Artificial Intelligence Lab

A* Search Algorithm Implementation

Task 01:

Implement the A* algorithm to find the shortest path in a grid where certain cells are blocked (obstacles). You are given a grid with obstacles. You need to implement the A* search algorithm to find the shortest path from a starting point (start_x, start_y) to a goal point (goal_x, goal_y). The grid is represented as a 2D list where:

- 0 represents an open space.
- 1 represents an obstacle.

The A* algorithm should use the Manhattan distance as the heuristic function.

1. Is this complete? Is this optimal?

Task 02:

Implement the A* search algorithm to find the shortest path in two types of graphs:

1. Unweighted Graph: A grid where each move between adjacent nodes has the same cost and assume the heuristic values. In this case, how is A* algorithm will be different from the BFS algorithm. Write your observations and recommendations for using/not using A* for weighted graph.
2. Weighted Graph: A grid where each move between adjacent nodes can have different costs (weights) and heuristics values.

Write code for both types of graphs using the A* algorithm.