State: an array representing tile positions

Actions: move a single till into an empty space

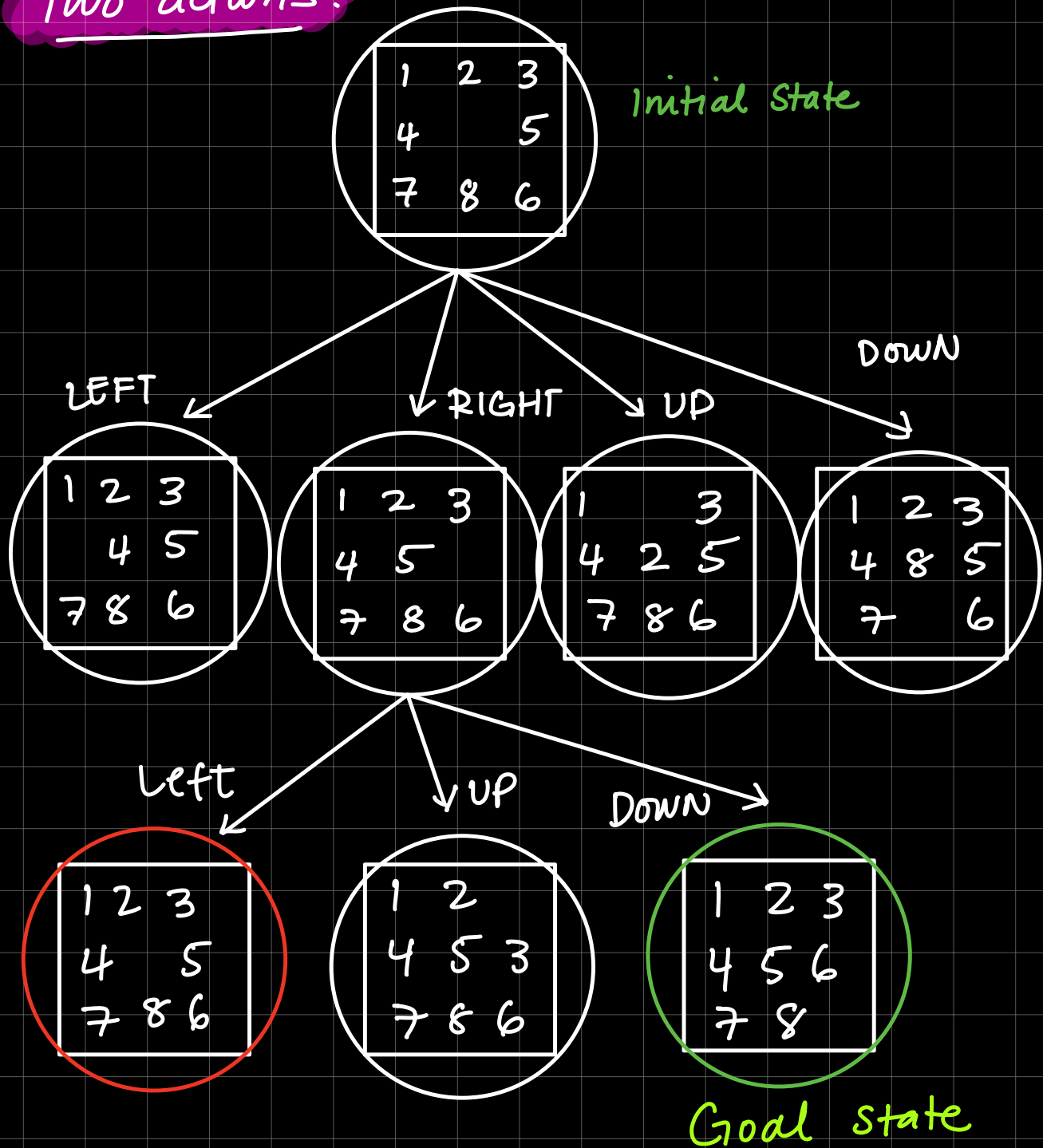
## Problem Description:

Initial: a state (board array)

Goal: a state (board array)

Operators: Left, Right, UP, Down

Two actions:



## Generic Search Algorithm:

```
# Returns a solution, or failure
function SEARCH(initial, succ, goal)
  if goal(initial) then return []    # Trivial solution
  # Node contains both a state and seq of actions
  node ← (initial, [])
  # Fringe is all nodes unexplored so far
  fringe ← {node}
  # Explored is set of states already seen
  explored ← {initial}
  loop do
    if EMPTY?(fringe) then return failure
    # States on the fringe guaranteed not to be a goal
    node ← CHOOSE-NEXT-NODE(fringe)
    explored += node.STATE
    for each (newstate, action) in succ(node.STATE) do
      if newstate in explored: next
      if goal(newstate) return (node.ACTIONS + action)
      fringe += (newstate, node.ACTIONS + action)
```

Cost Model:  $c([a_1, a_2, \dots, a_n]) = c(a_1) + \dots + c(a_n)$

