

Homework #1 Report

土木所交通組 R12521502 陳冠言

Part A : Show your autograder results and describe each algorithm.

Q1. Depth First Search

```
C:\Users\ericc\Downloads\AI\AI2024-hw1\AI2024-hw1>python autograder.py -q q1
Starting on 3-17 at 15:18:44

Question q1
=====
*** PASS: test_cases\q1\pacman_1.test
***   pacman layout:      mediumMaze
***   solution length: 130
***   nodes expanded:     146

### Question q1: 5/5 ###

Finished at 15:18:44

Provisional grades
=====
Question q1: 5/5
-----
Total: 5/5
```

Depth First Search (DFS) 是以深度優先的搜尋演算法，其主要特點是從起始狀態開始，一直向前搜索直到最深處，然後再回溯到上一個節點繼續搜索，並且直到找出目標或搜索完整個圖。在 DFS function 中主要使用 stack 來實現建構 DFS。先將起點放入 stack 中，接著不斷 pop 擴展到後繼節點，並將這些後繼節點推入堆疊中，如果堆疊不為空且未達到目標狀態則繼續搜索。

Q2. Breadth First Search

```
C:\Users\ericc\Downloads\AI\AI2024-hw1\AI2024-hw1>python autograder.py -q q2
Starting on 3-17 at 15:19:46

Question q2
=====
*** PASS: test_cases\q2\pacman_1.test
***   pacman layout:      mediumMaze
***   solution length: 68
***   nodes expanded:     269

### Question q2: 5/5 ###

Finished at 15:19:46

Provisional grades
=====
Question q2: 5/5
-----
Total: 5/5
```

Breadth First Search (BFS) 使用 queue 來實作廣度優先搜尋。先將其起點放入 queue，然後不斷 pop 最前面的元素，並擴展到後繼節點，接著將新節點放入尾端。BFS 保證了會先搜索到與起點相距最近的節點，然後逐步擴展到更遠的節點。如果找到目標狀態，則回傳到目標的動作，否則執行直到找出目標為止。

Q3. Uniform Cost Search

```
C:\Users\ericc\Downloads\AI\AI2024-hw1\AI2024-hw1>python autograder.py -q q3
Starting on 3-17 at 15:20:21

Question q3
=====
*** PASS: test_cases\q3\ucs_4_testSearch.test
***   pacman layout:      testSearch
***   solution length: 7
***   nodes expanded:    14
*** PASS: test_cases\q3\ucs_5_goalAtDequeue.test
***   solution:          ['1:A->B', '0:B->C', '0:C->G']
***   expanded_states:   ['A', 'B', 'C']

### Question q3: 10/10 ###

Finished at 15:20:21
:
Provisional grades
=====
Question q3: 10/10
-----
Total: 10/10
```

在程式中主要使用 priority queue 來實作 Uniform Cost Search。此演算法按照節點的最小總成本進行搜索，即考慮已經走過的路徑的總代價，並優先探索具有最小總成本的節點，先將起點加入 priority queue，且設定優先度為 0，並接著處理後續節點直到找到目標為止。

Q4. A* Search (null Heuristic)

```
C:\Users\ericc\Downloads\AI\AI2024-hw1\AI2024-hw1>python autograder.py -q q4
Starting on 3-17 at 15:20:57

Question q4
=====
*** PASS: test_cases\q4\astar_0.test
***   solution:          ['Right', 'Down', 'Down']
***   expanded_states:   ['A', 'B', 'D', 'C', 'G']

### Question q4: 15/15 ###

Finished at 15:20:57
:
Provisional grades
=====
Question q4: 15/15
-----
Total: 15/15
```

A Star Search 是一種啟發式搜尋演算法，結合了 heuristic 和 cost function，以選擇最低綜合成本的節點。但因為 heuristic 的參數設定為 nullHeuristic，因此回傳的值皆為 0，也代表 A* Search 僅根據實際成本來進行搜尋，不會考慮任何附加的預估成本。程式主要使用優先 priority queue 來實作 A* Search。先將起點放入 priority queue 然後循環處節點，每次彈出優先度最高的節點進行擴展。如果 pop 到目標值，則回傳其動作序列，否則將其後繼節點放入 priority queue。

Part B : Describe the difference between Uniform Cost Search and A* Contours

Answer :

Uniform Cost Search 是一種盲目地搜尋演算法，不使用任何啟發式訊息，它只根據每個節點的實際成本來進行搜尋，從而保證找到的解是最短的。而 A* Contours 是一種啟發式搜尋演算法，通常能夠更有效地找到解答，且它同時考慮實際成本和估計成本，從而在搜索過程中更有效率地選擇擴展節點。

Part C : Describe the idea of Admissibility Heuristic

Answer :

Admissibility Heuristic 主要需要滿足保守估計和低於實際成本兩個特點。其中保守估計，在 cornersHeuristic function 計算了從當前位置到未達到的每個角落的曼哈頓距離，且每次迭代中選擇最接近當前位置的未訪問 corner，這樣可確保估計的距離趨於保守。因此 cornersHeuristic 提供一個保守估計，有助於確保搜尋更有效率且正確。而低於實際成本特性是指函數回傳的成本是根據從當前位置到達目標的最小曼哈頓距離，即從當前位置到目標角落的最小移動步數，由於曼哈頓距離是一個保守的估計，因此 cornersHeuristic 回傳的成本會小於等於實際成本。

Appendix

python autograder.py

```
Provisional grades
=====
Question q1: 5/5
Question q2: 5/5
Question q3: 10/10
Question q4: 15/15
Question q5: 5/5
Question q6: 9/9
-----
Total: 49/49
```