Deadline: 21th-April-2021 Weight: 4.00

Lab Task # 07

You have been given with the code of **Breadth-First-Search** traversal on 2D Matrix i.e. full working code of your course assignment. You can download it from (Slate -> AI Lab -> Resources -> lab#07 -> updateAssignment.ipynb) and run it.

Note: This task is due for everyone with no concession!

Task

Your task is to modify the Breadth-First-Search function in the above code and use it to implement the followings;

- Greedy Best-First Search
- A* Search

Note: Use **Euclidian/Manhattan** distance as a heuristic function by comparing the indices of current state agent with the indices of goal state on 2DMatrix!

Heuristic Function:

```
function heuristic(node) =
  dx = abs(node.x - goal.x)
  dy = abs(node.y - goal.y)
  return D * (dx + dy)
```

Bonus!

Multiply the distance in steps by the minimum cost for a step. For example, if you're measuring in meters, the distance is 3 squares, and each square is 15 meters, then the heuristic would return 3 × 15 = 45 meters. If you're measuring in time, the distance is 3 squares, and each square takes at least 4 minutes to cross, then the heuristic would return 3 × 4 = 12 minutes. The units (meters, minutes, etc.) returned by the heuristic should match the units used by the cost function.