

# 國立中央大學八十八學年度碩士班研究生入學試題卷

所別： 數學研究所 不分組 科目： 資料結構 共 2 頁 第 1 頁

( 考題共六題,請任選五題作答 )

1. ( 每小題 4 分 ) 以下各類問題,以何種資料結構或方式處理較適當,請說明理由

- (a) 老鼠走迷宮
- (b) 模擬自動洗車機車輛的運作
- (c) 計算機內管理程式執行的優先順序設計
- (d) 航空公司的旅客航班最佳路線選定
- (e) 用大整數來計算  $\pi$  至少到小數點百位

2. 遞迴 (recursion)

- (a) ( 6 分 ) 函式若使用遞迴 (recursion) 方式撰寫,請問其優缺點分別為何?
- (b) 有 pascal 三角形為

```
      1
     1 1
    1 2 1
   1 3 3 1
  1 4 6 4 1
  ....
```

假設此 pascal 三角形的第  $i$  列,第  $j$  個可用  $\text{pascal}(i,j)$  表示,例如:  
 $\text{pascal}(2,1) = 2, \text{pascal}(4,2) = 6$ ,這裡  $i, j$  都由 0 起算

- i. ( 7 分 ) 請用數學方式表示  $\text{pascal}(i,j)$  的遞迴公式
- ii. ( 7 分 ) 請用常用的電腦語言或虛擬程式碼 (pseudo code) 求得 pascal 三角形第  $i$  列,第  $j$  個的值,如

```
int pascal( int i , int j ) {
    ....
}
```

3. 連結序列 (linked list)

- (a) ( 5 分 ) 在撰寫以連結序列為主的程式時,常會不小心而造成計憶空間遺失 (memory leak),請解釋其義?
- (b) ( 5 分 ) 假設有兩指標 (pointer),一指向連結序列的第一個結點 (node),另一指向最後一個結點,請問此連結序列的總結點數目,其值是否可由兩指標值相減而得,請說明理由
- (c) ( 5 分 ) 假設某連結序列的結點被定義為

```
struct Node {
    int val ;
    Node* ptr ;
};
```

若某一函式,  $\text{delete\_head\_node}$ , 需要去除一連結序列的第一個結點,請問函式的首行可否宣告成以下形式,請說明理由,若有錯,則須如何改正:

```
void delete_head_node ( Node* head ) ;
```

這裡  $\text{head}$  是一指向連結序列第一個結點的指標

參考用

注意：背面有試題

# 國立中央大學八十八學年度碩士班研究生入學試題卷

所別: 數學研究所 不分組 科目: 資料結構 共 2 頁 第 2 頁

- (d) (5分) 使用以上結點 (Node) 的定義, 針對某一連結序列的資料結構, 定義一函式, value, 來取得第  $i$  個結點的 val 值, 假設此函數宣告部份為

```
int value( int i );    註: i 從 0 起算
```

請問若想求得此連結序列所有結點內整數的總和, 使用以下程式碼執行是否有效率? 若沒有效率, 該如何處理較好 (用文字說明既可)

```
for( i = 0 ; i < total_node_number ; ++i ) {  
    sum = sum + value(i) ;  
}
```

這裡 total\_node\_number 為結點的總數, 假設以上所有的變數都已經過適當宣告及起始設定

## 4. 堆疊排序 (heap sort)

- (a) (10分) 五個數字 {9, 7, 4, 6, 5}, 請用 heap sort 方法將這五個數字, 由小排到大, 並將過程以二元樹方式畫出
- (b) (10分) 請証明 heap sort 的運算總次數為  $O(n \log_2 n)$

## 5. 中序 (infix), 前序 (prefix), 後序 (postfix)

- (a) (10分) 判斷以下的數學形式為何種形式 (既中序, 前序, 或後序), 然後再轉換成另外兩種:

i.  $1 + 2 - 3 * 4 / 5 * 6 / 7 - 8 / 9$

ii.  $+ * - * 1 2 3 * 4 5 / * 6 7 8$

- (b) (5分) 計算機處理數學式子時, 為何以前序與後序表示較中序為佳, 其中又以後序較好

- (c) (5分) 請繪圖說明如何用 stack 將以下數學式的值求出

$$(((1 + 2) * 4) / (6 - 3))$$

## 6. 二元樹 (binary tree)

- (a) (5分) 假設某二元樹的左邊子樹 (left subtree) 共有 3000 個結點 (node), 右邊子樹 (right subtree) 共有 1000 個結點, 請問若分別用 pre-order, in-order, post-order 的方式在此二元樹結點上移動, 則根結點 (root node) 是在第幾步後 (由 1 起算) 才到達
- (b) (5分) 若以陣列 (array) 來表示完全二元樹 (complete binary tree), 今某一結點在陣列的位置為第  $i$  位 (由 0 起算), 請推導與此結點同一層且位於最左邊的結點在陣列的位置為何?
- (c) (5分) 若以二元樹來儲存數學式子,  $+ - \times /$  等運算元 (operator) 被存在非樹葉結點 (non-leaf node), 數字則存在樹葉結點 (leaf node), 今有一後序 (postfix) 的數學式,  $1 2 + 3 / 4 - 5 6 + \times$ , 可用 pre-order 的方式在某二元樹結點一一尋得, 請問此二元樹應如何擺置這個數學式?
- (d) (5分) 針對上述的數學式子, 假設二元樹的樹葉結點須儲存整數, 非樹葉結點儲存字元, 請問結點的 struct 應如何設計, 並說明如何使用

參考用