

1. (10%) 有一顆 2-3 tree, 一開始是空的,

(a) 依序 insert

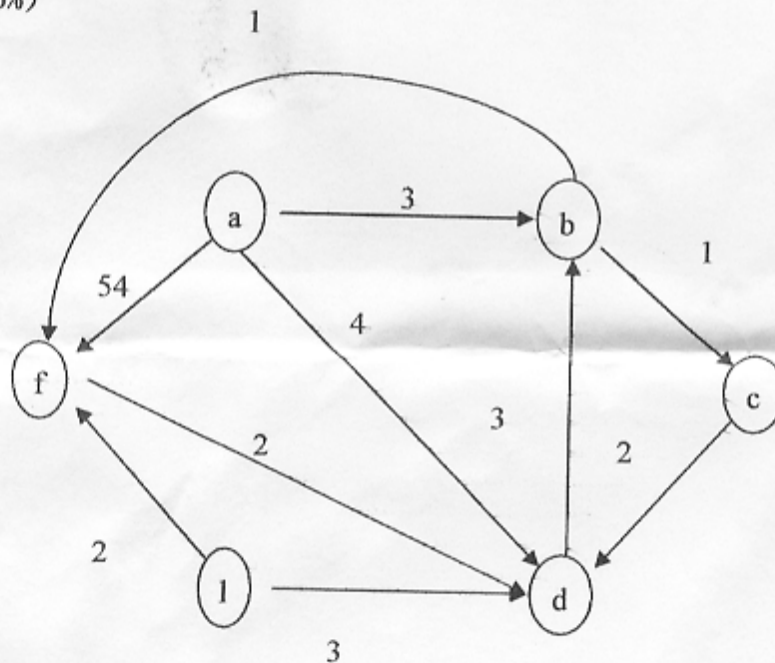
5, 2, 7, 0, 3, 4, 6, 1, 8, 9

之後, 繪圖顯示該樹

(b) 接著, delete 3, 之後

繪圖顯示該樹.

2. (6%)



上圖中, 邊 (arc) 之數字代表成本 (cost).

(一) 請用相鄰矩陣 (adjacency matrix) 表示此圖之成本

(二) 請用相鄰串列 (adjacency list) 表示此圖之成本

3. (7%) 假設我們用五個桶 (buckets) 的雜湊表 (hashing table), 每個桶可放一筆資料, 而雜湊函數 (hash function) h 為:

$$h(i) = i \bmod 5$$

使用線性探測 (linear probing) 來解決碰撞 (collision)。假設一開始雜湊表

是空的，依序放入 (insert) 23, 48, 35, 4, 10 五筆資料。請繪圖顯示最後雜湊表之內容。

4. (15%) 資工系全系課程可視為一個集合，裏面每個 element 即每門課，含：
(1) courseNo (課號)，(2) courseName (名稱)，(3) Description (簡介)等，請分別用下面三種資料結構來 represent 該集合，並寫出一個 operation，用 Java 寫出答案。

- (a) a linked list
- (b) a hash table
- (c) a binary search tree

5. (12%) 下面有 16 個數

22, 36, 6, 79, 26, 45, 75, 13, 31, 62, 27, 76, 33, 16, 62, 47

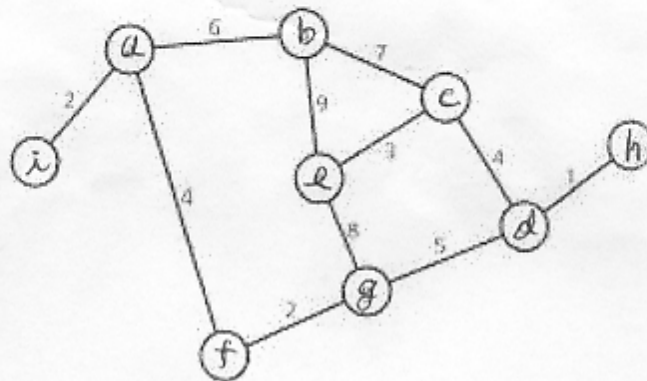
請分別用四種演算法進行排序：

- (a) quick sort
- (b) insertion sort
- (c) heap sort
- (d) radix sort, 視每個數為 range 0-9 的兩個數字(digits)

請繪簡圖表示之。

6. (30%) Consider the graph below. (a) Please design a data structure to implement the depth-first search. Then, please show the contents of the data structure and the traversal visits the vertices in the order beginning at vertex "a" using depth-first search. (b) Please design a data structure to implement the bread-first search and then show the contents of the data structure and the traversal visits the vertices in the order beginning at vertex "a" using breadth-first search. (c) Write a program using Prim's algorithm to find a minimum spanning tree of the graph beginning at vertex "a".

3/3



7. (12%) Please implement the stack by using an **array**.

```
#define MAX_STACK 100;
typedef int ITEM_TYPE;
typedef struct stack_type {
    ITEM_TYPE item[MAX_STACK];
    int top;
} STACK_TYPE;

void create_stack (STACK_TYPE *stack);(2%)
void destory_stack (STACK_TYPE *stack);(2%)
BOOLEAN empty_stack (STACK_TYPE *stack);(2%)
BOOLEAN full_stack (STACK_TYPE *stack);(2%)
void push (STACK_TYPE *stack, ITEM_TYPE new_item);(2%)
void pop (STACK_TYPE *stack, ITEM_TYPE *old_item);(2%)
```

8. (8%) Explain the following terms:

- (a) ADT
- (b) Data type
- (c) Array
- (d) Data structures