

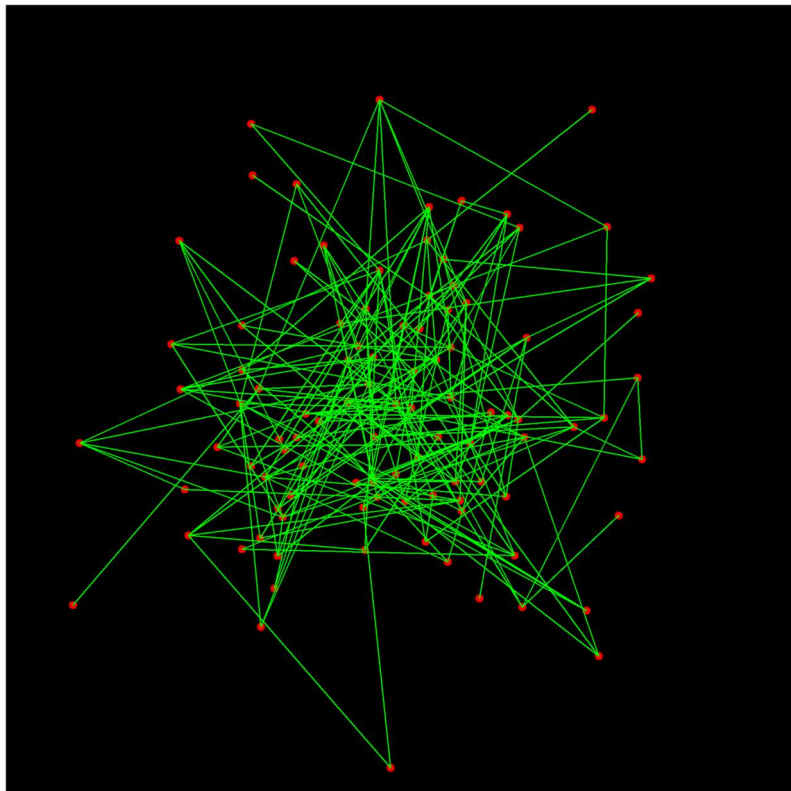
## 期末專題

110753201 資科碩一 曹昱維

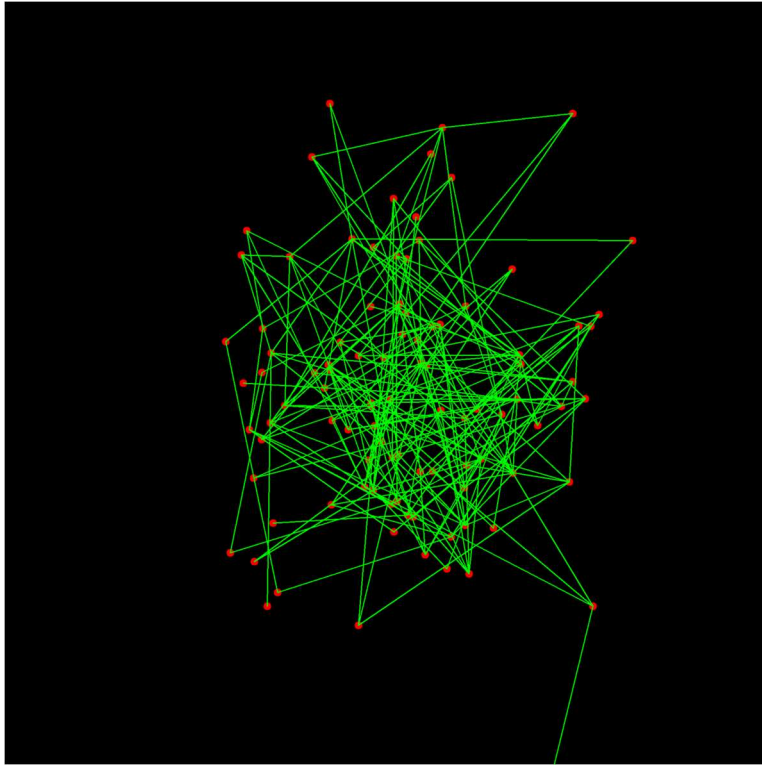
### ● 設計公路網路模型：

- 產生節點(vertex/node)座標的方式：
  - 從常態分佈 $N(\mu, \sigma^2)$ ,  $\mu = 1000, \sigma = 300$ 抽選 x 與 y 座標
  - 中心點: (1000,1000)
  - 99.7% 的節點距離中心點900以內的範圍
- 產生道路(edge)的方式：
  - 使用隨機數除以節點總數 n 的餘數來決定道路的起點與終點
  - 給定道路總數量
    - 當  $n < 100$  道路總數設定為  $\frac{n(n+1)}{8}$
    - 當  $n \geq 100$  道路總數設定200
  - 道路長度不能超過800
  - 所有的道路連接起來的 graph 必須是 strongly connected
- 公路網路圖(number of vertex=100, number of edge=200)
  - 圖的大小為2000X2000, 中心點為1000, 1000

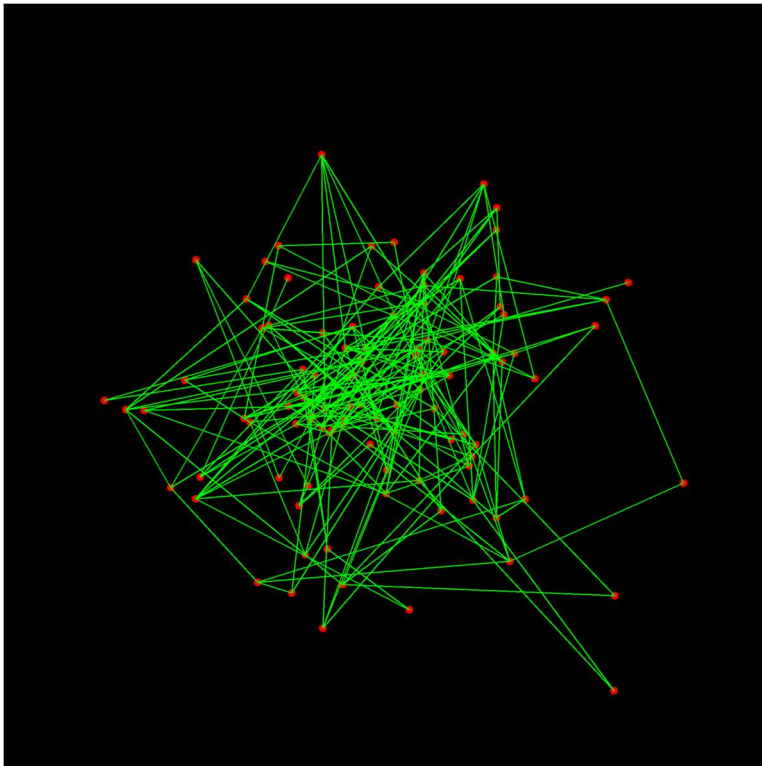
1.



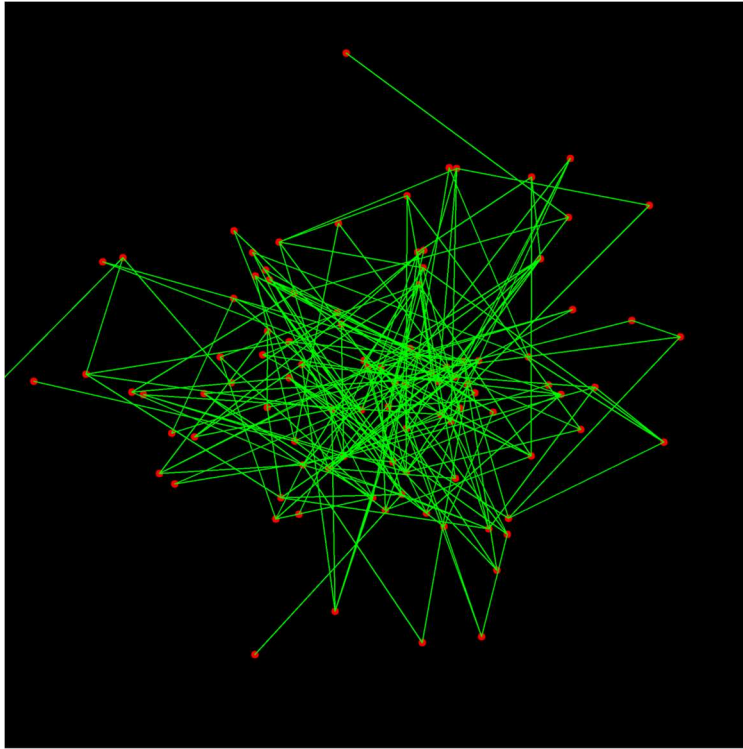
2.



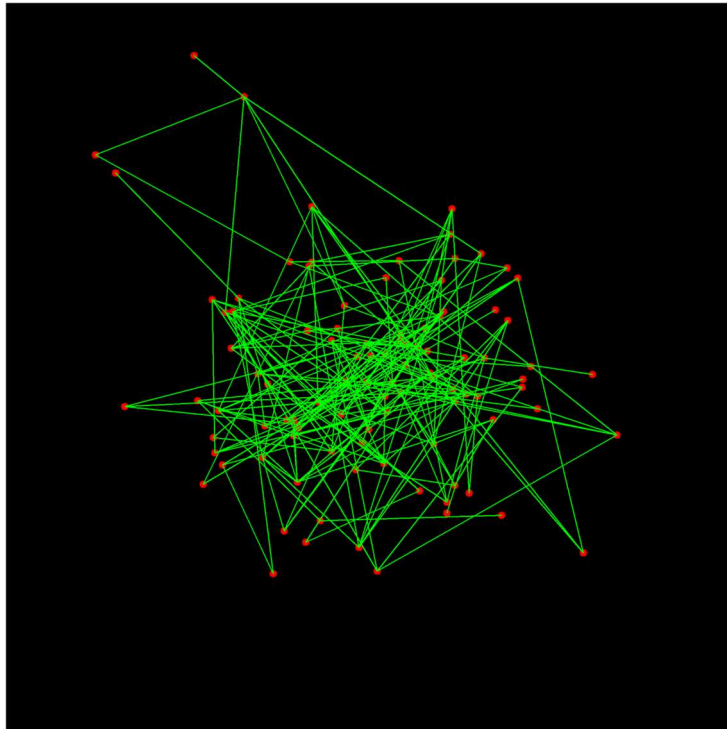
3.



4.

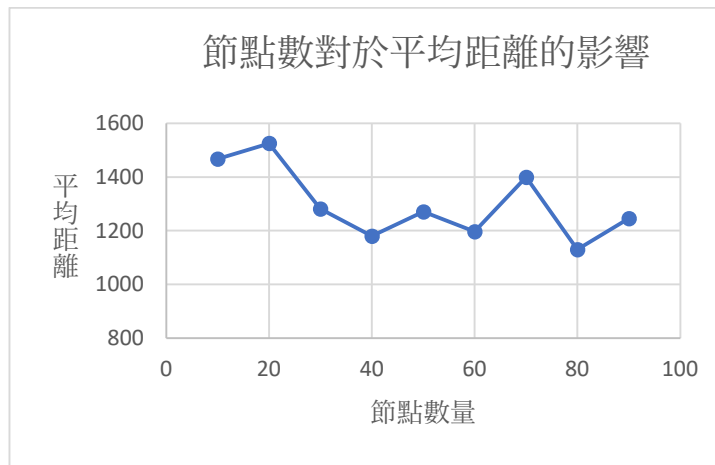


5.



- 撰寫程式探討節點數 $n$ 對於平均距離的影響：

- 下圖為在不同節點總數的情況下各節點之間的最短距離的平均值紀錄

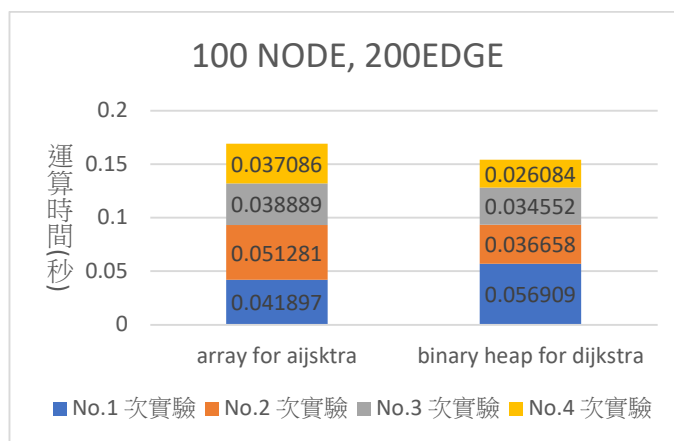


- 從圖表可以推測出節點總數與各節點之間的最短距離的相關性並不強
- 推測因素：

即使節點上升，但是點與點之間的距離依舊是按照常態分佈，而邊數量與節點總數的比例又是固定的，所以各節點之間的最短距離的平均不受節點總數的影響

- 比較資料結構對於 Dijkstra 演算法執行時間的影響：

- 下圖為節點數量為100與邊數量為200的情況下，執行時間的紀錄



- 從圖中可以看出在大部分的情況下使用 binary heap 的 Dijkstra 會比使用 array 的 Dijkstra 還要快

- 原始資料

- 不同資料結構對 dijkstra 的影響

100 NODE, 200EDGE		
實驗 ID	array for dijkstra	binary heap for dijkstra
1	0.041897	0.056909
2	0.051281	0.036658
3	0.038889	0.034552
4	0.037086	0.026084

- 不同節點總數的情況下各節點之間的最短距離的平均值

節點總數	平均距離
10	1467.2
20	1526.01
30	1281.49
40	1179.64
50	1270.36
60	1196.73
70	1400.08
80	1130.25
90	1246.3

- 原始程式碼：

- 可以參考附檔

- 或是網址：

[https://github.com/theabc50111/nccu\\_cs\\_hw/blob/main/DataStructure\\_HW/HW4/hw4.cpp](https://github.com/theabc50111/nccu_cs_hw/blob/main/DataStructure_HW/HW4/hw4.cpp)

- 參考來源

- <https://www.geeksforgeeks.org/dijkstras-algorithm-for-adjacency-list-representation-greedy-algo-8/>
- <https://favtutor.com/blogs/dijkstras-algorithm-cpp>
- <https://www.tutorialspoint.com/cplusplus-program-to-check-the-connectivity-of-undirected-graph-using-dfs>
- <https://www.techiedelight.com/check-given-graph-strongly-connected-not/>
- <https://www.tutorialspoint.com/cplusplus-program-to-generate-a-random-undirected-graph-for-a-given-number-of-edges>