

# Turnin作業9

- chapter 8
- Turnin Code: **c.hw9**
- Due Date: 2025/12/09 23:59 **Hard Deadline**

## 繳交方式說明

本次Turnin作業包含7個程式題，建議可以為它們建立一個資料夾，並將此次作業所有要上傳的檔案放置於該資料夾後，再使用 `turnin` 指令上傳作業。例如下面的操作在同學的家目錄裡建立了一個名為hw9的資料夾，並且切換到該資料夾後，針對第1題所要求的Performance.c進程式編撰：

```
[user@ws ~]$ mkdir hw9
[user@ws ~]$ cd hw9
[user@ws hw9]$ joe Performance.c
```

等到我們完成Performance.c的撰寫後，請自行加以編譯與執程式，確認正確後使用 `turnin c.hw9 Performance.c` 指令將加以上傳：

```
[user@ws c.hw9]$ turnin c.hw9 Performance.c
Turning in:
Performance.c -- ok
All done.
[user@ws c.hw9]$
```

當然，你也可以等到本次作業要求的所有題目都在hw9資料夾裡完成後，再使用 `turnin c.hw9 .` 指令一次將所有在目前資料夾中的所有檔案都加以上傳。假設你已經在hw9資料夾裡撰寫完所有題目，並且每個題目的程式檔案皆已編譯並確認執行結果正確後，我們可以使用以下指令將多餘的(不需要繳交的)檔案加以刪除後，一次將所有檔案繳交：

```
[user@ws c.hw9]$ ls
Cashier.c           IsSparseMatrix.c  PokerEvaluator.c
NumberComparison.c MaximumSubarray.c  InteractiveSort.c
Performance.c       a.out
[user@ws c.hw9]$ rm a.out
[user@ws c.hw9]$ ls
Cashier.c           IsSparseMatrix.c  PokerEvaluator.c
NumberComparison.c MaximumSubarray.c  InteractiveSort.c
Performance.c
```

```
[user@ws c.hw9]$ turnin c.hw9 .
Turning in:
./Performance.c -- ok
./IsSparseMatrix.c -- ok
./MaximumSubarray.c -- ok
./InteractiveSort.c -- ok
./NumberComparison.c -- ok
./PokerEvaluator.c -- ok
./Cashier.c -- ok
All done.
[user@ws c.hw9]$
```

如果繳交後想要查看已繳交的檔案及相關資訊，可以輸入`turnin -ls c.hw9` 指令，例如：

```
[user@ws c.hw9] c.hw9 $ turnin -ls c.hw9
.:
total 32
-rw-rw----. 1 turninman turnin 835 Dec 2 13:11 Cashier.c
-rw-rw----. 1 turninman turnin 1826 Dec 2 13:11 NumberComparison.c
-rw-rw----. 1 turninman turnin 2620 Dec 2 13:11 InteractiveSort.c
-rw-rw----. 1 turninman turnin 772 Dec 2 13:11 IsSparseMatrix.c
-rw-rw----. 1 turninman turnin 2353 Dec 2 13:11 MaximumSubarray.c
-rw-rw----. 1 turninman turnin 1799 Dec 2 13:11 Performance.c
-rw-rw----. 1 turninman turnin 4439 Dec 2 13:11 PokerEvaluator.c
```

註：本文使用  及  代表空白字元與Enter換行字元，並且將使用者輸入的部份使用灰階方式顯示。另外，出現在程式執行結果中的符號皆為半形。

## P1 我親愛的學費 Performance.c

著名的可愛職業訓練班招收非資訊系大學畢業生，經受訓後輔導學員投考相關職業證照。每位學員以三個月為一期，報名時必須選擇至少3門課、至多7門課，學員必須於每門課結束時報名參加證照檢定。勞委會職訓局為了統計可愛職業訓練班的收費情形與輔導考照績效，委託你為學員設計一個結訓成果計算程式Performance.c讓學員輸入本期所繳交的總學費(不同學員方案不同，或有修習同樣課程但收費不同之情況)，並輸入所修習之課程代碼與是否取得該課程相關證照後，計算輸出該學員本期修課績效。以下就此程式的輸出與輸出要求分項加以說明：

【輸入】：

- 學員輸入本期總學費（整數），若輸入小於等於0的數值或是輸入非數字則輸出Error!並終止程式。
- 輸入本期所修習之課程的課程代號A~G及是否通過相關狀態Y/N
  - 每位學員每期至少須修習3門課、至多7門課。
  - 每個課程代號只能輸入一次，否則程式輸出 Error!並終止程式。
  - 課程代號必須在 A~G 範圍內，否則輸出 Error!並終止程式。
  - 通過狀態必須為 Y 或 N否則輸出 Error!並終止程式。

- 若輸入重複課程代號，輸出 Error! 並終止程式。
- 若已完成輸入3門課的輸入後，可輸入Q提前結束輸入。

**【輸出】：**

取得學員的輸入後，請計算並顯示所繳交的總學費、通過證照數、平均每課程費用（四捨五入到小數點後兩位），並顯示該學員取得每張證照的平均成本（四捨五入到小數點後兩位）。舉例來說，某甲本期繳交10000元修習4門課，最終通過並取得2張證照，在此情況下，其總學費、通過證照數、平均每課程費用與取得每張證照的平均成本分別為10000, 2, 2500與5000。

**【課程表】：**

課程代號	課程名稱	相關證照
A	C程式設計	程式設計丙級檢定
B	C plus plus 程式設計	C plus plus certificate
C	資料庫應用	SQLPlus
D	網頁設計與應用	web design certificate
E	線性代數	linear algebra certificate
F	視覺化程式設計	Visual Programming certificate
G	RFID導論	RFID Introduction certificate

**【註：本題有使用浮點數運算，請使用double型態。】**

輸出結果可以參考如下：

```
[3:23 user@ws hw] ./a.out
Input total fee: 10000
Input course(A-G) and pass status(Y/N), or Q to quit: A Y
Input course(A-G) and pass status(Y/N), or Q to quit: C N
Input course(A-G) and pass status(Y/N), or Q to quit: E Y
Input course(A-G) and pass status(Y/N), or Q to quit: F N
Input course(A-G) and pass status(Y/N), or Q to quit: Q
Total fee: 10000
Passed certificates: 2
Average per course: 2500.00
Average per certificate: 5000.00
[3:23 user@ws hw] ./a.out
Input total fee: 7500
Input course(A-G) and pass status(Y/N), or Q to quit: B Y
Input course(A-G) and pass status(Y/N), or Q to quit: D Y
Input course(A-G) and pass status(Y/N), or Q to quit: G Y
Input course(A-G) and pass status(Y/N), or Q to quit: Q
Total fee: 7500
Passed certificates: 3
Average per course: 2500.00
Average per certificate: 2500.00
[3:23 user@ws hw] ./a.out
Input total fee: 20000
Input course(A-G) and pass status(Y/N), or Q to quit: A N
```

```

Input▲course▲(A-G)▲and▲pass▲status▲(Y/N),▲or▲Q▲to▲quit:▲B▲Y↵
Input▲course▲(A-G)▲and▲pass▲status▲(Y/N),▲or▲Q▲to▲quit:▲C▲N↵
Input▲course▲(A-G)▲and▲pass▲status▲(Y/N),▲or▲Q▲to▲quit:▲D▲Y↵
Input▲course▲(A-G)▲and▲pass▲status▲(Y/N),▲or▲Q▲to▲quit:▲E▲Y↵
Input▲course▲(A-G)▲and▲pass▲status▲(Y/N),▲or▲Q▲to▲quit:▲F▲N↵
Input▲course▲(A-G)▲and▲pass▲status▲(Y/N),▲or▲Q▲to▲quit:▲G▲Y↵
Total▲fee:▲20000↵
Passed▲certificates:▲4↵
Average▲per▲course:▲2857.14↵
Average▲per▲certificate:▲5000.00↵
[3:23 user@ws hw] ./a.out↵
Input▲total▲fee:▲8000↵
Input▲course▲(A-G)▲and▲pass▲status▲(Y/N),▲or▲Q▲to▲quit:▲B▲T↵
Error!↵
[3:23 user@ws hw] ./a.out↵
Input▲total▲fee:▲12000↵
Input▲course▲(A-G)▲and▲pass▲status▲(Y/N),▲or▲Q▲to▲quit:▲A▲Y↵
Input▲course▲(A-G)▲and▲pass▲status▲(Y/N),▲or▲Q▲to▲quit:▲A▲N↵
Error!↵
[3:23 user@ws hw] ./a.out↵
Input▲total▲fee:▲9000↵
Input▲course▲(A-G)▲and▲pass▲status▲(Y/N),▲or▲Q▲to▲quit:▲A▲Y↵
Input▲course▲(A-G)▲and▲pass▲status▲(Y/N),▲or▲Q▲to▲quit:▲Q↵
Error!↵
[3:23 user@ws hw] ./a.out↵
Input▲total▲fee:▲0↵
Error!↵
[3:23 user@ws hw]

```

## P2 找零 `Cashier.c`

有一臺收銀機，它的功能之一是會將要找零的硬幣數量還有紙鈔數量告訴收銀員。假設該國發行的貨幣有以下面額：

- 1 元
- 5 元
- 10 元
- 50 元
- 100 元
- 500 元
- 1000 元

請設計一個C語言的程式 `Cashier.c`讓使用者輸入應付與已付金額（皆為正整數），若輸入非正整數的字元則輸出`Error!`並跳出程式，輸出所需找零的最少紙鈔與硬幣數量（假設硬幣數量及紙鈔數量充足），其執行結果可參考以下輸出內容：

```

[3:23 user@ws hw] ./a.out↵

```

```
Enter the total amount: 100
Enter the amount paid: 500
1 > 0
5 > 0
10 > 0
50 > 0
100 > 4
500 > 0
1000 > 0
[3:23 user@ws hw] ./a.out
Enter the total amount: 263
Enter the amount paid: 1000
1 > 2
5 > 1
10 > 3
50 > 0
100 > 2
500 > 1
1000 > 0
[3:23 user@ws hw] ./a.out
Enter the total amount: 1000
Enter the amount paid: 1000
1 > 0
5 > 0
10 > 0
50 > 0
100 > 0
500 > 0
1000 > 0
[3:23 user@ws hw] ./a.out
Enter the total amount: -1
Enter the amount paid: 1000
Error!
[3:23 user@ws hw] ./a.out
Enter the total amount: 1000
Enter the amount paid: -1
Error!
[3:23 user@ws hw] ./a.out
Enter the total amount: 1000
Enter the amount paid: 500
Error!
[3:23 user@ws hw] ./a.out
Enter the total amount: q
Error!
[3:23 user@ws hw] ./a.out
Enter the total amount: 100
Enter the amount paid: q
Error!
```

[3:23 user@ws hw]

【提示：本題可以將鈔票及貨幣依面額由大到小存放到一個陣列中，然後使用迴圈的第i個回合裡，將欲找零之金額除以陣列的第i個元素，其商數即為該面額所應找零的數目、其餘數則進行迴圈的下一回合繼續運算。】

### P3 最大連續區間的總和 MaximumSubarray.c

請設計一個C語言的程式 MaximumSubarray.c 使用者將輸入一個整數陣列，陣列共有10個數字元素，如果使用者輸入非數字則輸出Error! 請你找出其中一段「連續」的元素，使它們加起來的數字最大，並輸出這個最大總和。



【助教的小提示】：假設我們將使用者所輸入的10個整數放在陣列data[]裡，為了要找出在data[]陣列中總合最大的連續元素，我們可以假設具有最大總合的連續元素是從索引位置i開始到索引位置j為止，因此程式的目的就是要找出i與j的數值為何。為此我們將可以使用巢狀迴圈，依序從外層的i=0, 1, 2, ...9以及內層的j=i,i+1, ...,9逐一地將從data[i]到data[j]範圍內的元素加總，並將其結果保留在另一個二維陣列sum[10][10]裡。舉例來說sum[3][5]將存放data[3]+data[4]+data[5]的和, sum[5][9]則將存放data[5]+data[6]+data[7]+data[8]+data[9]的和。完成sum[][]陣列的計算後，再從其中找出最大的數值即為所求。

```
[3:23 user@ws hw] ./a.out↵
Please enter 10 integers:↵
10 2 55 -1 -7 -2 1 66 2 -35↵
Maximum subarray sum: 126↵
[3:23 user@ws hw] ./a.out↵
Please enter 10 integers:↵
0 0 -1 22 66 1 22 773 1 2↵
Maximum subarray sum: 887↵
[3:23 user@ws hw] ./a.out↵
Please enter 10 integers:↵
-1 -5 -2 -7 -2 -1 -6 -7 -2 -1↵
Maximum subarray sum: -1↵
[3:23 user@ws hw] ./a.out↵
Please enter 10 integers:↵
a 1 3 6 1 3 6 1 0 8↵
Error!↵
[3:23 user@ws hw]
```

## P4 判斷稀疏矩陣 `IsSparseMatrix.c` (此處首字母為大寫的 I(丕))

稀疏矩陣是一個矩陣中零元素的數量多於非零元素的矩陣。

請設計一個 C 語言的程式 `IsSparseMatrix.c` (使用者將輸入  $3 \times 3$  的矩陣且每個元素都是整數 (如果輸入非數字則輸出 `Error!` 並結束程式) )，最後輸出該矩陣是否為稀疏矩陣，其執行結果可參考以下輸出內容：

```
[3:23 user@ws hw] ./a.out↵
Enter data:↵
1 2 3↵
0 0 0↵
1 2 3↵
It is not a sparse matrix!↵
[3:23 user@ws hw] ./a.out↵
Enter data:↵
1 0 1↵
0 1 0↵
0 0 0↵
It is a sparse matrix!↵
[3:23 user@ws hw] ./a.out↵
Enter data:↵
1 a 1↵
Error!↵
[3:23 user@ws hw]
```

## P5 1A2B 數字比對 `NumberComparison.c`

請設計一個 C 語言程式 `GuessNumber.c` 讓使用者進行 1A2B 數字比對遊戲。系統不會隨機產生答案，而是由使用者自行輸入一個 4 位數、且每個數字不可重複的整數作為正確答案。接著玩家會不斷輸入猜測的 4 位數字 (如果輸入的數值非數字，或是輸入四個相同的數字，則輸出 `Invalid input!` 並由程式判斷其結果，判斷後輸出幾 A 幾 B 直到輸入正確答案。

代號	判斷邏輯
A	位置正確以及數字正確
B	位置錯誤但數字正確

當猜中 4A0B 時，遊戲結束。

本題的輸出結果可參考如下：

```
[3:23 user@ws hw] ./a.out↵
Enter answer: 1234↵
```

```

Guessaa4-digitnumber:2143
0A4B
Guessaa4-digitnumber:1547
1A1B
Guessaa4-digitnumber:3214
2A2B
Guessaa4-digitnumber:1234
4A0B
Correct!Theansweris1234.
[3:23 user@ws hw] ./a.out
Enteranswer:5678
Guessaa4-digitnumber:1234
0A0B
Guessaa4-digitnumber:1238
1A0B
Guessaa4-digitnumber:0000
Invalidinput!
Guessaa4-digitnumber:a1b3
Invalidinput!
Guessaa4-digitnumber:5678
4A0B
Correct!Theansweris5678.
[3:23 user@ws hw] ./a.out
Enteranswer:1a23
Invalidanswer!
[3:23 user@ws hw]

```

## P6 互動式排序程式 InteractiveSort.c

請設計一個 C 語言程式，使用 泡沫排序法 Bubble sort 來處理陣列排序。程式需具備互動功能，並滿足以下要求：

### 【第一部分：輸入 10 個整數】

程式啟動後，要求使用者輸入 10 個整數以存入陣列。

如果輸入的內容不是整數（例如輸入字母、符號），則顯示 Error! 並且要求使用者重新輸入整個陣列。

### 【第二部分：輸入指令】

成功輸入 10 個不重複的整數後，程式必須提示使用者輸入以下指令：

指令	功能
1	對陣列進行升序排列，並輸出結果。
2	對陣列進行降序排列，並輸出結果。
3	重新輸入 10 個整數（回到第一部分）。
q	結束此程式。

使用者輸入指令時若輸入 1、2、3 或 q 則執行相對應的功能。如果輸入其他字元或數值，則程式需顯示 Error! 並且要求使用者重新輸入指令。

此題的執行結果可參考如下：

```
[3:23 user@ws hw] ./a.out↵
Please input 10 integers:↵
12 5 8 3 20 1 15 7 10 0↵
Please input command(1 ascending, 2 descending, 3 re-enter, q quit):↵
1↵
Ascending result: 0 1 3 5 7 8 10 12 15 20↵
Please input command(1 ascending, 2 descending, 3 re-enter, q quit):↵
q↵
[3:23 user@ws hw] ./a.out↵
Please input 10 integers:↵
50 23 15 78 12 34 56 67 89 10↵
Please input command(1 ascending, 2 descending, 3 re-enter, q quit):↵
2↵
Descending result: 89 78 67 56 50 34 23 15 12 10↵
Please input command(1 ascending, 2 descending, 3 re-enter, q quit):↵
q↵
[3:23 user@ws hw] ./a.out↵
Please input 10 integers:↵
12 5 8 3 20 1 15 7 10 0↵
Please input command(1 ascending, 2 descending, 3 re-enter, q quit):↵
1↵
Ascending result: 0 1 3 5 7 8 10 12 15 20↵
Please input command(1 ascending, 2 descending, 3 re-enter, q quit):↵
3↵
Please input 10 integers:↵
89 78 67 56 50 34 23 15 12 10↵
Please input command(1 ascending, 2 descending, 3 re-enter, q quit):↵
1↵
Ascending result: 10 12 15 23 34 50 56 67 78 89↵
Please input command(1 ascending, 2 descending, 3 re-enter, q quit):↵
q↵
[3:23 user@ws hw] ./a.out↵
Please input 10 integers:↵
2 5 1 2 5 2 1 4 2 1 5↵
Please input command(1 ascending, 2 descending, 3 re-enter, q quit):↵
q↵
[3:23 user@ws hw] ./a.out↵
Please input 10 integers:↵
1 2 a 4 5 6 7 8 9 10↵
Error!↵
Please input 10 integers:↵
1 2 3 1 2 4 11 5 1 5↵
Please input command(1 ascending, 2 descending, 3 re-enter, q quit):↵
1↵
Ascending result: 1 1 1 2 2 3 4 5 5 11↵
Please input command(1 ascending, 2 descending, 3 re-enter, q quit):↵
```

```

q↵
[3:23 user@ws hw] ./a.out↵
Please input 10 integers:↵
1 2 3 4 5 6 7 8 9 10↵
Please input a command (1 ascending, 2 descending, 3 re-enter, q quit):↵
2↵
Descending result: 10 9 8 7 6 5 4 3 2 1↵
Please input a command (1 ascending, 2 descending, 3 re-enter, q quit):↵
5↵
Error!↵
Please input a command (1 ascending, 2 descending, 3 re-enter, q quit):↵
q↵
[3:23 user@ws hw] ./a.out↵
Please input 10 integers:↵
1 5 1 5 7 7 3 1 2 6 1 7 @↵
Error!↵
Please input 10 integers:↵
1 2 5 2 1 6 2 5 7 1 3↵
Please input a command (1 ascending, 2 descending, 3 re-enter, q quit):↵
1↵
Ascending result: 1 1 2 2 3 5 5 6 7 2 1↵
Please input a command (1 ascending, 2 descending, 3 re-enter, q quit):↵
q↵
[3:23 user@ws hw]

```

## P7 互動式撲克牌分析器 `PokerEvaluator.c`

請設計一個 C 語言程式撲克牌分析器 `PokerEvaluator.c`。程式開始執行後，將會持續要求使用者輸入手上的撲克牌，直到使用者輸入 `q` 為止。使用者所輸入的撲克牌是由「點數」與「花色」組成，其中點數包含 `A, 2, 3, 4, ..., 9, T, J, Q, K`。花色方面則使用 `S, H, D` 與 `C` 來代表 `Spades` (黑桃)、`Hearts` (紅心)、`Diamonds` (方塊) 與 `Clubs` (梅花)。若使用者所輸入的撲克牌的點數或花色不正確，程式必須顯示 `Error!` 並要求使用者重新輸入一次牌組。若是的撲克牌有重複的牌，則也請顯示 `Error!` 並要求使用者重新輸入一次牌組。本題必須依據下表判斷使用者所輸入的撲克牌屬於何種牌型：

牌型	說明
同花順 <code>Straight Flush</code>	點數連續且同花色
鐵支 <code>Four of a Kind</code>	4 張相同點數
葫蘆 <code>Full House</code>	3 張相同點數 + 2 張相同點數
同花 <code>Flush</code>	5 張同花色，但非連續
順子 <code>Straight</code>	5 張點數連續，但非同花
其他 <code>High Card</code>	以上皆不符合

此題執行結果可參考以下輸出：

【註：本題輸出結果有部分註解，請忽略註解的輸出。】

```

[3:23 user@ws hw] ./a.out↵
Please input a hand of 5 cards (or q to quit):↵
HT S2 DJ CA D7↵
High Card↵
Please input a hand of 5 cards (or q to quit):↵
H2 H3 H4 H5 H6↵
Straight Flush↵
Please input a hand of 5 cards (or q to quit):↵
HA H2 H4 H3 H5↵
Straight Flush↵
Please input a hand of 5 cards (or q to quit):↵
HT HA HJ HK HQ↵
Straight Flush↵
Please input a hand of 5 cards (or q to quit):↵
C9 D9 H9 S9 D3↵
Four of a Kind↵
Please input a hand of 5 cards (or q to quit):↵
HK CK SK D8 S8↵
Full House↵
Please input a hand of 5 cards (or q to quit):↵
S2 S5 S9 SJ SK↵
Flush↵
Please input a hand of 5 cards (or q to quit):↵
D4 H5 S6 C7 D8↵
Straight↵
Please input a hand of 5 cards (or q to quit):↵
q↵
[3:23 user@ws hw] ./a.out↵
Please input a hand of 5 cards (or q to quit):↵
HT S2 DJ CA HT (註解)#重複牌
Error!↵
Please input a hand of 5 cards (or q to quit):↵
HT S2 DJ CA X7 (註解)#非法花色
Error!↵
Please input a hand of 5 cards (or q to quit):↵
HT S2 DH CA D7 (註解)#非法點數
Error!↵
Please input a hand of 5 cards (or q to quit):↵
q↵
[3:23 user@ws hw]

```

From:

<https://junwu.nptu.edu.tw/dokuwiki/> - Jun Wu的教學網頁

國立屏東大學資訊工程學系

**CSIE, NPTU**

Total: 293174

Permanent link:

<https://junwu.nptu.edu.tw/dokuwiki/doku.php?id=c:2025fall-hw9>

Last update: **2025/12/09 13:57**

