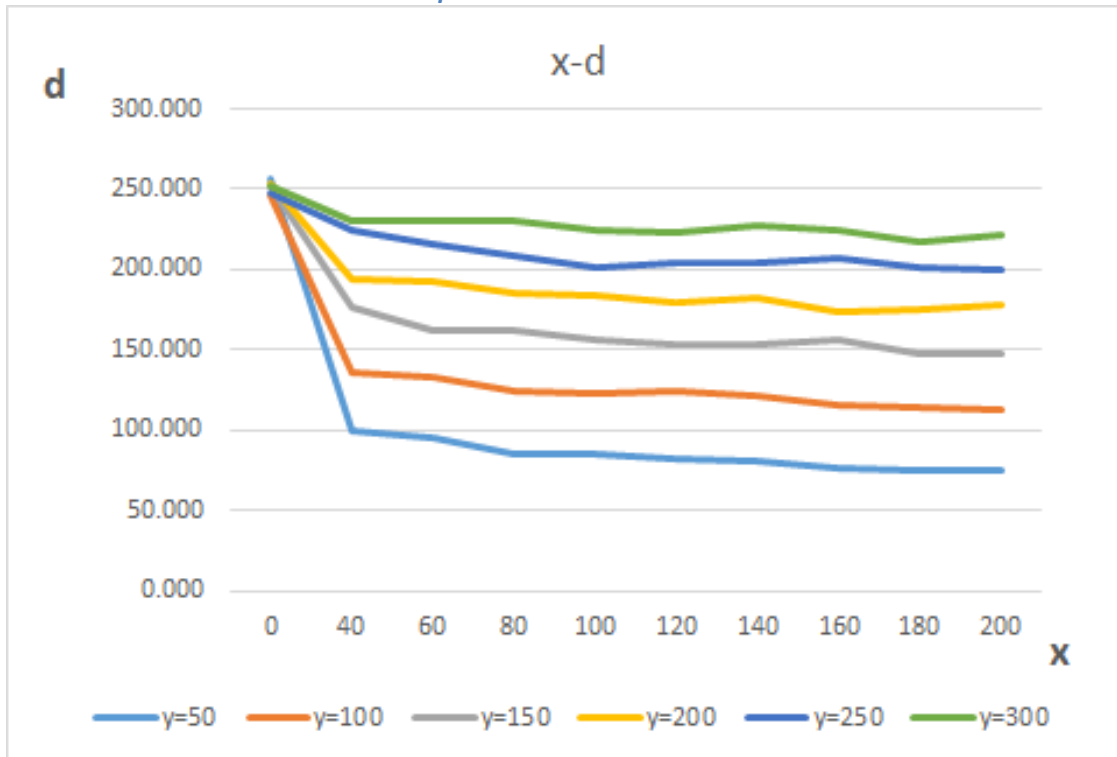


Small World

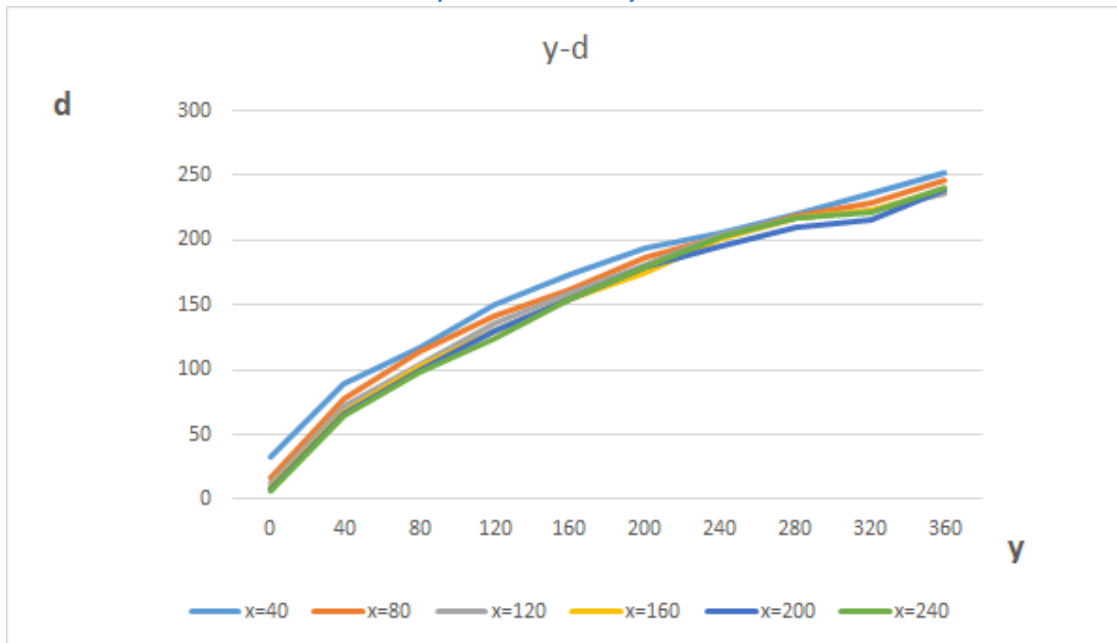
107703020 資科二 邱品硯

1. What is the relationship between x and d ?



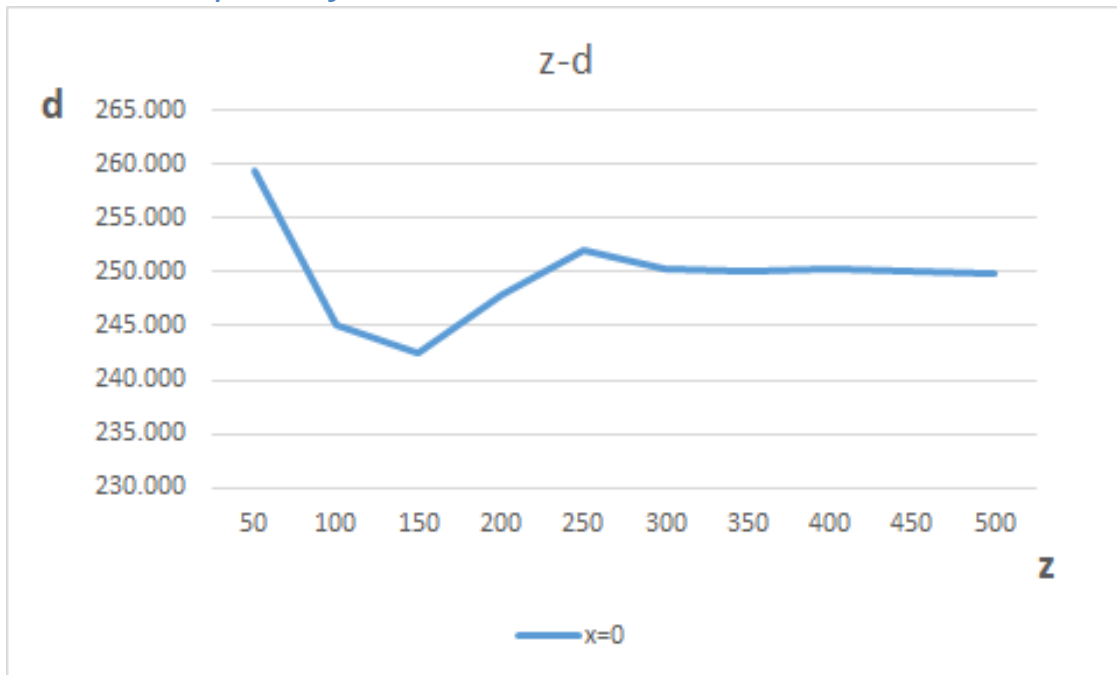
1. 當 $x = 0$ 時：
 d 皆在 250 左右，表示沒有額外增加路徑的條件下，平均最短路徑為 250
proof：任意兩點的距離可能總和 = $1 + 2 + 3 + \dots + 499 + 500 + 499 + \dots + 3 + 2 + 1$
平均距離為 $250000/999 = 250.25$
2. 當 $y < 250$ 時：
 d 的值隨者 x 的增加，而逐漸趨近於 y
3. 當 $y \geq 250$ 時：
 d 的值會逐漸趨近於小於 y 的值，因為隨者路徑的增長，在相距 $< y$ 的兩點間，會走原始 cycle 的路徑

2. What is the relationship between y and d ?



y 與 d 成高度正相關，當 y 大於 200 後，成長漸趨平緩

3. How to choose z properly to reflect the true average distance between all pairs of source and destination?

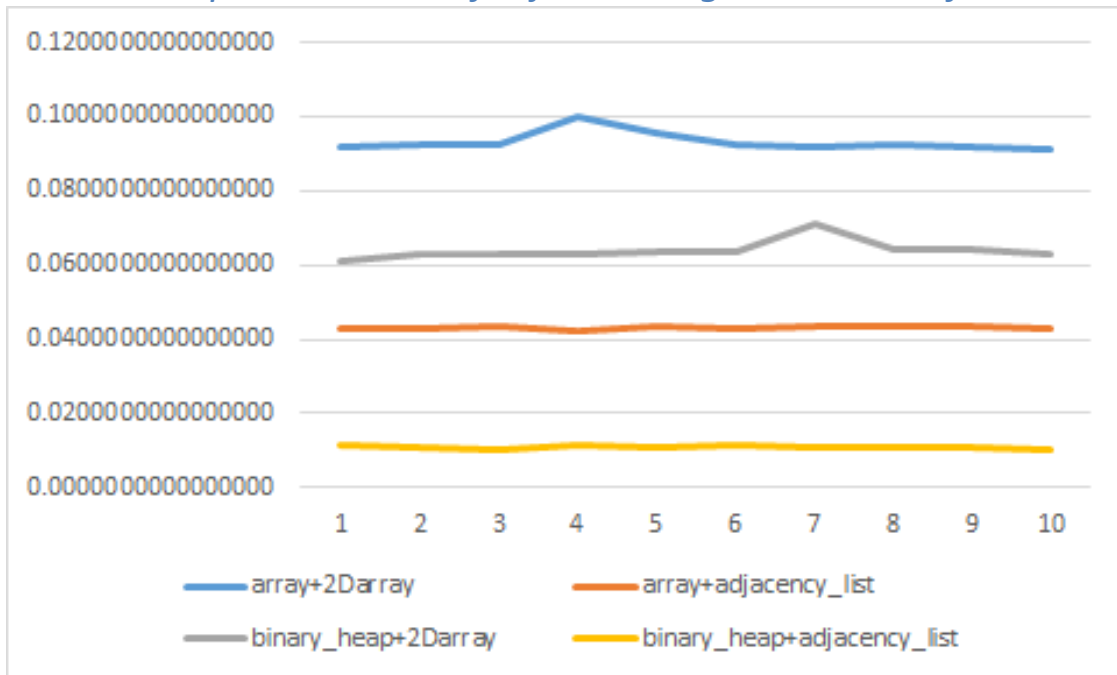


如題 1 所述，當 $x = 0$ 時，平均最短距離為 250

上表為當 $x = 0$ 時， z 與 d 的關係

當 z 越大，數據越精確。

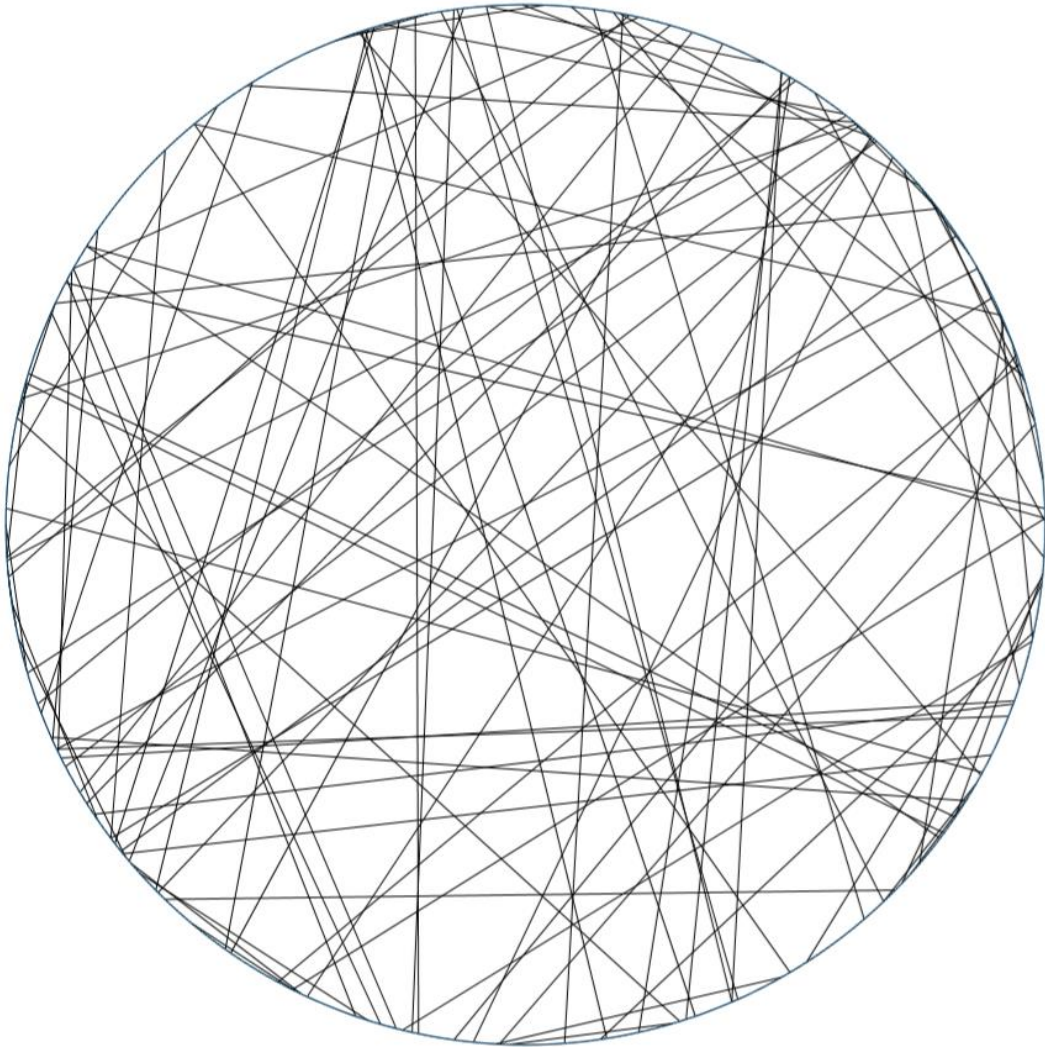
4. Which implementation of Dijkstra's Algorithm is the fastest?



1. 用 2D-array 來存取 Graph 時，走訪整個圖需要花 $O(V^2)$
若用 adjacency list，則走訪整個圖只需要花 $O(V + E)$ $\begin{cases} V: \text{頂點數} \\ E: \text{邊的數量} \end{cases}$
故 2D-array 花的時間比 adjacency list 長
2. heap 用 array 時，更新資料需要花 $O(n)$
若用 binary heap，則只需要花 $O(\log n)$
故 array 花的時間比 binary heap 長

整體來說，四個中最有效率的方式就是 binary heap + adjacency list

Draw a graph where $x = 100$



參考網站

Source code : [Geeksforgeeks](#)

Graph : [networkx](#)