

# 110 計算機程式設計

## (Computer Programming 1279)

111.01.03

上機作業、考試繳交方式：

每次上課都會有上機作業，必須在課堂上完成，有 1~2 小時可用，在下課前上傳到 I-learning，上傳時只需要上傳 **.c 檔**。除非有事先請假，否則不接受事後補交！

上機作業與考試的格式：

1. 依照下列格式命名：`學號_p#.c` (# => 題號)

例：

**基本題 第一題** => 4109056099\_**p1.c**

**加分題 第一題** => 4109056099\_**b1.c**

2. 上傳後可以上傳修改版本(在遲交期限以前)，如果想要上傳更新的版本，檔案名稱維持一樣直接上傳，助教只會對最新的版本評分。

3. 程式內開頭第 1 行到第 4 行，以註解的方式寫下列資訊：

```
/*  
*   系級：  
*   學號：  
*   姓名：  
*   聯絡電子郵件：  
*/
```

4. 程式輸入輸出部分有底線的為 input 沒有的 output

例: 100 (100 為輸入)

212.00 (212.00 為輸出)

5. 請注意 不要使用下列兩個 function 用了作業會錯

System("pause")

fflush()

- 6. 每筆測資限制時間為 8 秒**

### Hw13 基本題：

1. 依序輸入 N 個檔案名，試著開啟 N 個檔案。(25%)
  - (1). 如果成功開啟，則於螢幕上印 "<檔案名> opened successfully"
  - (2). 如果無法開啟，則於螢幕上印 "<檔案名> opened failed"

題目規範：

- $1 \leq N \leq 10$
- 檔案名稱長度最多 500 個字元(可以 char 陣列處理)
- 輸入格式：  
N  
檔案 1  
檔案 2  
...  
檔案 N

程式輸出範例：

輸入：

```
7
testfile\p1_ex1
testfile\p1_ex2.in
testfile\p1_ex3.out
testfile\p1_ex4.c
testfile\p1_ex5.txt
testfile\p1_ex6.txt
testfile\p1_ex7.txt
```

輸出：

```
testfile\ p1_ex1 opened successfully
testfile\ p1_ex2.in opened successfully
testfile\ p1_ex3.out opened successfully
testfile\ p1_ex4.c opened successfully
testfile\ p1_ex5.txt opened successfully
testfile\ p1_ex6.txt opened failed
testfile\ p1_ex7.txt opened failed
```

2. 輸入一個檔案名 F，讀取檔案 F 中的資料，並將 F 中每一行的兩數相加，並輸出答案(印至螢幕上)。(25%)

題目規範：

- 檔案 F 名稱長度最多 500 個字元(可以 char 陣列處理)
- 檔案輸入以 EOF 結尾
- 數字皆以 int 格式處理即可
- 輸入格式：(檔案 F 中)

$a_1 \ b_1$

$a_2 \ b_2$

$a_3 \ b_3$

...

- 輸出格式：(螢幕上)

$a_1 + b_1$

$a_2 + b_2$

$a_3 + b_3$

...

程式輸入輸出範例：

輸入：

testfile\p2\_ex1.in

輸出(螢幕上)：

3

5

7

9

11

3. 輸入一個檔案名 F，並輸入連續數個字元，將這些字元寫入檔案 F 儲存。  
(25%)

題目規範：

- 檔案 F 名稱長度最多 500 個字元(可以 char 陣列處理)
- 字元僅包含大小寫字母和各式標點符號，以'\n'結尾
- 字元數沒有限制，保證系統可執行
- 輸入格式：  
檔案 F  
數個字元

程式輸入輸出範例 1：

輸入：

p3\_001.out

"I think, therefore I am."

輸出(檔案 p3\_001.out 中)：

"I think, therefore I am."

程式輸入輸出範例 2：

輸入：

p3\_002.out

All or nothing, now or never.

輸出(檔案 p3\_002.out 中)：

All or nothing, now or never.

4. 輸入一個正整數  $N$  代表  $N$  位學生，假設每位學生各有名字和三科考試成績，分別為程式設計、微積分、機率，請根據之前學過的 `malloc` 和指標，以動態宣告的方式宣告結構陣列並輸入值，接著寫一個函式 `void sort(int n, struct data** students)` 將結構傳入此函式，輸出結果為三科成績的總和由大到小排序。函式參數依序為  $N$  位學生、宣告的結構。

題目規範：

- 請使用以下方式宣告 structure:

```
struct data {  
    char name[100];           //學生姓名  
    int CP;                   //程式設計成績  
    int calculus;             //微積分成績  
    int probability;          //機率成績  
};
```

- $1 \leq N \leq 1000$

- 總和一樣，則依其輸入順序先後排序。

程式輸入輸出範例 1：

輸入：

3

Alice 0 0 0

Bob 50 50 100

Charlie 26 58 66

輸出：

Bob 50 50 100

Charlie 26 58 66

Alice 0 0 0

程式輸入輸出範例 2：

輸入：

5

Alice 84 21 61

Bob 74 31 51

Charlie 62 10 0

David 69 99 47

Frank 32 14 65

輸出:

David 69 99 47

Alice 84 21 61

Bob 74 31 51

Frank 32 14 65

Charlie 62 10 0

### Hw13 加分題：

1. 輸入三個檔案名 F、G 和 H，從 F 和 G 讀取輸入，F 和 G 分別放了一行字母，要求把這兩行字母合併，並按字母順序排列，將結果寫入 H。(10%)

題目規範：

- 檔案名稱長度最多 500 個字元(可以 char 陣列處理)
- F 和 G 中字母數不限
- F 和 G 中字母皆小寫，皆以'\n'結尾
- 字母總數不超過 int 上限

程式輸入輸出範例：

輸入：

testfile\b1\_001\_F.in

testfile\b1\_001\_G.in

testfile\b1\_001\_H.out

輸出(檔案 b1\_001\_H.out 中)：

abcddeeffgghijkl

2. 給定一字串  $S$ ， $S$  僅包含  $(、)、[、]、{$  和  $}$ ，6 種括號，請判斷字串  $S$  是否是一串合法的括號排序。(10%)

題目規範：

- $1 \leq S \text{ 長度} \leq 100000$
- 輸入以 '\n' 結尾
- 判斷是否為合法括號排序：
  - (1). 左括號必須用相同類型的右括號閉合。
  - (2). 左括號必須以正確的順序閉合。

程式輸入輸出範例 1：

()  
Yes

程式輸入輸出範例 2：

()[]{}  
Yes

程式輸入輸出範例 3：

()  
No

程式輸入輸出範例 4：

(())  
No

程式輸入輸出範例 5：

{[]}  
Yes