LAB 1: Implementation Of Toy Problem

8 - puzzle Problem

Problem Formulation :

The 8 puzzle consists of eigth numbered, movable tiles set in a 3 × 3 frame. One cell of the frame is always empty thus making it possible to move an adjacent numbered tile into the empty cell.

The initial & the goal states are given.

The problem is to change the initial configurations to the final one, by moving the tiles one by one in the single empty ~~state~~ space, thus achieving the goal conf.

Problem Solving

Instead of moving the tiles in the empty space, we can visualize moving the empty space in~~to~~ place of the tile. The empty space can move ~~no~~ only in 4 directions : 1) Up 2) Down 3) Right 4) Left

* The empty space cannot move diagonally & can only take one step a time.

Methods to solve the problem : a) Uninformed searching
b) Informed searching

Heuristic search is an informed search technique

A heuristic value tells an algorithm which path will provide the solution as early as possible.

A* algorithm

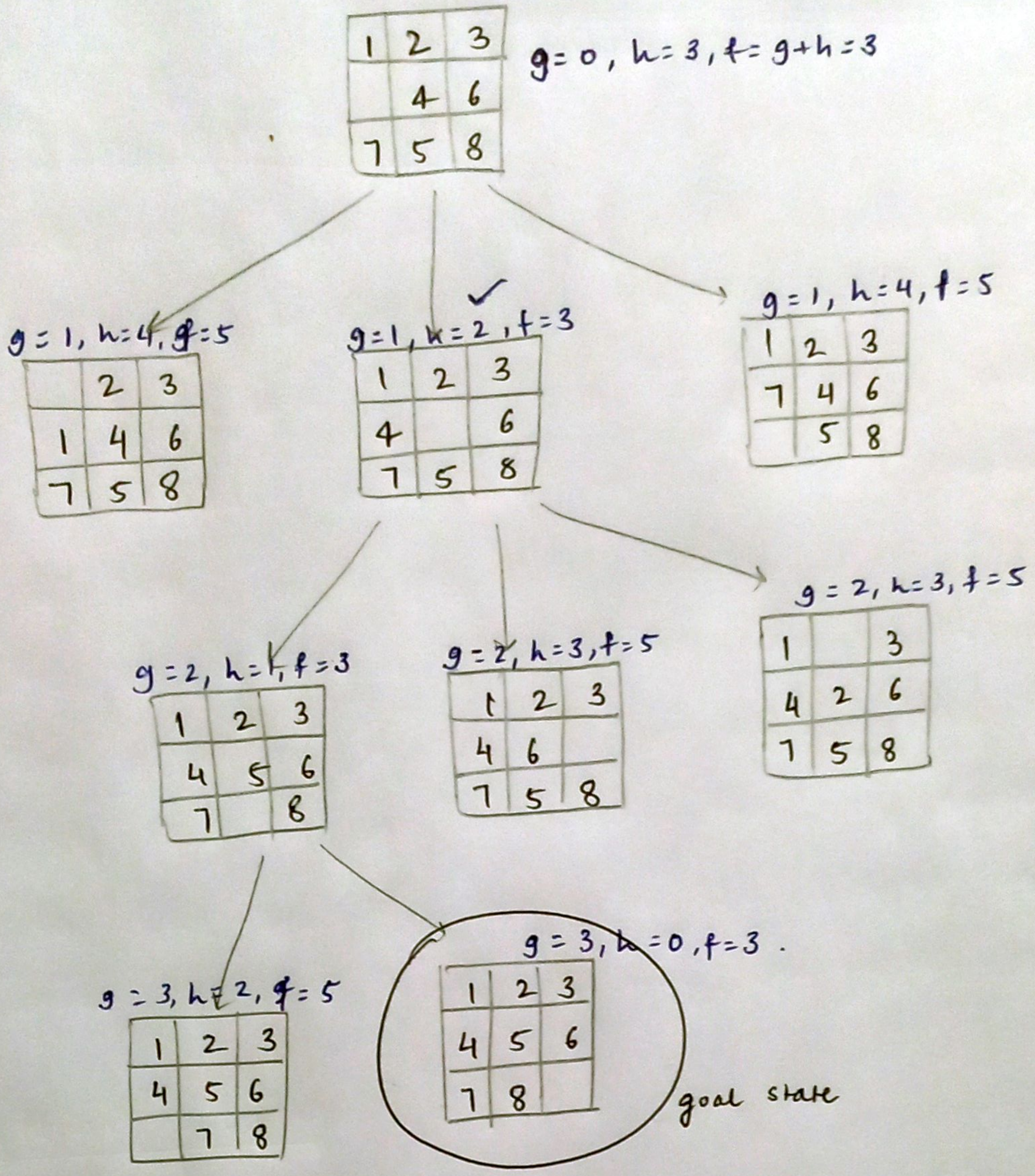
This algorithm keeps a track of each visited node which helps in ignoring the nodes that are already visited, saving a huge amount of time.

A* uses a combination of heuristic value h-score as well as g-score.

$$f(\text{score}) = \underset{\substack{\downarrow \\ \text{estimation} \\ \text{cost from} \\ n \text{ to goal node}}}{h(\text{score})} + \underset{\substack{\downarrow \\ \text{actual cost} \\ \text{from start} \\ \text{node to } n}}{g(\text{score})}$$

We define the h-score as the no. of misplaced tiles by comparing the current state to the goal state or summation of Manhattan distance b/w misplaced nodes.

g-score will remain as no. of nodes traversed from a start node to get to the current node.



Algorithm.

- It maintains 2 list : OPEN and CLOSE
- OPEN contains those nodes that have been evaluated by the heuristic func, but haven't been expanded yet.
- CLOSED contains the nodes already visited.
- Define a list OPEN, contains start node S
- Remove node n with smallest value of f(n) from OPEN and move it to CLOSED.
- If node n is goal state return success & exit.
- Expand node n
- If any successor to n is goal, return success.