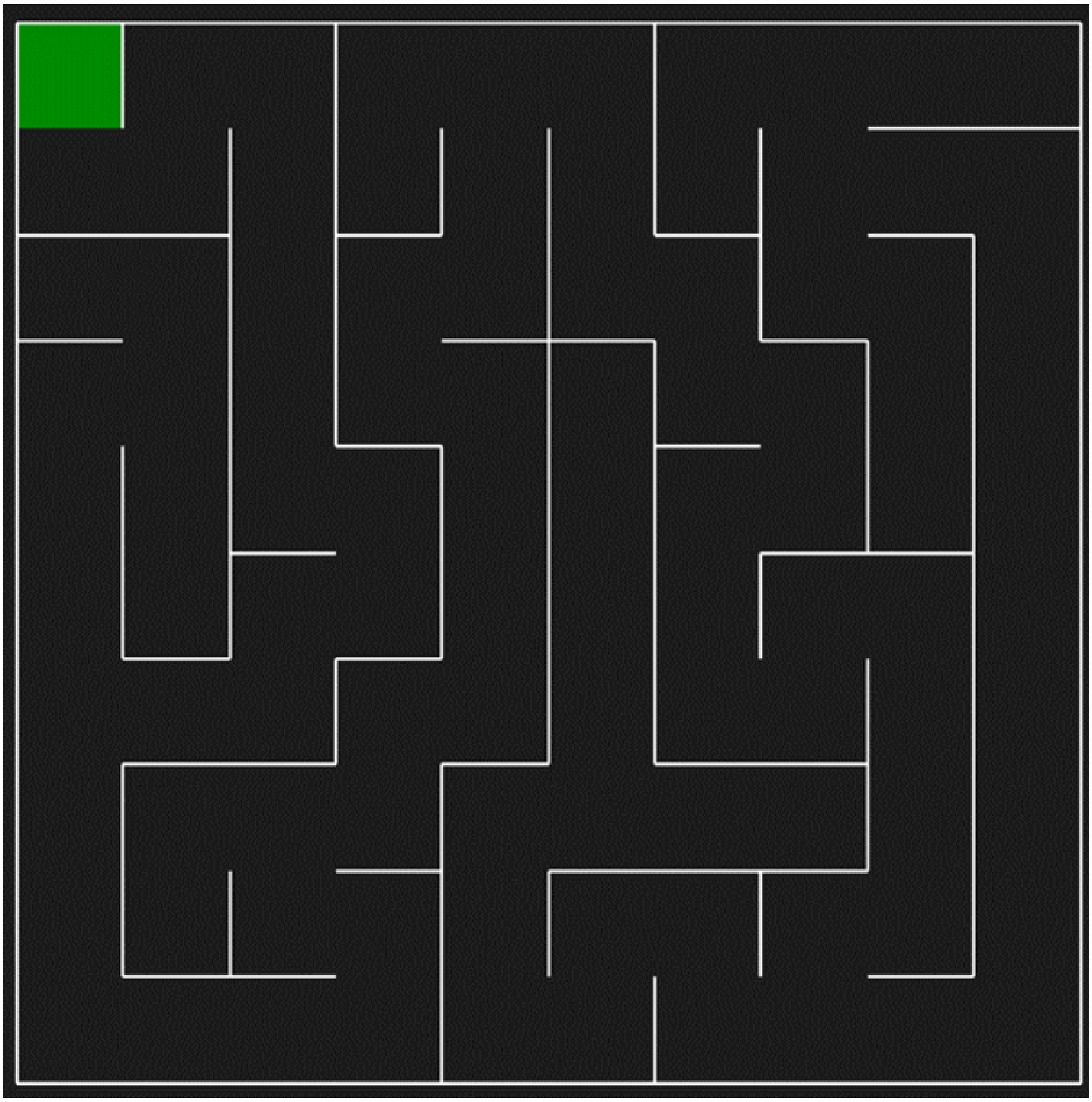
Github link:link

A* Search Algorithm

We consider a 2D MAZE having many obstacles, and we start from the start cell and reach our goal cell/node.

We generate our maze using the python library (pyamaze)

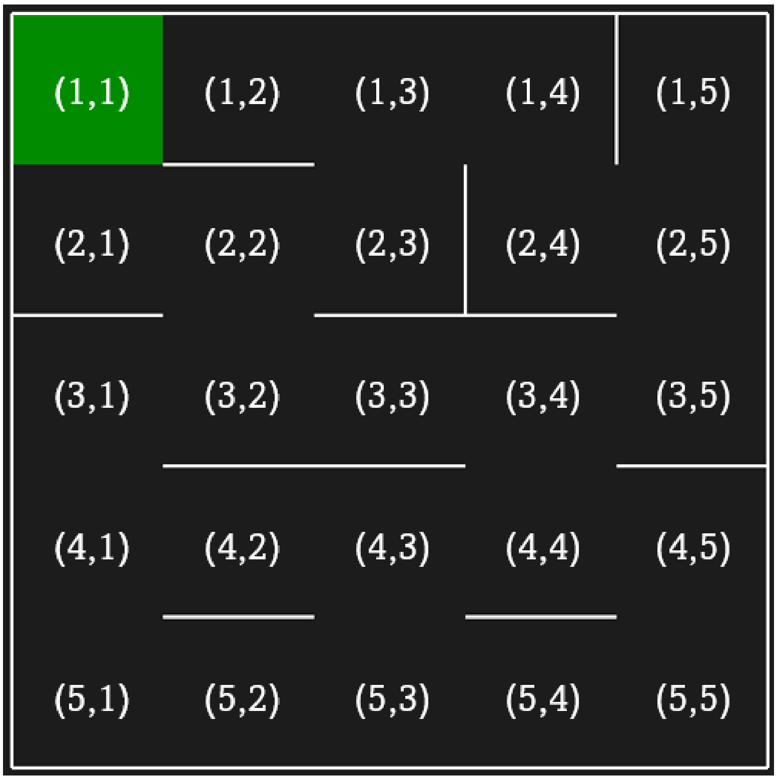
Ex: A 10X10 maze which is generated by pyamaze



Heuristic function:

- I choose Manhatten distance as my heuristic.
- Manhatten's distance = |x1 - x2| + |y1 - y2| , Where (x1,y1) and (x2,y2) are coordinates of our node/cell in a network graph model

Range of coordinates in a maze:
(1,1) till (n,n)
Ex:



(1,1)	(1,2)	(1,3)	(1,4)	(1,5)
(2,1)	(2,2)	(2,3)	(2,4)	(2,5)
(3,1)	(3,2)	(3,3)	(3,4)	(3,5)
(4,1)	(4,2)	(4,3)	(4,4)	(4,5)
(5,1)	(5,2)	(5,3)	(5,4)	(5,5)

Here our
Goal node/Cell is a green cell as shown above (i.e having coordinates (1,1))
Start cell is having coordinates (n,n) if the maze is nxn maze

Our Heuristic is Consistent and admissible.
Therefore, We always get the optimal path with this heuristic when applied in the A* algorithm for our network graph model.