

## 實驗項目- 檔案讀寫 fwrite、fread 使用

### 一、本節目的：

- 學習使用 fwrite、fread 方法
- 比較一般文字檔與二進位檔案

### 二、設計重點：

- 利用 fscanf、fwrite、fread 方法來完成讀寫檔案

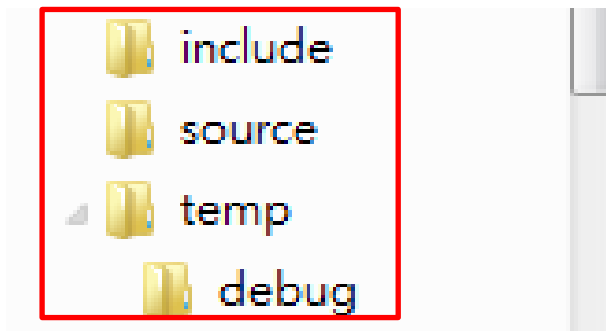
### 三、實驗題目：

- 利用 fscanf 方法，將文字檔(credit.txt)讀入記憶體，在利用 fwrite 方法轉存記憶體的內容成二進位檔案
- 最後，以 fread 方法再次讀取二進位檔案至記憶體，並顯示至螢幕。

### 四、設計步驟：

#### 1. 建立新的空專案

Step1-在 C:\c\_code 資料夾內新增名為 “Ch7\_Lab2” 的資料夾，再於 Ch7\_Lab2 資料夾內分別建立 include、source、temp 等資料夾，建立後需要在 temp 資料夾內新增名為 “debug”的資料夾，建立完成後如下圖



Step2-參照 Ch1\_Lab3 中 “1.建立新的空專案” Step2~Step4，設定相關路徑位置為 C:\c\_code\ Ch7\_Lab2

#### 2. 路徑設定、新增 .c 檔 .h 檔

Step1-參照 Ch1\_Lab3 中 “2. 路徑設定、新增 .c 檔” Step1~Step8，新增 Main.c 檔與設定相關屬性設定。

### 3. 撰寫 C 語言程式

The screenshot shows the Microsoft Visual Studio interface for a project named 'Lab2'. The 'Solution Explorer' on the left shows the project structure with 'main.c' selected under the 'source' folder. The 'Code' window displays the contents of 'main.c'. Three yellow callout boxes with red dashed lines point to specific parts of the interface:

- Step1-點擊兩下開啟 main.c**: Points to the 'main.c' file in the Solution Explorer.
- Step2-於 main.c 頁面下撰寫程式**: Points to the 'main.c' tab in the Code window.
- Step3-在此處撰寫 C 語言程式**: Points to the C code in the Code window.

```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  struct clientData
5  {
6      int acctNum;
7      char lastName[15];
8      char firstName[10];
9      double balance;
10 };
11
12 int main(void)
13 {
14     FILE *pRead, *pWrite;
15     FILE *cfPtr;
16     int cnt = 0;
17     struct clientData client = { 0, "", "", 0.0 };
18     // ... (code continues) ...
19     return 0;
20 }
```

main.c 程式碼：

使用 fscanf()讀取文字檔案"C://c\_code/Ch7\_Lab2/credit.txt" (ASCII 編碼)

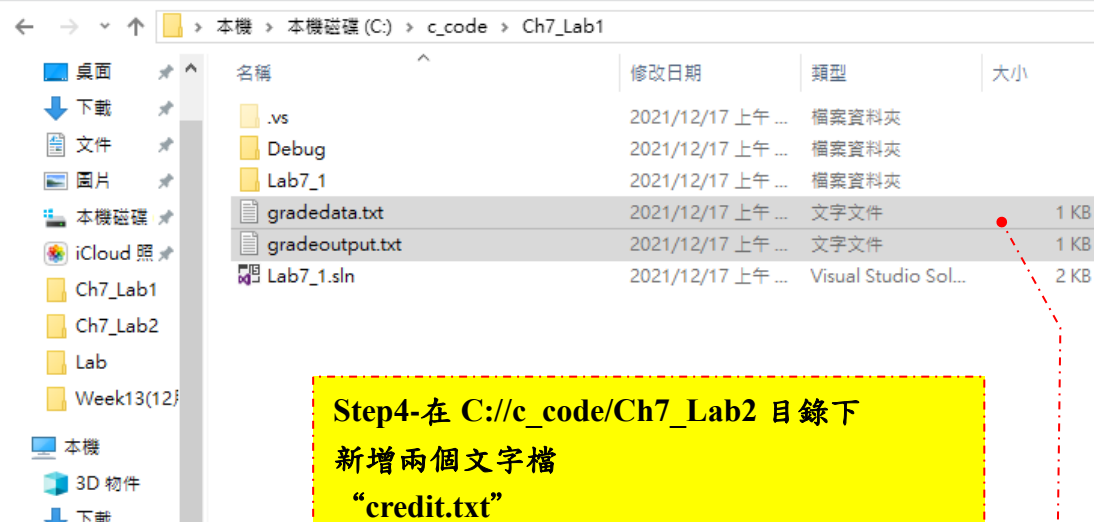
使用 fwrite()轉存成二進位檔案"C://c\_code/Ch7\_Lab2/credit\_bin.txt"

```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  struct clientData
5  {
6      int acctNum;
7      char lastName[15];
8      char firstName[10];
9      double balance;
10 };
11
12 int main(void)
13 {
14     FILE *pRead, *pWrite;
15     FILE *cfPtr;
16     int cnt = 0;
17     struct clientData client = { 0, "", "", 0.0 };
18
19     pRead = fopen("C://c_code/Ch7_Lab2/credit.txt", "r");
20     if (NULL == pRead)
21     {
22         return 0;
23     }
24
25     pWrite = fopen("C://c_code/Ch7_Lab2/credit_bin.txt", "wb");
26     if (NULL == pWrite)
27     {
28         fclose(pRead);
29         return 0;
30     }
31
32     while (!feof(pRead))
33     {
34         fscanf(pRead, "%d %s %s %lf", &client.acctNum, client.lastName,
35             client.firstName, &client.balance);
36
37         fwrite(&client, sizeof(struct clientData), 1, pWrite);
38
39         printf("%-6d%-16s%-11s%10.2f\n", client.acctNum,
40             client.lastName, client.firstName, client.balance);
41     }
42
43     fclose(pRead);
```

不同開檔方式

使用 fread()讀取二進位檔案"C://c\_code/Ch7\_Lab2/credit\_bin.txt"，並顯示出來至螢幕

```
43 fclose(pRead);
44 fclose(pWrite);
45 printf("已讀取文字檔的ASCII資料，並轉存成二進位資料檔案\n\n");
46 system("pause");
47
48 printf("\n讀取二進位資料檔案，並格式化輸出如下:\n");
49 if ((cfPtr = fopen("C://c_code/Ch7_Lab2/credit_bin.txt", "rb")) == NULL)
50 {
51     printf("File could not be opened.\n");
52 }
53 else
54 {
55     printf("%-6s%-16s%-11s%10s\n", "Acct", "Last Name", "First Name", "Balance");
56
57     cnt = fread(&client, sizeof(struct clientData), 1, cfPtr);
58
59     while (cnt > 0)
60     {
61         printf("%-6d%-16s%-11s%10.2f\n", client.acctNum,
62             client.lastName, client.firstName, client.balance);
63
64         cnt = fread(&client, sizeof(struct clientData), 1, cfPtr);
65     }
66
67     fclose(cfPtr);
68 }
69
70 system("pause");
71 return 0;
72 }
```



**Step4-在 C://c\_code/Ch7\_Lab2 目錄下**

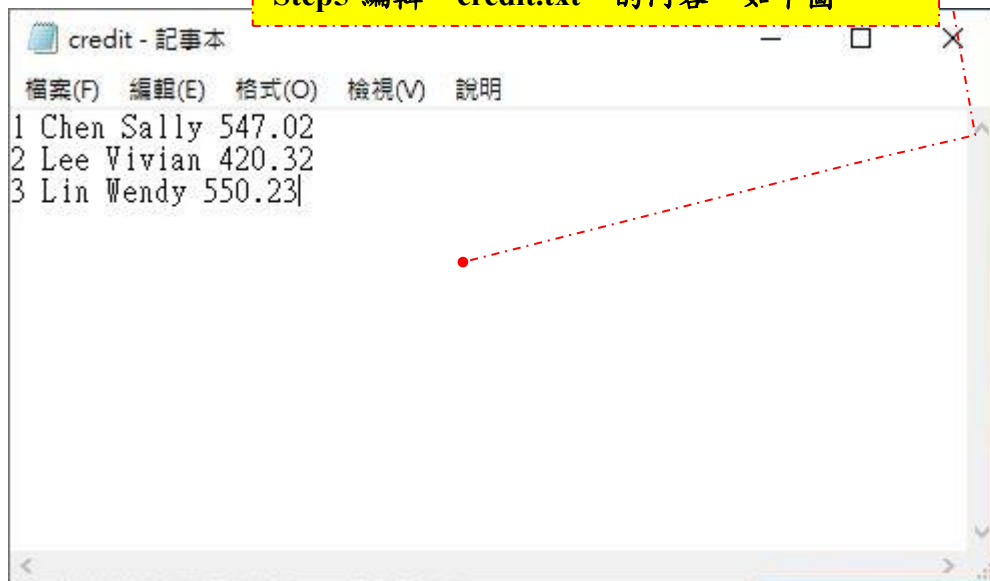
**新增兩個文字檔**

**"credit.txt"**

**"credit\_bin.txt"**

**\*務必確認程式碼中的讀/寫檔案路徑與 txt 文字檔路徑相同，否則會造成讀/寫檔案失敗**

**Step5-編輯“credit.txt”的內容，如下圖**

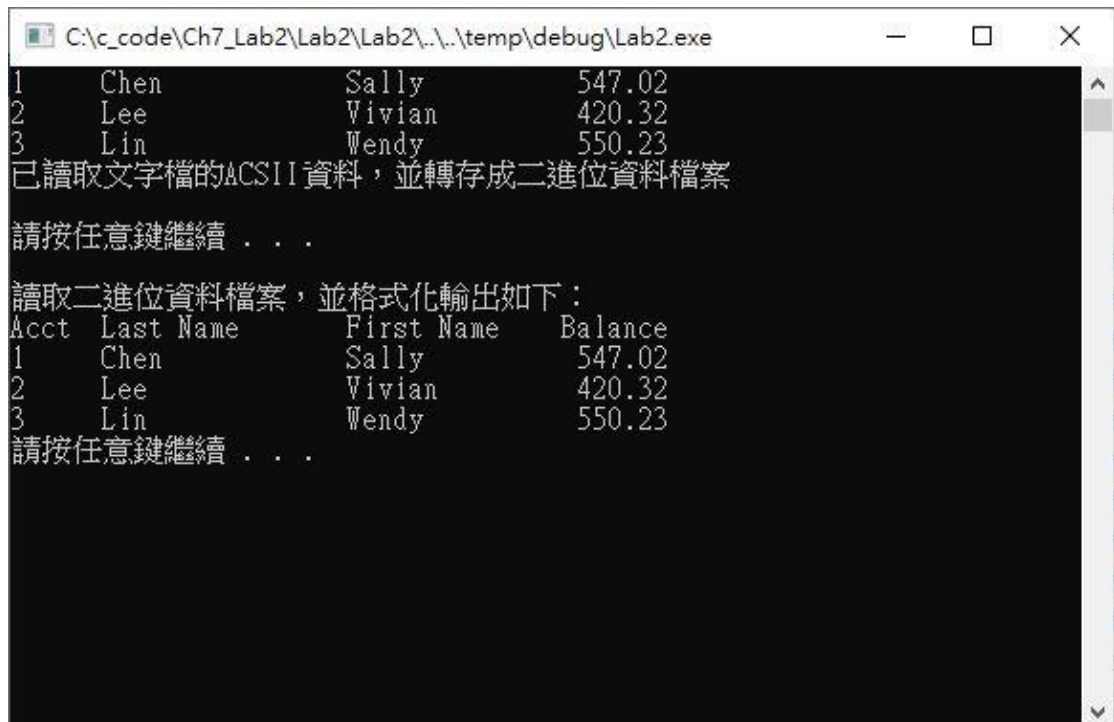


註：credit\_bin.txt 不需要編輯，保持空白(將由程式執行後，做檔案寫入動作)

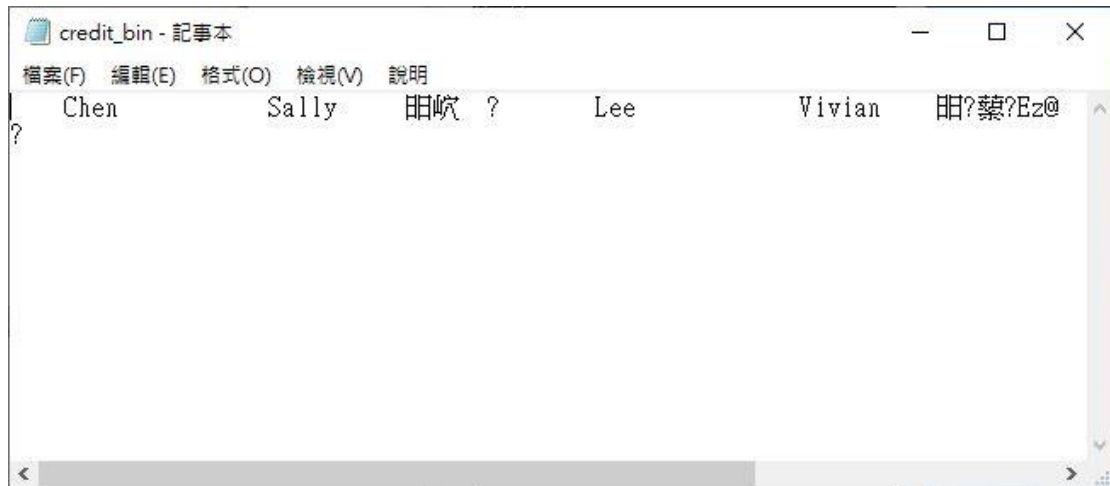
**Step6-點選開始偵測**



**4. 執行與測試程式結果**



執行完後，檢查 credit\_bin.txt 是否有資料寫入



!!! 注意：

1. 若碰到特定警告造成無法編譯，則在組態屬性，C/C++的進階中加入停用特定警告 4996。
2. 若檔案無法讀取，可更換置 d 槽，或再新增一個 TXT 文字檔。
3. 若檔案無法讀取，請檢查是否檔名拼字錯誤。
4. 程式 34 行讀取長浮點數為 longfloat 取 lf，而非數字 1。

