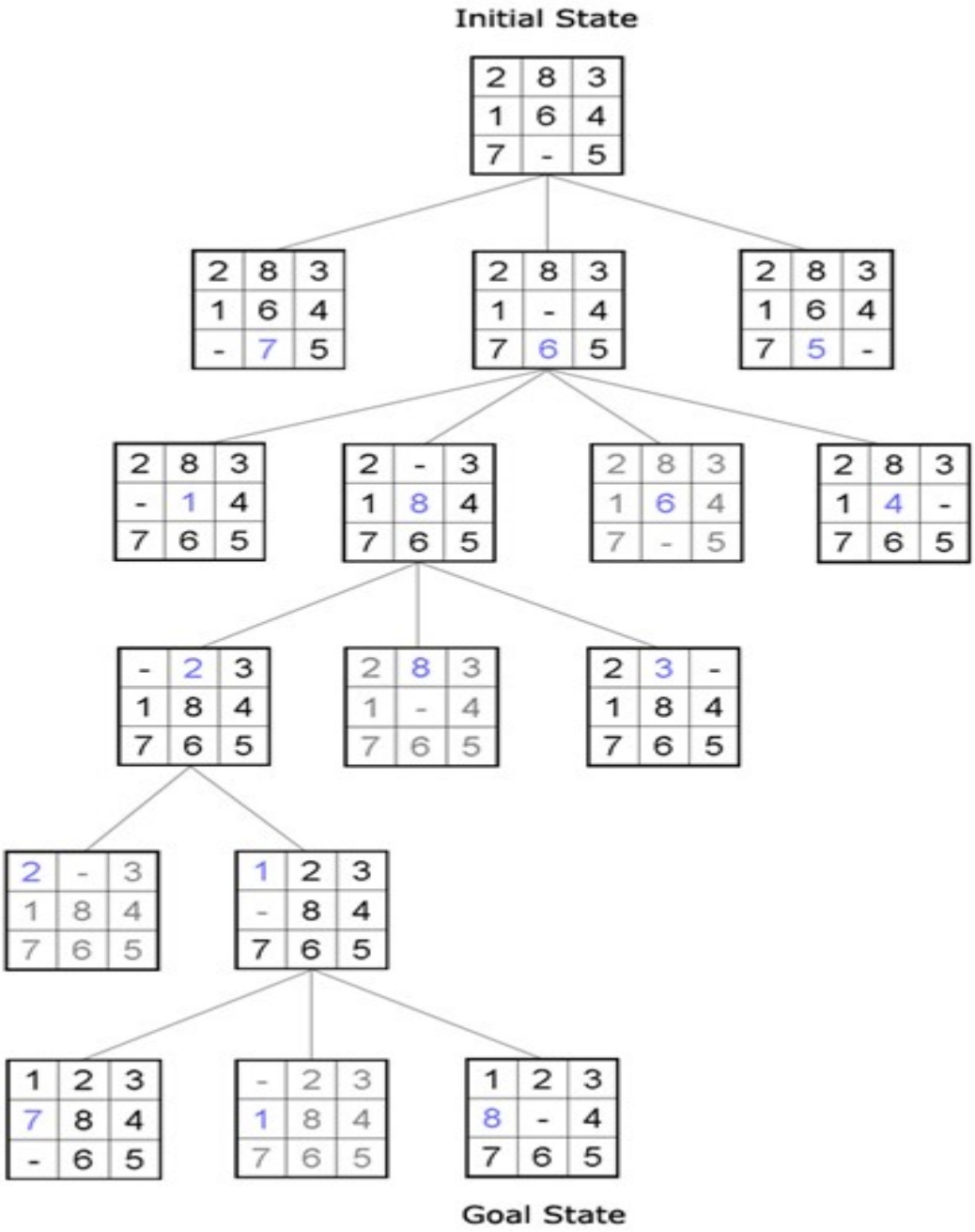
8 Puzzle Game using DFS



-First we find all possible solution from start state to the goal state(State Space)
-Second we draw that State Space as tree graph , its root is the start space
-Third we use Depth First Search to explore the tree from its root until we find the goal

-These are the steps of DFS:-

*Explore deeper in the tree whenever possible
*Edges are explored out of the most recently discovered vertex V that still has unexplored edges
*when all V's edges has been explored , backtrack to the vertex from which V was discovered

-Main points in the code :-
*we represent each state of the puzzle as list from 9 positions , the position with the 0 value represent the empty box
*we represent the possible transition from state to another with the fact move(OldState,NewState).
*we represent the state space , with the move facts

*after that we implement the DFS/4 rule which take 4 parameters
1-StartState, 2-GoalState
3-EmptyCheckList “we put in it the visited states in order to avoid loops”
4-Path which will be a list contains the path from StartState to the GoalState
* we also implement the DFS/3 with 3 parameters to be interface with user , which in-turn call DFS/4

*DFS steps :-
-find possible transition from StartState to anther State call it State2
-check if these State2 is in the visited list
true , go back and find anther State2
false , add the state2 to the check list then continue
-make State2 as StartState and try DFS to find the goal from that new state
-the algorthim reach the end when we the StartState become the GoalState , that means we find the goal ,

*we implement showPuzzleState/1 rule which take list”State” and try to display it
*we implement showPuzzleSolution/1 which take the path we find and try to display it be calling showPuzzleState on each item