

## 程式設計：資料結構

### C 範例試卷

#### 【認證說明與注意事項】

- 一、本項考試為操作題，所需總時間為 150 分鐘，時間結束前需完成所有考試動作。成績計算滿分為 100 分，合格分數為 70 分。
- 二、操作題為五大題，第一大題至第五大題每題 20 分，總計 100 分。
- 三、操作題所需的檔案皆於 C:\ANS.CSF\各指定資料夾內讀取。題目存檔方式，請依題目指示儲存於 C:\ANS.CSF\各指定資料夾，測驗結束前必須自行存檔，並關閉開發環境，檔案名稱錯誤或未自行存檔者，均不予計分。
- 四、本項考試禁止使用外部套件與 gets() 函數，否則該題不予計分。
- 五、輸入與輸出的格式必須完全相同，每一行字、空白都要一樣，特別注意輸出後有無空白與換行。
- 六、注意全型、半型字元、英文字母大小寫、小數點的位數是否與題目的要求相同。
- 七、每一題於測驗時間內可進行重複送評。請提交程式碼檔案，而非執行檔或執行的結果。請注意提交的檔案是否適用該題目（請檢查有無交錯題目）。
- 八、每一題至少有 1 組評分測試資料顯示於題目中，且至少有 1 組隱藏的評分測試資料。
- 九、題目如有需要進行檔案讀寫，在本機撰寫程式碼自行測試時，程式開啟檔案或寫入檔案的路徑，是依據您電腦中 C 的啟動位置。在提交評分時，程式所開啟或寫入的檔案，必須與程式碼檔在同一層，例：input = fopen("read.txt", "rt")。

壹、操作題 100%(第一題至第五題每題 20 分)

## 一、二維陣列矩陣轉置運算

### 1. 題目說明：

請依照下列題意進行作答，使輸出值符合題意要求。完成結果請儲存於 C:\ANS.CSF\CP01 資料夾，檔案名稱請另存新檔為 **CPA01.c** 再進行評分。

### 2. 設計說明：

(1) 請撰寫一程式，首先要求使用者輸入矩陣的列數 (m) 與行數 (n)，代表要創建矩陣  $A$  有  $m \times n$  個元素，接著依序輸入  $m \times n$  個元素值 (皆為整數)，請計算  $A$  的轉置矩陣  $A^T$ ，並輸出兩個矩陣  $A$  與  $A^T$ 。

\* 提示：矩陣 (Matrix) 指的是外觀為  $m \times n$  的二維資料，其中  $a_{ij}$  表示矩陣中第  $i$  列第  $j$  行的元素，而位置可表示成  $(i, j)$ 。

\* 提示：矩陣  $A$  的轉置 (Transpose) 是另一個矩陣  $A^T$ ，方法是將矩陣  $A$  中每個元素從位置  $(i, j)$  轉換到  $(j, i)$ 。例如：

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}, A^T = \begin{bmatrix} 1 & 4 \\ 2 & 5 \\ 3 & 6 \end{bmatrix}$$

\* 提示：輸出矩陣時，每列皆使用中括號 ([、]) 包覆，且整數間使用半形空白間隔。

### 3. 輸入輸出：

#### (1) 輸入說明

第 1 列：輸入矩陣的列數  $m$ 。

第 2 列：輸入矩陣的行數  $n$ 。

第 3~ $m \times n + 2$  列：依序輸入矩陣元素 (皆為整數)。

#### (2) 輸出說明

1. 初始創建矩陣  $A$ 。

2. 轉置後矩陣  $A^T$ 。

#### (3) 範例輸入 1

```
2
3
1
2
3
4
5
6
```

範例輸出 1

```
Original:
[1 2 3]
[4 5 6]
Transpose:
[1 4]
[2 5]
[3 6]
```

範例輸入 2

```
3
3
-1
25
-5
33
100
-78
56
63
-99
```

範例輸出 2

```
Original:
[-1 25 -5]
[33 100 -78]
[56 63 -99]
Transpose:
[-1 33 56]
[25 100 63]
[-5 -78 -99]
```

4. 評分項目：

| 項   | 目            | 配 分 | 得 分 |
|-----|--------------|-----|-----|
| (1) | 符合設計說明輸出正確格式 | 20  |     |
| 總   | 分            | 20  |     |

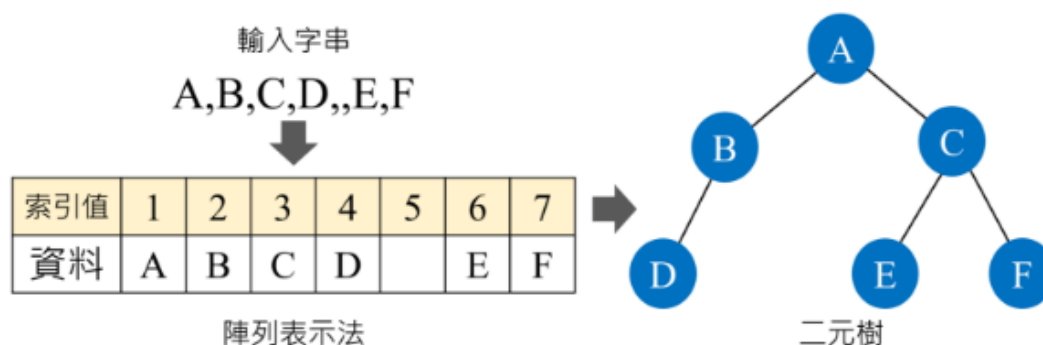
## 二、二元樹的建立與走訪

### 1. 題目說明：

請依照下列題意進行作答，使輸出值符合題意要求。完成結果請儲存於 C:\ANS.CSF\CP02 資料夾，檔案名稱請另存新檔為 **CPA02.c** 再進行評分。

### 2. 設計說明：

- (1) 撰寫程式讓使用者輸入一個以陣列（Array）表示完全二元樹（Complete Binary Tree）的字串，例如：「A,B,C,D,,E,F」，其中節點（Node）以大寫英文字母表示，且節點與節點間以半形逗號（,）隔開，轉換方式如下圖。



- (2) 針對轉換後的二元樹，請以中序走訪（Inorder）依序輸出所有節點，並按照「字母順序」輸出此二元樹的所有葉節點（Leaf node）。

\* 提示：節點數量小於 20 個。

### 3. 輸入輸出：

#### (1) 輸入說明

一個代表完全二元樹的字串，節點以大寫英文字母表示，且節點與節點間以半形逗號（,）間隔。

#### (2) 輸出說明

第 1 列：以中序走訪依序輸出所有節點。

第 2 列：依照「字母順序」輸出此二元樹的所有葉節點。

#### (3) 範例輸入 1

A,B,C,D,,E,F

範例輸出 1

DBAECF

DEF

範例輸入 2

A,B,C,D,E,,,,F,G

範例輸出 2

DBFEGAC

CDFG

4. 評分項目：

| 項   | 目            | 配 分 | 得 分 |
|-----|--------------|-----|-----|
| (1) | 符合設計說明輸出正確格式 | 20  |     |
| 總   | 分            | 20  |     |

### 三、氣泡排序

#### 1. 題目說明：

請依照下列題意進行作答，使輸出值符合題意要求。完成結果請儲存於  
C:\ANS.CSF\CP03 資料夾，檔案名稱請另存新檔為 **CPA03.c** 再進行評分。

#### 2. 設計說明：

(1) 請撰寫一程式，首先要求使用者輸入正整數  $N$  ( $1 \leq N \leq 20$ )，代表有  $N$  個排序資料，接著依序輸入  $N$  個整數，每個數字之間以半形空白間隔，請將這些資料採「氣泡排序 (Bubble Sort) 演算法」由小到大排序，並輸出原始數列與每次循環排序的結果。

#### 3. 輸入輸出：

##### (1) 輸入說明

第 1 列：輸入正整數  $N$  ( $1 \leq N \leq 20$ ) 為排序個數。

第 2 列：依序輸入  $N$  個整數，整數間以半形空白間隔。

##### (2) 輸出說明

先輸出原始數列，接著依序換列輸出每次循環排序的結果，整數間以半形空白間隔。

(排序完成即停止，若輸入之數列已經是排序結果 (沒有發生交換)，輸出僅顯示「原始數列」)

##### (3) 範例輸入 1

```
5
2 52 65 4 3
```

##### 範例輸出 1

```
2 52 65 4 3
2 52 4 3 65
2 4 3 52 65
2 3 4 52 65
```

##### 範例輸入 2

```
7
1 2 3 4 5 6 7
```

##### 範例輸出 2

```
1 2 3 4 5 6 7
```

#### 4. 評分項目：

| 項   | 目            | 配 分 | 得 分 |
|-----|--------------|-----|-----|
| (1) | 符合設計說明輸出正確格式 | 20  |     |
| 總   | 分            | 20  |     |

#### 四、一筆畫遊戲

##### 1. 題目說明：

請依照下列題意進行作答，使輸出值符合題意要求。完成結果請儲存於 C:\ANS.CSF\CP04 資料夾，檔案名稱請另存新檔為 **CPA04.c** 再進行評分。

##### 2. 設計說明：

- (1) 尤拉路徑是圖論 (Graph Theory) 中被廣為人知的一個經典問題，又被稱為「一筆畫問題」，起源於哥尼斯堡七橋問題；數學家尤拉 (Euler) 在 1736 年解開了七橋問題並提出相關結論與定理：如果能夠從圖上某一點當作起點出發，找出一條路徑可以通過圖上所有邊恰好一次，就好像用畫筆可以連續地畫過該圖形的所有邊，該路徑就稱為「尤拉路徑 (Euler path)」。
- (2) 以下是尤拉對於一筆畫問題的重點結論：「一個連通圖中，所有節點的分支度都是偶數，或只有 2 個節點的分支度是奇數，則該圖就可以用一筆畫完成；否則無法一筆畫。」
- (3) 一個無向圖 (Undirected Graph) 通常表示為  $G=(V,E)$ ，其中  $V$  為節點 (Vertex) 的集合， $E$  為邊 (Edge) 的集合，且任一個邊是由兩個相連接的節點所構成，當標示為  $(a,b)$  即表示節點  $a$  與  $b$  之間有邊連接。
- (4) 請設計一支程式，可以讀取一個無向圖的資料 (邊的列表)，然後判斷該圖形是否能夠一筆畫完成，並輸出圖形的相鄰矩陣結果。(注意：題目給的圖形一定是「連通圖」)。
  - \* 提示：任一節點的「分支度 (Degree)」為該節點連接的邊數、亦即是對外所連接的節點數。
  - \* 提示：「連通圖 (Connected Graph)」中的任兩個節點必存在一條路徑相通；反之則為「非連通圖」。
  - \* 提示：可使用「二維陣列 (Two-Dimensional Array)」資料結構來實作「相鄰矩陣 (Adjacency Matrix)」以紀錄圖形中節點與節點的關係 (邊)。

##### 3. 輸入輸出：

###### (1) 輸入說明

第 1 列：包含兩個正整數  $M$ 、 $N$ ，並以半形空白間隔，其中  $M$  為節點數、 $N$  為邊數。

第 2~ $N+1$  列：每一列包含兩個正整數  $a$ 、 $b$ ，並以半形空白間隔，代表圖形的某一個邊連接節點  $a$  與  $b$ ，且  $1 \leq a, b \leq M$ 。

###### (2) 輸出說明

第 1 列：若圖形可以「一筆畫完成」，則輸出「YES」；否則輸出「NO」。

第 2~ $M+1$  列：輸出圖形的相鄰矩陣，共有  $M$  列由 0 與 1 所構成的資料列，每一列有  $M$  個數字，請使用「1」表示列、行相對應的節點之間有邊；使用「0」表示沒有對應邊的存在，每個數字之間須以半形空白間隔。

(3) 範例輸入 1

```
3 3
1 2
1 3
3 2
```

範例輸出 1

```
YES
0 1 1
1 0 1
1 1 0
```

範例輸入 2

```
5 8
1 2
1 3
1 4
1 5
2 5
3 2
3 4
5 4
```

範例輸出 2

```
NO
0 1 1 1 1
1 0 1 0 1
1 1 0 1 0
1 0 1 0 1
1 1 0 1 0
```

4. 評分項目：

| 項   | 目            | 配 分 | 得 分 |
|-----|--------------|-----|-----|
| (1) | 符合設計說明輸出正確格式 | 20  |     |
| 總   | 分            | 20  |     |

## 五、失落的數字寶藏

### 1. 題目說明：

請依照下列題意進行作答，使輸出值符合題意要求。完成結果請儲存於 C:\ANS.CSF\CP05 資料夾，檔案名稱請另存新檔為 **CPA05.c** 再進行評分。

### 2. 設計說明：

- (1) 在一個古老的數字世界中，有一個傳說，講述著一個隱藏的數字寶藏，它包含了一系列的數字，而寶藏就隱藏在序列中「數字和最大」的連續子序列中。
- (2) 傳說中有一位勇敢的探險者，他決心要找到這個隱藏的數字寶藏。但是，他面對的挑戰是這個數字序列非常長，包含了正整數、負整數或 0。探險者需要一個特殊的工具來快速找到這個最有價值的子序列。
- (3) 你的任務是幫助這位探險者，設計程式接收一個包含多個整數的序列，然後找出這個序列中具有最大數字和的連續子序列。找到這個子序列就等於找到了傳說中的數字寶藏。

### 3. 輸入輸出：

#### (1) 輸入說明

一個包含多個整數的序列，整數間以半形空白間隔。

#### (2) 輸出說明

第 1 列：最大連續子序列的總和。

第 2 列：最大連續子序列和的起始位置（以陣列 0 開始）。

第 3 列：最大連續子序列和的結束位置（以陣列 0 開始）。

#### (3) 範例輸入 1

-4 1 2 9 -5

#### 範例輸出 1

12

1

3

#### 範例輸入 2

-3 4 7 -1 10 0 -2 -8 8 1 -3 0 7 -8 1 9 3 -4 10 -2

#### 範例輸出 2

34

1

18

### 4. 評分項目：

| 項   | 目            | 配 分 | 得 分 |
|-----|--------------|-----|-----|
| (1) | 符合設計說明輸出正確格式 | 20  |     |
| 總   | 分            | 20  |     |