

類 科：資訊處理

科 目：資料結構

考試時間：2小時

座號：_____

※注意：(一)禁止使用電子計算器。

(二)不必抄題，作答時請將試題題號及答案依照順序寫在試卷上，於本試題上作答者，不予計分。

一、假設我們有一個由 26 個英文字母所構成的文字檔。

(一)請說明如何建構一棵霍夫曼樹 (Huffman tree) 來壓縮該文字檔。(15 分)

(二)請說明如何利用你所述之方法建構的霍夫曼樹壓縮該文字檔。(5 分)

(三)請說明如何解壓縮利用你所述方法壓縮的文字檔。(5 分)

二、下列的虛擬碼程式片段中，I 和 S 均為遞迴函式 (recursive function)，I 和 S 的參數 A 是一個整數陣列；I 和 S 的參數 i 為不為負的整數，主要是做為陣列 A 的索引 (index)。假設陣列 A 的元素個數為 n ，且其索引值為 0 到 $n-1$ 之間的數值。虛擬碼 swap x and y 的意思是將變數 x 與變數 y 的儲存值互換；亦即執行之後變數 x 的儲存值為執行前變數 y 的儲存值，執行之後變數 y 的儲存值為執行前變數 x 的儲存值。令 $T(n)$ 為呼叫函式 $I(A, n-1)$ 的執行時間。 $T(n)$ 會隨著陣列 A 所儲存的數值不同而有所不同。

```
S(A, i) {
    If  $i \leq 0$ , then return;
    S(A,  $i - 1$ );
    I(A,  $i$ );
    Return; }
I(A, i) {
    If  $i \leq 0$ , then return;
    If  $A[i] < A[i - 1]$  {
        swap  $A[i]$  and  $A[i - 1]$ ;
        I(A,  $i - 1$ ); }
    Return; }
```

(一)請用 O -notation 表示 $T(n)$ 的上界 (upper bound)；請用 Ω -notation 表示 $T(n)$ 的下界 (lower bound)。(5 分)(二)請說明 $T(n)$ 最大時，程式開始執行前陣列 A 所儲存的數值有何特性？理由為何？(5 分)(三)請問函式 $S(A, n-1)$ 的時間複雜度為何？請說明理由。(5 分)(四)請問執行函式 $S(A, n-1)$ 後，陣列 A 儲存的內容有何特性？請證明你的觀察。(10 分)

(請接背面)

類 科：資訊處理
科 目：資料結構

三、堆積 (heap) 是一棵完整二元樹 (complete binary tree)，每個節點儲存一個鍵值 (key value)，且每一個內部節點 (internal node) 的鍵值都不比其子節點的鍵值小。

(一)請畫一棵七個節點的堆積，其節點儲存的鍵值形成的集合為

{100, 10, 55, 69, 38, 27, 48}。(5分)

(二)請說明如何利用陣列 (array) 實做一棵 n 個節點的堆積。(5分)

(三)假設一棵 n 個節點的完整二元樹，其每個節點儲存一個鍵值，除了根節點 (root) 之外，其他內部節點的鍵值均不比其子節點的鍵值小。請用虛擬碼描述將這樣的一棵二元樹調整成堆積的演算法。(10分)

(四)請說明如何利用上述演算法將一棵 n 個節點之堆積的根節點儲存的鍵值刪除，得到一棵儲存其餘 $n-1$ 個鍵值的堆積。(5分)

四、我們想設計一個動態資料結構儲存數字集合 $S = \{0, 1, 2, \dots, n-1\}$ 的倆倆沒有交集，而且聯集等於 S 的子集合。初始時有 n 個元素，個數為 1 的子集合，分別為 $\{0\}, \{1\}, \dots, \{n-1\}$ 。我們希望這個資料結構可以支援以下兩個功能：

1. $\text{union}(x, y)$: $x, y \in S$ 。 $\text{union}(x, y)$ 將包含 x 的子集合與包含 y 的子集合聯集得到一個新的子集合，原來的子集合不再存在。

2. $\text{equivalence}(x, y)$: $x, y \in S$ 。 $\text{equivalence}(x, y)$ 判斷 x 與 y 是否屬於同一個子集合，若屬於同一個子集合，則回傳“TRUE”，否則回傳“FALSE”。

上述兩個函式必須能夠依任何順序交替執行。

(一)請描述一個可以達成上述需求而且 $\text{union}(x, y)$ 與 $\text{equivalence}(x, y)$ 的時間複雜度均為 $O(\log n)$ 的資料結構。(15分)

(二)請用虛擬碼描述可以在上述資料結構運作的 $\text{union}(x, y)$ 函式。(5分)

(三)請用虛擬碼描述可以在上述資料結構運作的 $\text{equivalence}(x, y)$ 函式。(5分)