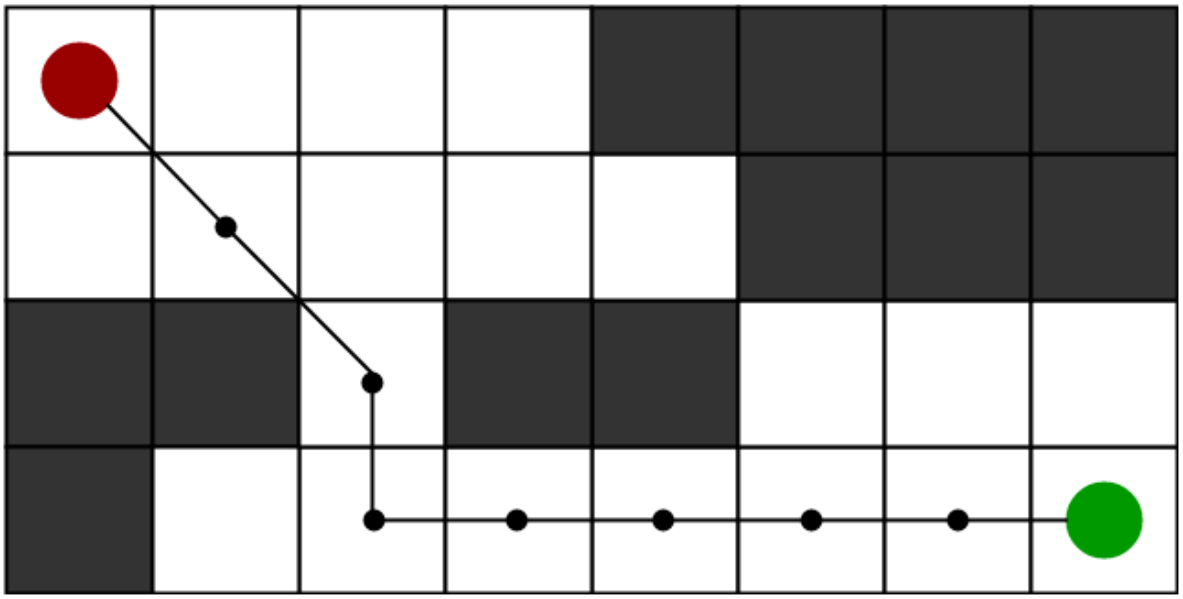
CSE 404

Artificial Intelligence Lab

Assignment on A* Search

Problem Definition

For this assignment, your objective is to create an intelligent agent that can navigate through a maze containing obstacles, aiming to reach a goal from a starting point. Imagine that your agent begins in a specific cell within the grid (highlighted in red). The agent's goal is to discover a path, preferably the shortest one, to reach the designated destination cell (highlighted in green). Your task involves assisting the agent in utilizing the A* search algorithm to find and follow this optimal path.



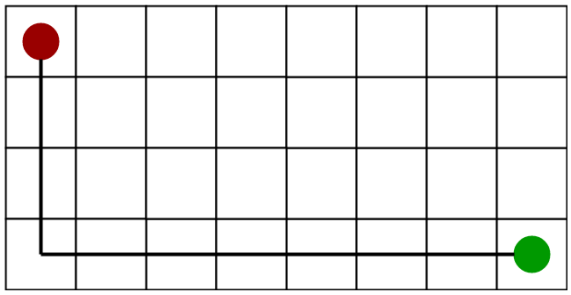
The agent can move from a cell to its adjacent (up, down, left, right) or diagonal cell. It cannot move to a cell if the cell has an obstacle in it.

Heuristics

The heuristics applicable to solving this problem include (but are not restricted to) :

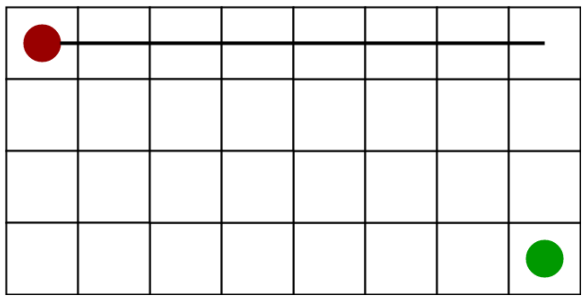
Manhattan Distance : The Manhattan distance is essentially the total of the absolute differences between the x and y coordinates of the goal cell and the current cell. Mathematically,

$$Manhattan\ Distance = \left|Goal_x - Current_x\right| + \left|Goal_y - Current_y\right|$$



Diagonal Distance : The Diagonal Distance is determined by taking the larger of the absolute differences between the x and y coordinates of the goal cell and the current cell. Mathematically,

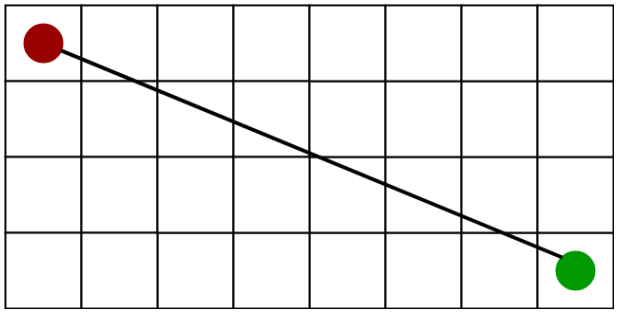
$$Diagonal\ Distance = max(\left|Goal_x - Current_x\right|, \left|Goal_y - Current_y\right|)$$



Euclidean Distance :

Euclidean Distance is the straight-line distance between two points in a space. Mathematically,

$$Euclidean\ Distance = \sqrt{(Goal_x - Current_x)^2 + (Goal_y - Current_y)^2}$$



Input and Output

You will take input from a text file (input.txt). The file will have the following lines :
The first line will contain two space separated integers m and n. These integers will be used to initialize the grid of dimension (m x n). The second line will contain one integer k, which denotes the number of obstacles. The subsequent k lines will specify the coordinates of these obstacles. Finally, the last two lines will indicate the coordinates of the starting cell and the destination cell, respectively.

m n	// For initializing a (m x n) grid
k	// number of coordinates that have obstacles
x_1 y_1	// Coordinates of 1st obstacle
x_2 y_2	// Coordinates of 2nd obstacle
.	
.	
.	
x_k y_k	// Coordinates of k-th obstacle
$start_x$ $start_y$	// Coordinates of the start cell
$goal_x$ $goal_y$	// Coordinates of the goal cell

Your code will output the path that the agent takes to reach the destination. Additionally, it will display the path's cost (number of cells along the path). It's recommended to include the runtime of your code executing the A* search algorithm, as this duration will provide insight into the relative effectiveness of the heuristics.

Sample Input and Output

Input

```
10 10

21

1 3
2 3
3 3
4 2
4 3
4 4
5 1
5 2
5 3
6 2
```

6 3
6 4
7 3
7 4
7 5
8 4
8 5
8 6
9 5
9 6
9 7

0 0
9 9

Output