INTRODUCTION TO ARTIFICIAL INTELLIGENT
REPORT LAP 1

1. Information:

Name: Nguyễn Quý Thanh.
Student ID: 18127210.

2. Search strategies implementation:

- Breadth First Search.
- Uniform Cost Search.
- Iterative Deepening Search.
- Greedy Best First Search.
- A* Graph Search.

3. Level of completion:

100% - successfully implemented all algorithms.

4. Brief overview:

- **Search strategies:**

i. Breadth First Search:

Use queue as frontier to maintain search order.
Test goal right when a state is generated.

ii. Uniform Cost Search:

Use priority queue (min - heap) as frontier to maintain search order.
Test goal when a state is popped out of the heap.

iii. Iterative Deepening Search:

An improvement of Depth Limited Search DLS, which cutoff Depth First Search when it reaches limit.
Use stack as frontier to maintain search order.
Test goal right when a state is generated.

iv. Greedy Best First Search:

Use a heuristic function (Manhattan distance) as a cost function.

Always choose the state with lowest heuristic value.
Use priority queue (min - heap) as frontier to maintain search order.
Test goal right when a state is generated (greedy strategy).

v. A* Graph Search:

Most well – known Best First Search strategy.
Use a heuristic function (Manhattan distance to the goal) and a cost function (cost to go from the start) as a total cost.
Always choose the state with lowest total cost.
Use priority queue (min - heap) as frontier to maintain search order.
Test goal when a state is popped out of the heap.

5. Extras:

- A Frontier class wraps the heapq module.
- A Maze class for OOP style.
- Use lazy delete strategy for updating / inserting to the frontier.