

姓名: 王關平Kenneth Ong Kuan Phing 學號: 41047041S

1. 對於這次作業的編寫及測試使用的硬體和軟體為： 軟硬體規格:

- 2020 Macbook Air M1
  - 8-core CPU with 4 performance cores and 4 efficiency cores
  - 7-core GPU, 8-core GPU
  - 16-core Neural Engine
- python3 version 3.11.2 執行檔案方式為

```
python3 <filename>.py
```

使用原因:

- 我只有這台筆電
- python語法簡單明瞭

聯絡號碼: 0916-303325

2. 執行方式

```
$ python3 <檔案名稱>.py
```

測試輸入

1. 0 1 0 1 0
2. 1 0 1 1 0 1 1
3. 1 1 1 1 1 0 0 1
4. 1 0 0 1 1 0 1 0 0
5. 0 1 1 0 0 1 1 1 1 0

以上的測資皆使用python程式([generate-test.py](#))產生的，其中0和1為隨機產生的。

4. 第一支IDS中我使用的方法為利用遞迴的方式對新的state呼叫Depth Limited Search函數, 產生一顆recursion tree, 所使用的資料結構為Tree. 同時, 我也創造了一個list, 記載這先前拜訪過的盤面狀態, 若發現目前的盤面之前有拜訪過的話, 則不再繼續往下遞迴, 因為再往下只會產生同樣的pattern。

消耗時間：

測資一：7.033348083496094e-05

測資二：0.08513712882995605

測資三：0.7685933113098145

測資四：2.091130018234253

測資五：53.173689126968384

消耗空間： $O(d)$  where d is the depth of the tree.

5. 第二支IDA\*中我也一樣使用遞迴的方式對新的state呼叫Search函數, 產生一顆recursion tree, 所使用的資料結構為Tree.

同時, 我也創造了一個list, 記載這先前拜訪過的盤面狀態, 若發現目前的盤面之前有拜訪過的話, 則不再繼續往下遞迴, 因為再往下只會產生同樣的pattern。

消耗時間：

測資一：5.698204040527344e-05

測資二：0.00013494491577148438

測資三：0.0001659393310546875

測資四：0.0001380443572998047

測資五：0.00029778480529785156

程式能解的最大盤面為60

6. 此作業遇到的一些狀況及困難有：

- 該如何實作heuristic function
- 設法擋掉先前出現過的盤面

7. 參考文獻或網站

- <https://www.geeksforgeeks.org/iterative-deepening-searchids-iterative-deepening-depth-first-searchiddfs/>
- [https://www.algorithms-and-technologies.com/iterative\\_deepening\\_a\\_star/python](https://www.algorithms-and-technologies.com/iterative_deepening_a_star/python)
- [https://www.algorithms-and-technologies.com/iterative\\_deepening\\_dfs/python](https://www.algorithms-and-technologies.com/iterative_deepening_dfs/python)