

資料結構

一、欲決定某一演算法的計算時間，一般是由計算指令執行的次數來決定。試決定下列片斷程式的執行次數：

(a) for $i \leftarrow 1$ to n do (6 %)
 for $j \leftarrow 1$ to n do
 $x \leftarrow x + 1$
 end
end

(b) for $i \leftarrow 1$ to n do (6 %)
 $j \leftarrow i$
 for $k \leftarrow (j + 1)$ to n do
 $x \leftarrow x + 1$
 end
end

(c) 請列出欲排序 n 個數時，使用 bubble sort, quick sort, heap sort 及 merge sort 所需之 average time 及 worst time。並由上述演算法中任寫出一種你認為其 average time 最小的 sorting 演算法。 (12 %)

(d) 用你所寫的 sorting 演算法排序 { 23, 42, 74, 13, 94, 58, 12 } (6 %)

二、假設有一圖(graph)描述如下：

$$G = (N, E)$$

$$N(G) = \{ a, b, c, d \}$$

$$E(G) = \{ \langle a, b \rangle, \langle a, c \rangle, \langle b, d \rangle, \langle c, d \rangle, \langle b, a \rangle, \langle d, c \rangle \}$$

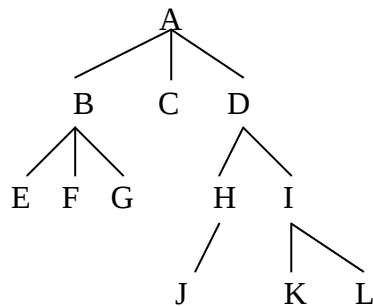
其中 N 及 E 分別代表節點(node)及邊(edge)所成的集合。

(a) 請以 adjacency matrix 及 adjacency list 表示此圖。 (6 %)

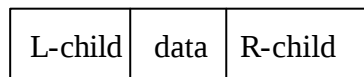
(b) 請寫出圖形追蹤的 depth-first search 及 breadth-first search 的演算法，並分別舉一例說明之。 (14 %)

- (c) 試比較上述兩者所需之資料結構。 (5 %)
- (d) 在一 weighted graph 中，對已知之起始與終節點，上述兩種圖形追蹤法可確定找到最短路徑?為什麼? (5 %)

- 三、(a) 何謂二元樹 (binary tree) ? (3 %)
- (b) 請說明如何將下列樹化為二元樹。 (6 %)



- (c) 請以 preorder, inorder, postorder 追蹤上述轉換過的二元樹。 (6 %)
- (d) 假設一個二元樹的節點結構為: (15 %)



其中 L-child, R-child 為鏈結(link), 分別指向左右兩子樹, 請寫出判斷兩二元樹是否相等的方法。

- 四、試比較二元搜尋(binary search)及雜湊搜尋(hashing search), 並任舉兩種你所知道的雜湊函數(hashing function)說明之。 (10%)