

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. 建立專案

方法 A. 透過 Github Classroom 下載並開啟專案

注意：透過方法 A 建立專案後，直接跳至步驟 **3. 撰寫 C 語言程式**

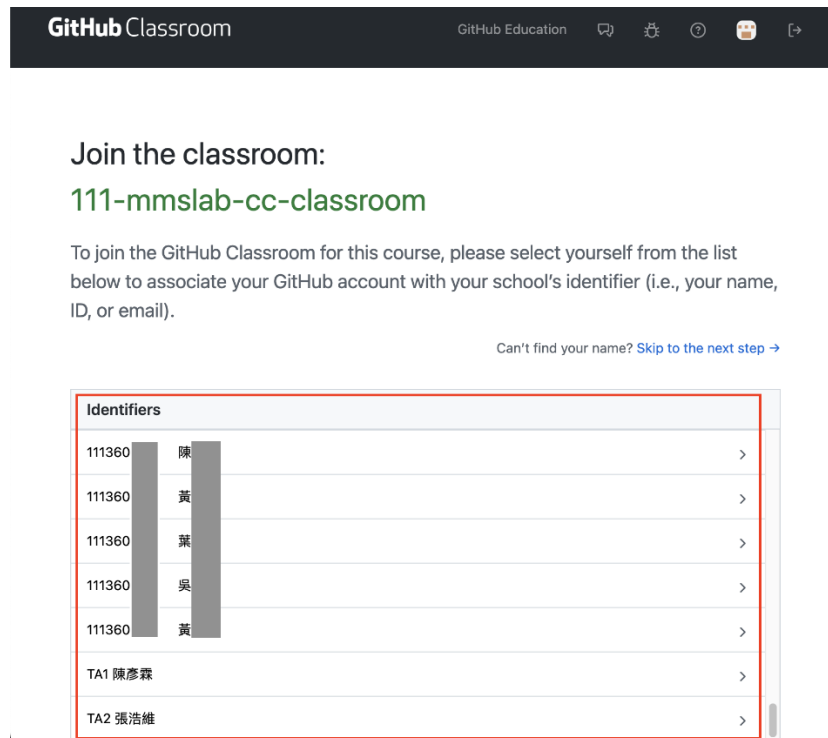
Step1. 點擊 Github Classroom 連結

- Lab：<https://tools-api.italkutalk.com/cc/ch7-lab>

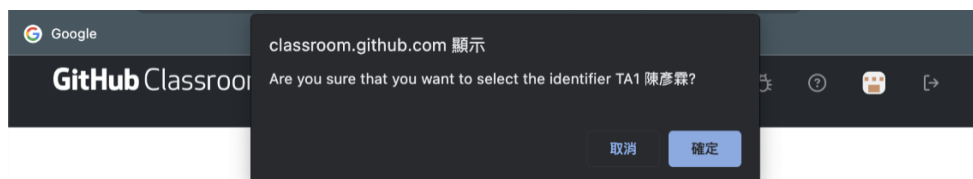
Step2. 將自己的 Github 帳號與 Classroom 學生連結

同一門課程 Github Classroom 的作業或實驗僅需連結一次（若曾經連結過，可以略過此步驟）。

在學生清單中，會列出本門課程尚未被連結的學生，請找到並點擊自己的學號/姓名



點擊後會跳出確認提示，確認無誤點擊確定。



如果遇到以下問題：

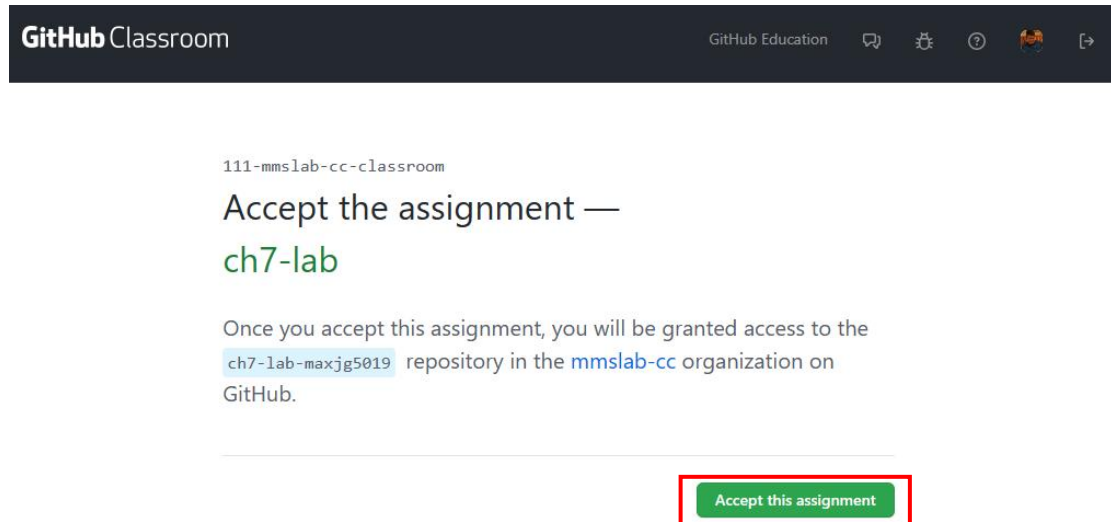
1. 名單中找不到自己的學號姓名

2. 選擇錯人

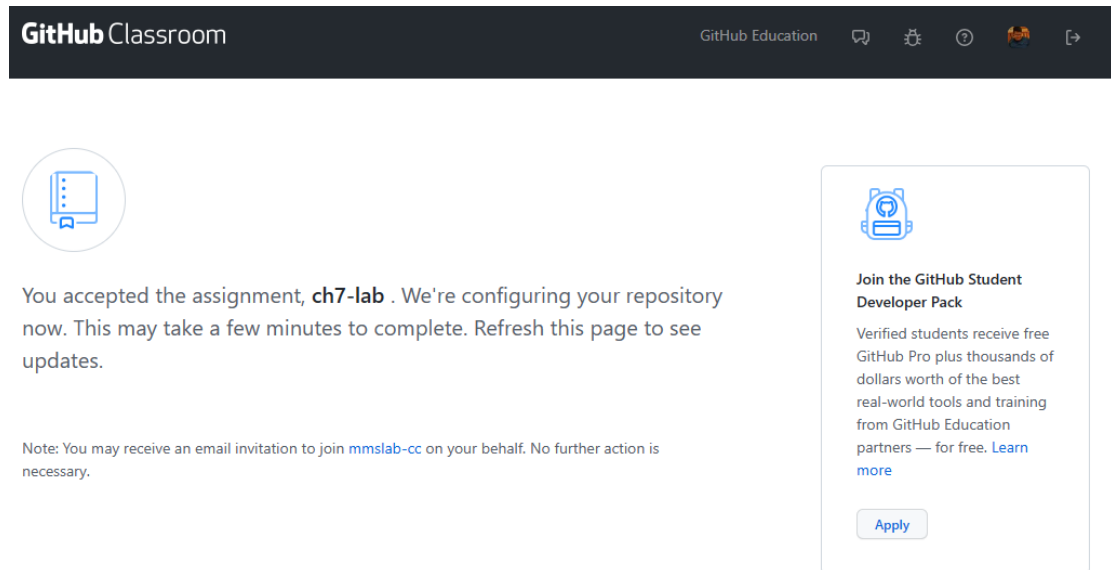
3. 學號姓名錯誤

請與課堂助教反應，助教將會協助處理。

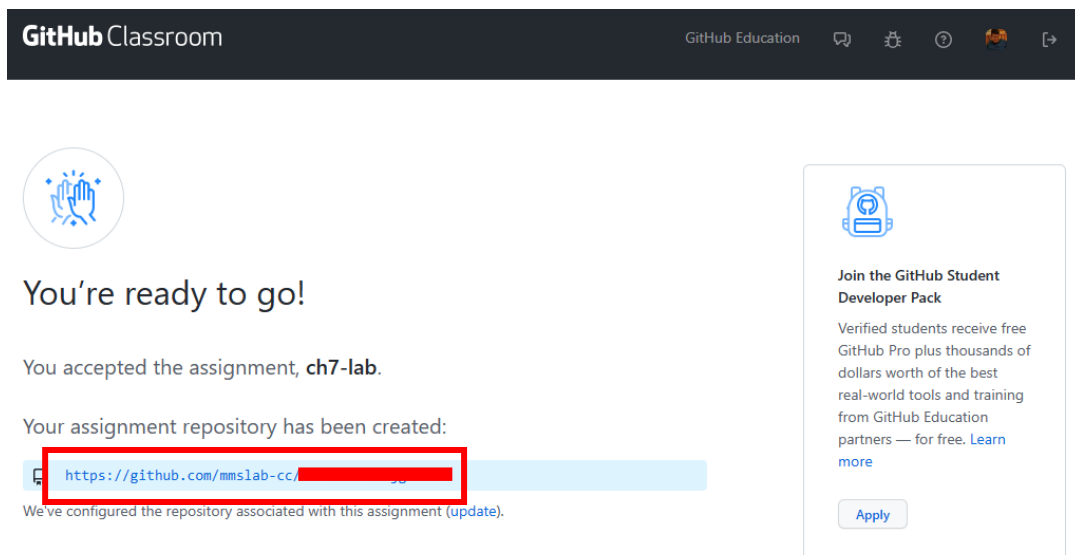
Step3. 接受 Assignment，點擊 Accept this assignment



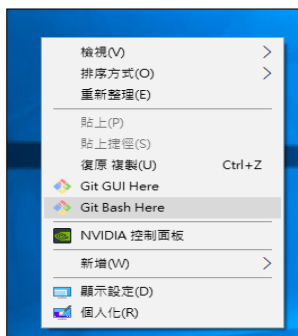
接受 Assignment 後，Github classroom 會幫你建立專屬的 repository，而建立專屬的 repository 需要一段時間，請等 10 秒左右刷新此頁面



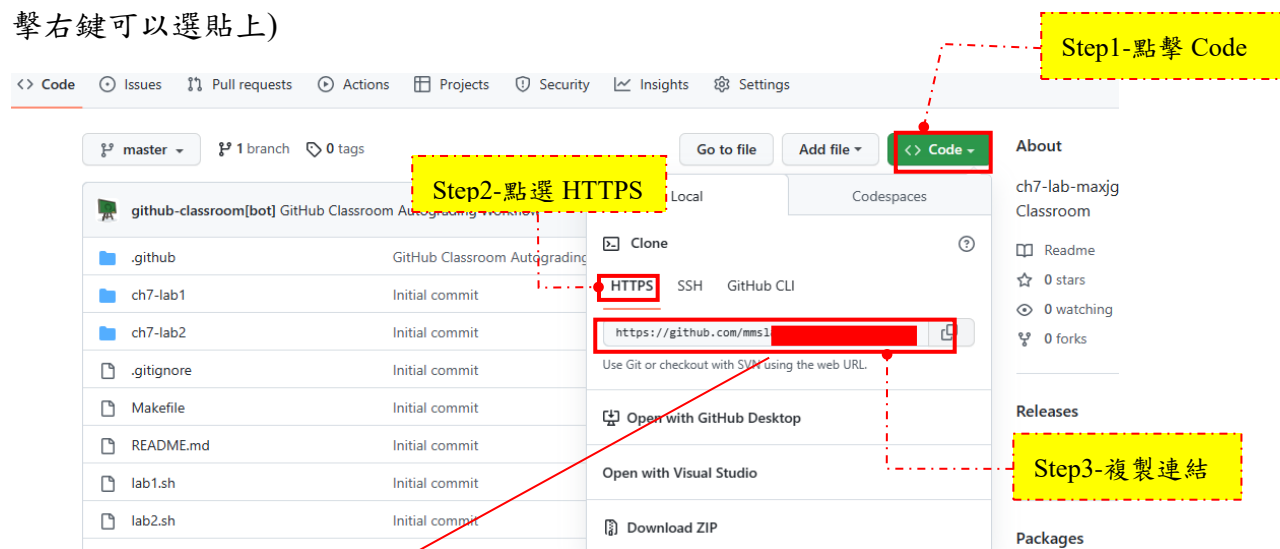
刷新頁面後，將會看到屬於自己的 repository 連結，並點擊該連結。



Step4. 將專案 Clone 到自己電腦
到桌面開啟 Git Bash



複製專案在遠端資料庫的位置，並在 Git bash 輸入指令進行下載專案(命令列點擊右鍵可以選貼上)



\$ git clone <https://github.com/xxx/xxx.git> (記得改成自己的資料庫連結網址)

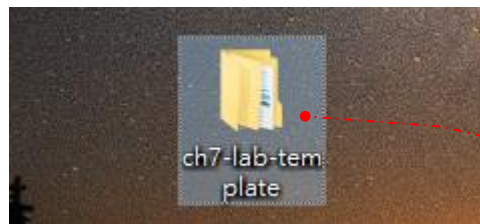
輸入指令情況：

```
Jdway@DESKTOP-GBM49C1 MINGW64 ~/Desktop
$ git clone https://github.com/your-github-id/lab2.git
Cloning into 'lab2'...
remote: Counting objects: 7, done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 7 (delta 0), reused 7 (delta 0), pack-reused 0
Unpacking objects: 100% (7/7), done.

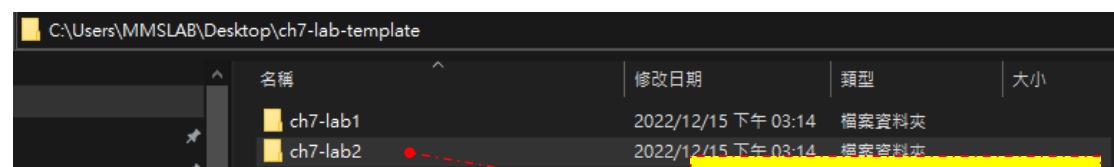
Jdway@DESKTOP-GBM49C1 MINGW64 ~/Desktop
$ |
```

Step5. 完成下載專案後請到桌面開啟剛下載的專案資料夾，開啟專案檔案。

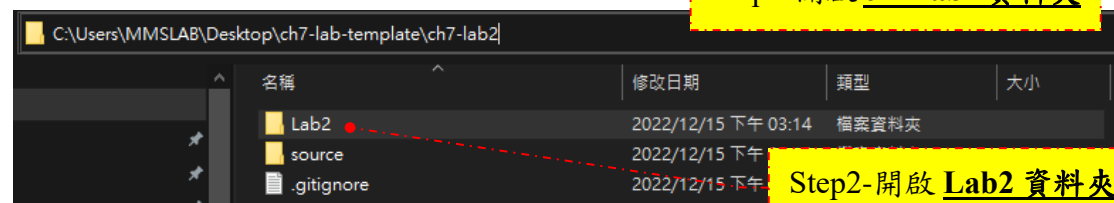
檔案路徑為：Desktop\ch7-lab-{你的 Github 帳號 ID}\ch7-lab2\Lab2\Lab2.sln



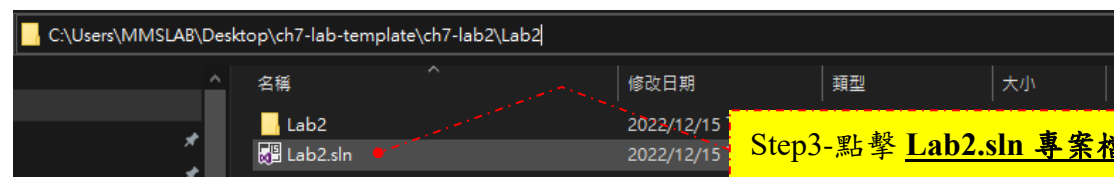
於**桌面**開啟專案資料夾資料夾
資料夾名稱會是
ch1-lab-{你的 Github 帳號 ID}



Step1-開啟 **ch7-lab2** 資料夾



Step2-開啟 **Lab2** 資料夾

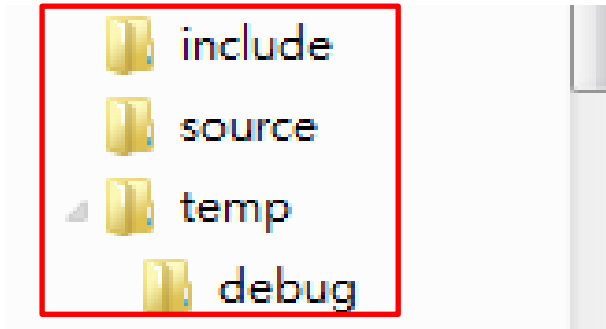


Step3-點擊 **Lab2.sln** 專案檔案

注意：透過方法 A 建立專案後，直接跳至步驟 3.撰寫 C 語言程式

方法 B. 透過 Visual Studio 新建專案

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 with the following components:

- Top Menu Bar:** 檔案(F), 編輯(E), 檢視(V), 專案(P), 建置(B), 偵錯(D), 小組(M), 工具(T), 測試(S), 分析(N), 視窗(W), 說明(H).
- Toolbar:** Includes icons for file operations, a dropdown for 'Debug', a dropdown for 'x86', and a dropdown for '本機 Windows 偵錯工具'.
- Left Panel (Solution Explorer):** Shows a project named 'Lab2' with a 'source' folder containing 'main.c'. A red dashed line points from 'main.c' to the code editor.
- Code Editor:** Displays the content of 'main.c'. The code includes headers, a struct definition, and a main function. A red dashed line points from the 'main.c' tab to the code editor.
- Right Panel:** Contains the 'Output' window.
- Bottom Status Bar:** Shows '100 %', '第 73 行', '第 1 欄', '字元 1', 'INS', and '新增至原始檔控制'.

Three yellow callout boxes with red dashed lines provide instructions:

- 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 Editor.
- Step3-在此處撰寫 C 語言程式**: Points to the code area within the 'main' function in the Code Editor.

```
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     pWrite = fopen("C://credit_bin.txt", "wb");
20     if (NULL == pWrite)
21     {
22         return 0;
23     }
24
25     pWrite = fopen("C://credit_bin.txt", "wb");
26     if (NULL == pWrite)
27     {
```

main.c 程式碼：

使用 fscanf() **讀取** 文字檔案 "Desktop/ ch7-lab-{你的 Github 帳號 ID}/
ch7-lab2/Lab2/Lab2/ **credit.txt** " (ASCII 編碼)

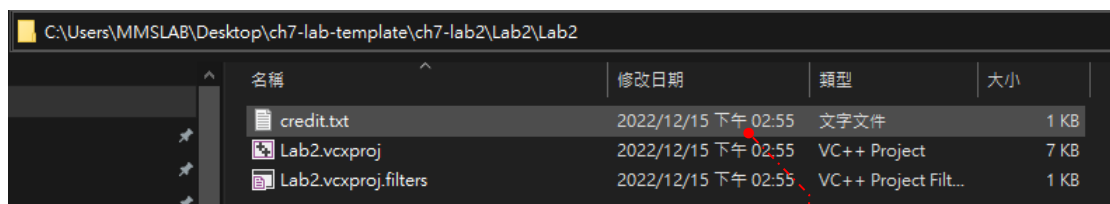
使用 fwrite() **轉存** 成二進位檔案 "Desktop/ ch7-lab-{你的 Github 帳號 ID}/
ch7-lab2/Lab2/Lab2/ **credit_bin.txt**"

```
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("./credit.txt","r");
20     if(NULL == pRead)
21     {
22         return 0;
23     }
24
25     pWrite=fopen("./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);
44 }
```

不同開檔方式

使用 fread()讀取二進位檔案" Desktop/ ch7-lab-{你的 Github 帳號 ID}/
ch7-lab2/Lab2/Lab2/ credit_bin.txt"，並顯示出來至螢幕

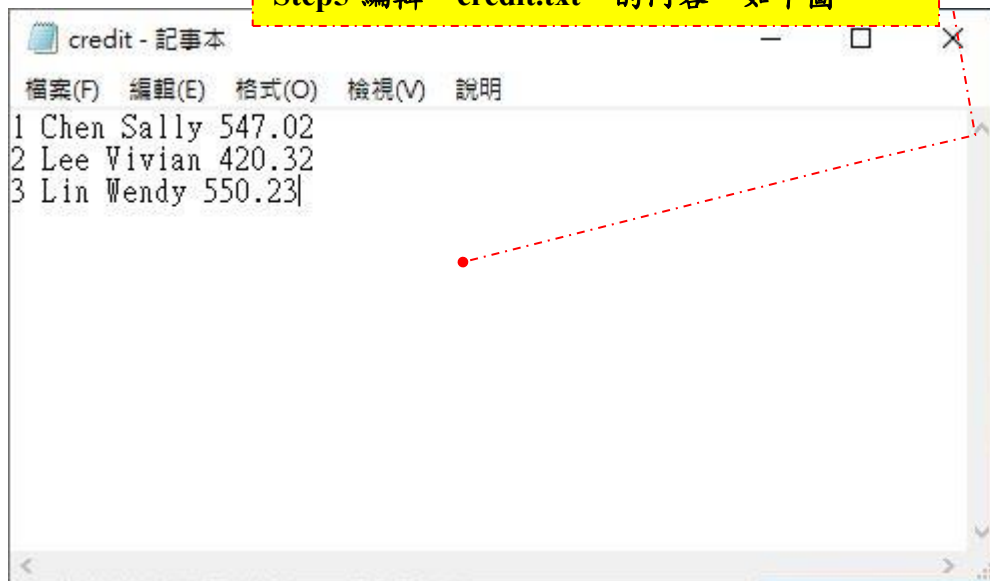
```
43     fclose(pRead);
44     fclose(pWrite);
45     printf("已讀取文字檔的ASCII資料，並轉存成二進位資料檔案\n\n");
46     system("pause");
47
48     printf("\n讀取二進位資料檔案，並格式化輸出如下:\n");
49     if ((cfPtr=fopen("./credit_bin.txt", "rb")) == NULL)
50     {
51         printf("File could not be opened.\n");
52     }
53     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 }
```



名稱	修改日期	類型	大小
credit.txt	2022/12/15 下午 02:55	文字文件	1 KB
Lab2.vcxproj	2022/12/15 下午 02:55	VC++ Project	7 KB
Lab2.vcxproj.filters	2022/12/15 下午 02:55	VC++ Project Filt...	1 KB

**Step4-在專案目錄 Desktop\ ch7-lab-{你的
Github 帳號 ID}\ ch7-lab2\Lab2\Lab2\
找到文字檔 “credit.txt” 並開啟該檔案**

Step5-編輯“credit.txt”的內容，如下圖



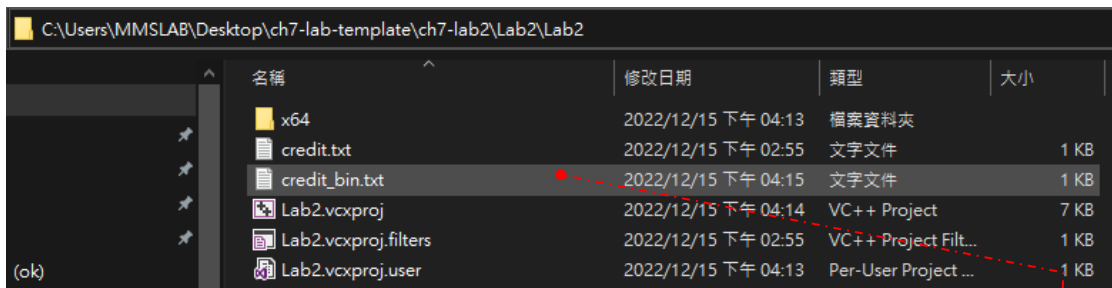
Step6-點選開始偵測



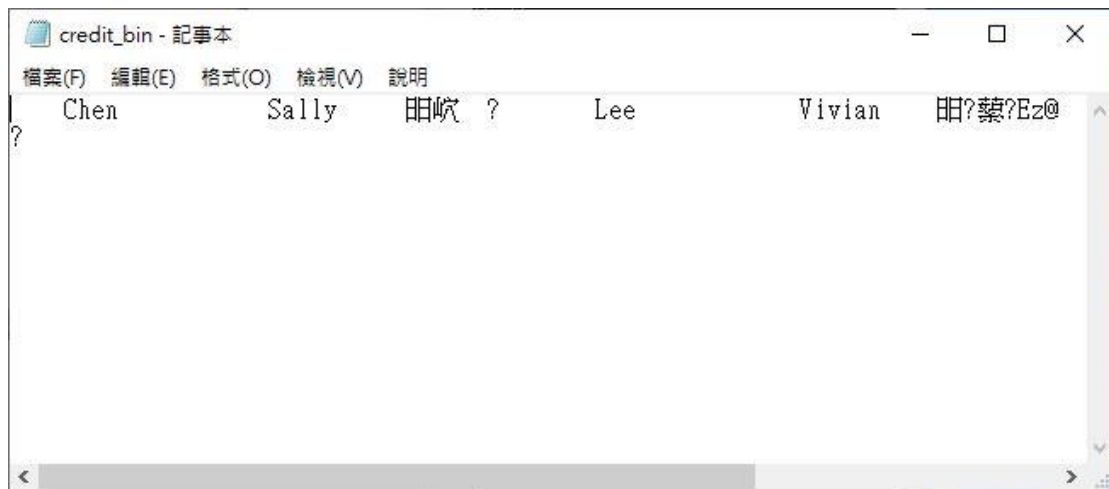
4. 執行與測試程式結果



註：執行到這邊需按任意鍵，才会有後續結果輸出



**Step7-在專案目錄 Desktop\ ch7-lab-{你的 Github 帳號 ID}
\ ch7-lab2\Lab2\Lab2\
確認文字檔 “credit_bin.txt” 是否有成功寫入產生並開啟該檔案**

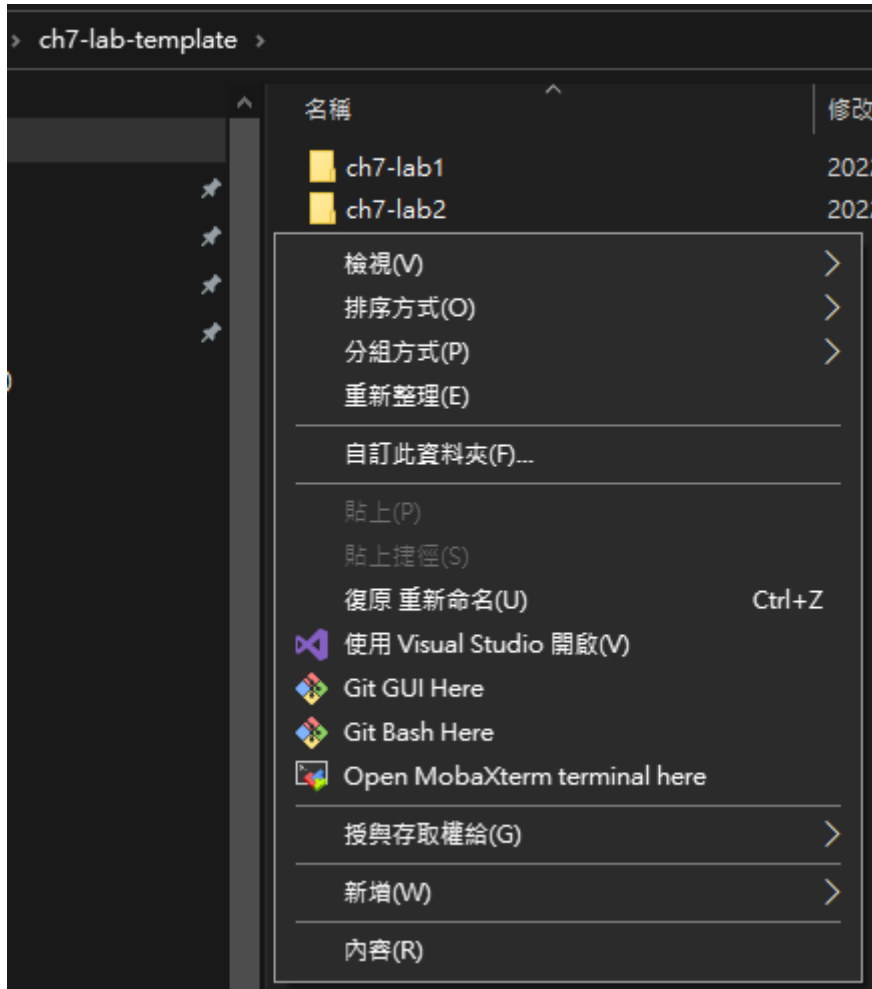


- 請將結果呈現給助教檢查

● 檢查通過後上傳實驗至 Github Classroom

請參考從下方的上傳專案說明，將專案透過 Git 指令 push 到 Github classroom

1. 專案編輯完成後，於本周課程專案的資料夾點選滑鼠右鍵，選擇 **Git Bash Here** 開啟 Git 終端機。



2. 於終端機中依序輸入以下指令，將程式上傳至遠端資料庫。

```
git add .  
  
git commit -m "[修改的內容描述]"  
  
git push origin master
```

- 本周課程 Lab 到此結束，謝謝各位同學參閱

!!! 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 3 4 行讀取長浮點數為 longfloat 取 lf，而非數字 1。

