

113-1 Artificial Intelligence HW1 Report

智能所 312581029 廖永誠

Questions:

1. Provide the route planning from Arad to the destination station Bucharest

- Path from Arad to Bucharest: ['Arad', 'Sibiu', 'Fagaras', 'Bucharest']

2. Provide the route planning from Oradea and Mehadia to the destination station Bucharest, respectively.

- Path from Oradea to Bucharest: ['Oradea', 'Sibiu', 'Fagaras', 'Bucharest']
- Path from Mehadia to Bucharest: ['Mehadia', 'Dobreta', 'Craiova', 'Pitesti', 'Bucharest']

3. Discuss the Pros and Cons of Greedy Best-First Search.

- Pros

1. Greedy Best-First Search is often faster than others uniformed search algorithms. Compare to BFS or DFS, Greedy Best-First Search can find the result in fewer nodes explored.
2. Greedy Best-First Search is easy to implement. It only needs to sort the nodes by the heuristic function and then expand the node with the smallest heuristic value.
3. Space is less than other algorithms. Greedy Best-First Search only needs to store the nodes in the frontier, which is less than BFS or other algorithms.

- Cons

1. Greedy Best-First Search is not guaranteed to find the optimal solution. Since it only considers the heuristic value, it may not find the optimal solution.
2. If the heuristic function is not well-designed, the search may get stuck in a loop. So the heuristic function should be admissible and consistent.

4. Provide suggestions for improving the disadvantages of Greedy Best-First Search.

- In my opinion, the disadvantage that Greedy Best-First Search is that it can have possible to get stuck in a loop.

- To design a suggestions for improving but not changing the algorithm to much, we can add a mechanism to avoid the loop. For example, we can add a mechanism to avoid the loop by storing the visited nodes in the frontier. If the node has been visited, we can skip it and expand the next node. This mechanism can avoid the loop and make the search more efficient.