

A* algorithm.1. Initialise:

- Start with the initial state of puzzle
- Set goal state.

2. Queue

To store the states of puzzle

$$f(n) = g(n) + h(n)$$

$g(n)$: is the no of moves taken from start.

$h(n)$ → misplaced tiles heuristic
which counts how many tiles
are not in their goal positions

3. States:

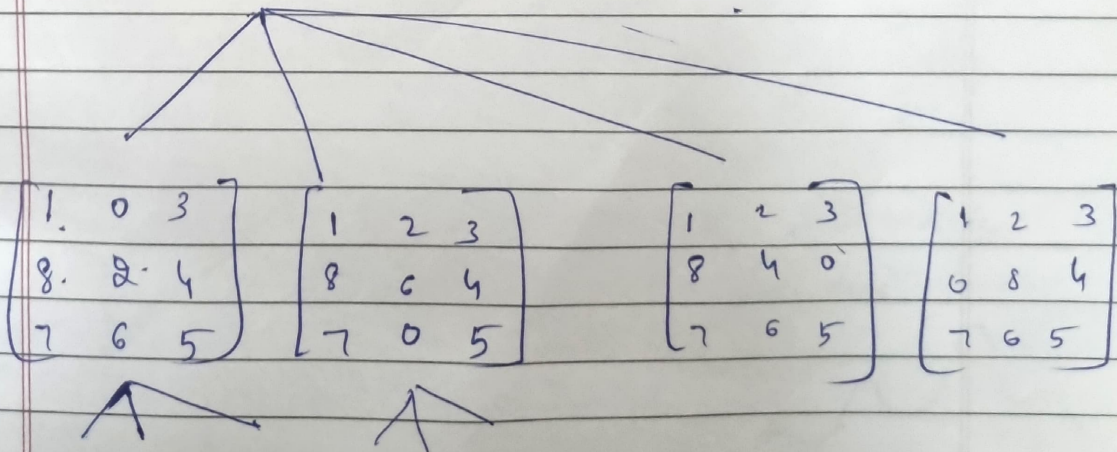
- If the goal state is achieved then return goal state achieved
- if the goal state not achieved then given out the possible moves.

Initial state

$$\begin{bmatrix} 1 & 2 & 3 \\ 8 & 0 & 4 \\ 7 & 6 & 5 \end{bmatrix}$$

Goal State.

$$\begin{bmatrix} 2 & 8 & 1 \\ 0 & 4 & 3 \\ 7 & 6 & 5 \end{bmatrix}$$



Pseudocode

- 1) initialize Prior queue with initial State
Set $g(n) = 0$
Set $h(n) = \text{no of misplaced tiles}$
Set $f(n) = g(n) + h(n)$
- 2) while Queue $\neq \emptyset$
 - Remove the state with Smaller $f(n)$
 - if 'curr-State' = goal-State
Return solution
 - generate all the new States
- Calculate $g(n)$, $h(n)$, $f(n)$
- 3) If goal reached then return the solution

~~proceed~~