

Lab-03

* A* Algorithm

function A* search (problem) returns a solution or failure

node \leftarrow a node n with n state:
problem.initial state.

frontier \leftarrow a priority queue ordered by ascending $g(n)$ & only elements

Loop do

if empty (frontier) then return failure.

$n \leftarrow \text{pop}(\text{frontier})$

if problem.goalTest(n .state) then return solution

for each action a in problem.actions(n .state) do

$n' \leftarrow \text{childNode}(\text{problem}, n, a)$
insert(n' , $g(n') + h(n')$, frontier)

Lab-3

Output.

Enter 3x3 puzzle board

Row 1 : 1 2 3

Row 2 : 4 0 5

Row 3 : 7 8 6

1 2 3

4 0 5

7 8 6

1 2 3

4 5 6

7 8 6

1 2 3

4 5 6

7 8 0

Manhattan Dist

Tile 1 :

current (0,0)

goal (0,0)

$$\text{Distance} = |0-0| + |1-1| = 0$$

Tile 2 :

current (0,1)

goal (0,1)

$$\text{Distance} = |0-0| + |1-1| = 0$$

Tile 3 :-

current (0,2)

goal (0,2)

$$\text{Distance} = |0-0| + |2-2| = 0$$

Tile 4

current (1, 0)

goal (1, 0)

$$\text{Distance } (1-1) + (0-0) = 0$$

Tile 5

current (1, 1)

goal (2, 2)

$$\text{Distance} = (1-2) + (1-2) = 1$$

Tile 6

current (1, 2)

goal (1, 1)

$$\text{Distance} = (1-1) + (1-2) = 1$$

Tile 7

current (2, 0)

goal (2, 0)

$$\text{Distance} = 0$$

Tile 8

current (2, 1)

goal (2, 1)

$$\text{Distance} = 0$$

Tile 9

current (2, 2)

goal (1, 2)

$$\text{Distance} = (1-2) + (2-2) = 1$$