

實驗項目 - 檔案讀寫 fscanf、fprintf 使用

一、本節目的：

- 學習使用 fscanf、fprintf 方法
- 學習如何讀寫文字檔

二、設計重點：

- 利用 fscanf、fprintf 方法來完成讀寫文字檔程式

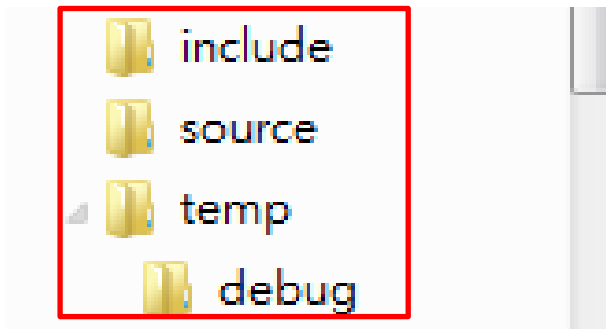
三、實驗題目：

- 利用 fscanf、fprintf 方法完成一程式，可讀取成績檔(gradedate.txt)，並計算平均總和，將每位學生各科成績、總和、平均寫入輸出檔(gradeoutput.txt)中。並利用結構 typedef struct _student 用來記錄學生姓名(name)、國文成績(chinese)、英文成績(english)、數學成績(math)、成績總和(sum)、成績平均(avg)。

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_Lab1” 的資料夾，再於 Ch7_Lab1 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

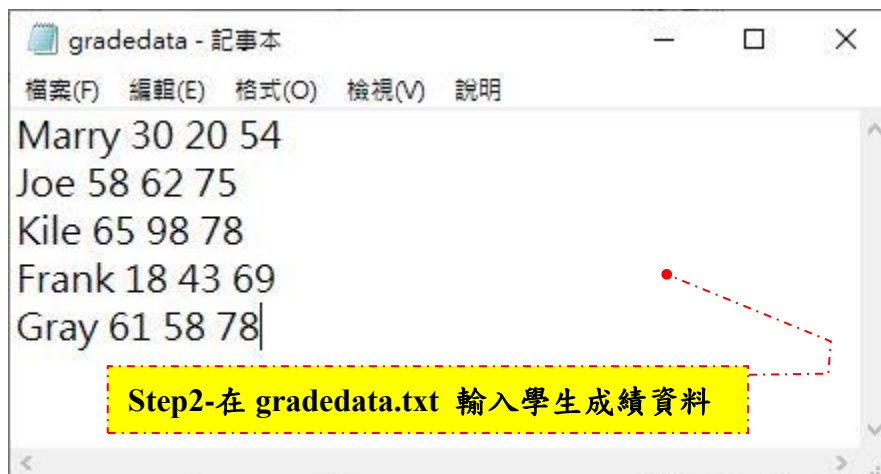
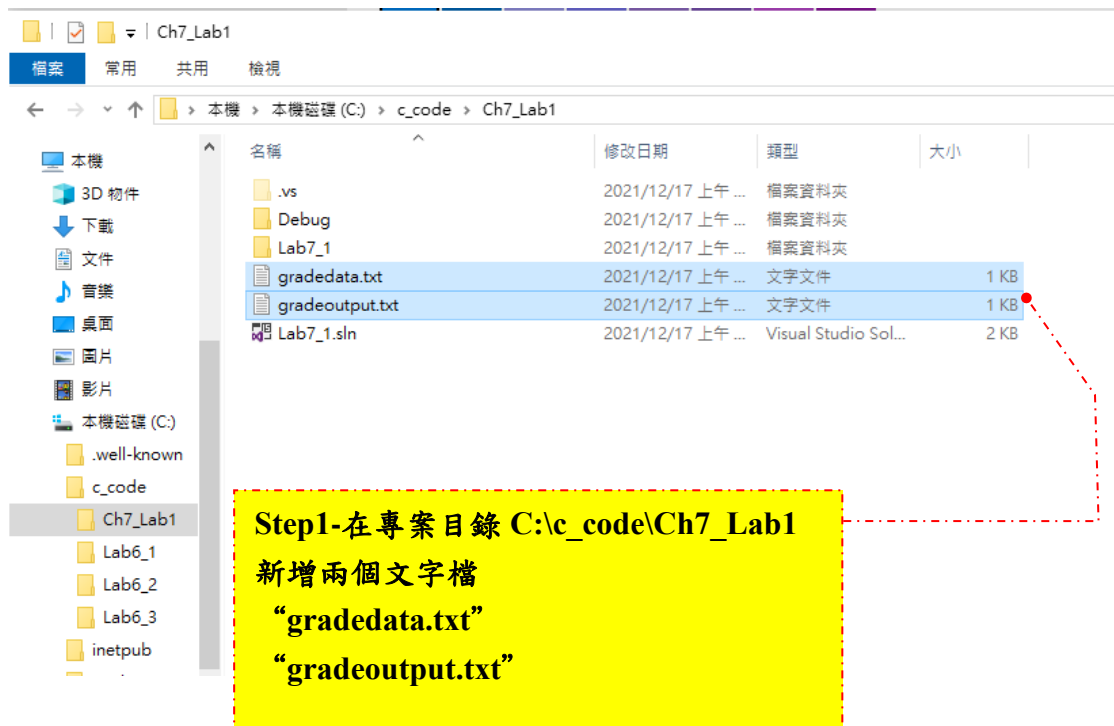


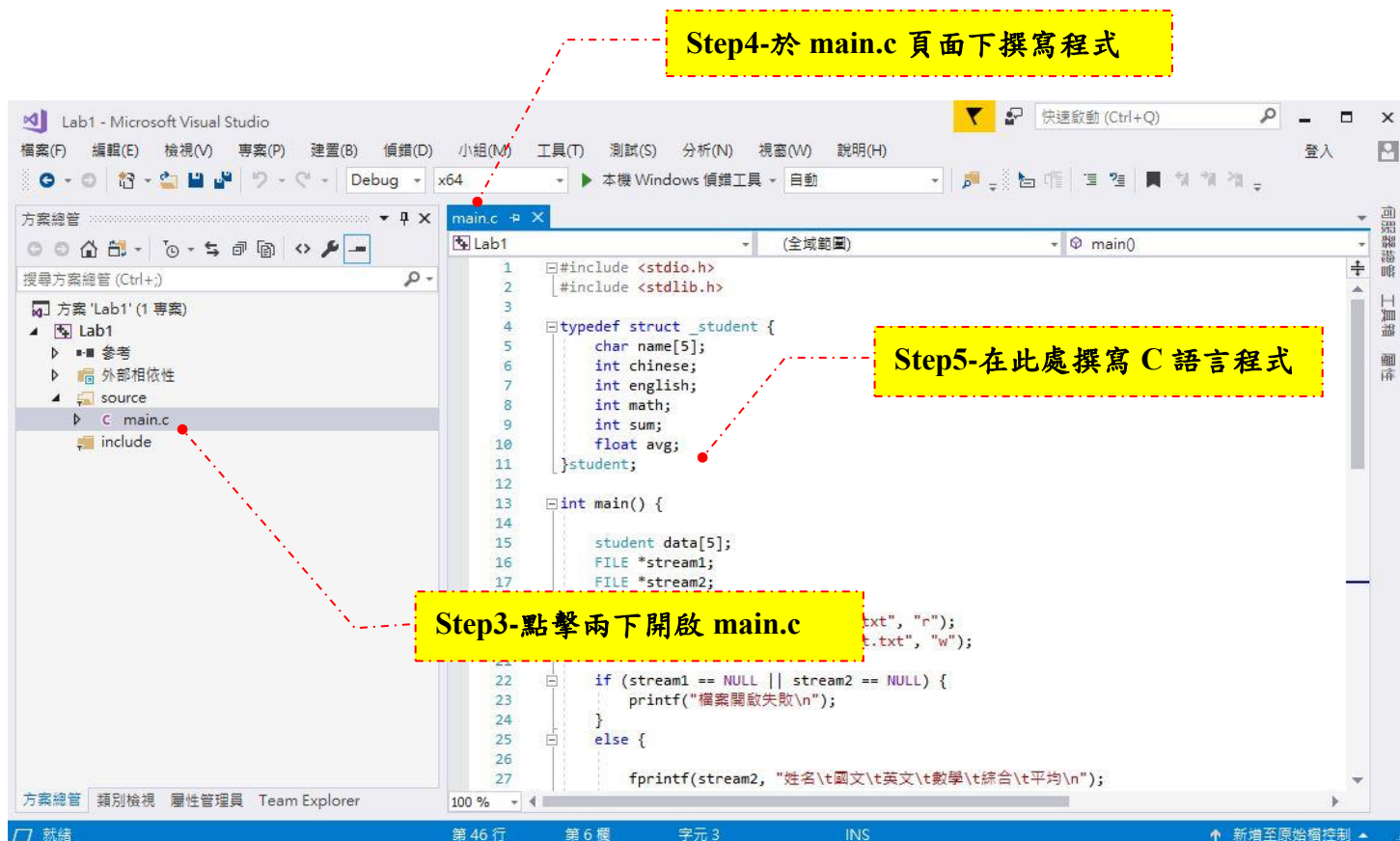
Step2-參照 Ch1_Lab3 中 “1.建立新的空專案” Step2~Step4，設定相關路徑位置為 C:\c_code\ Ch7_Lab1

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





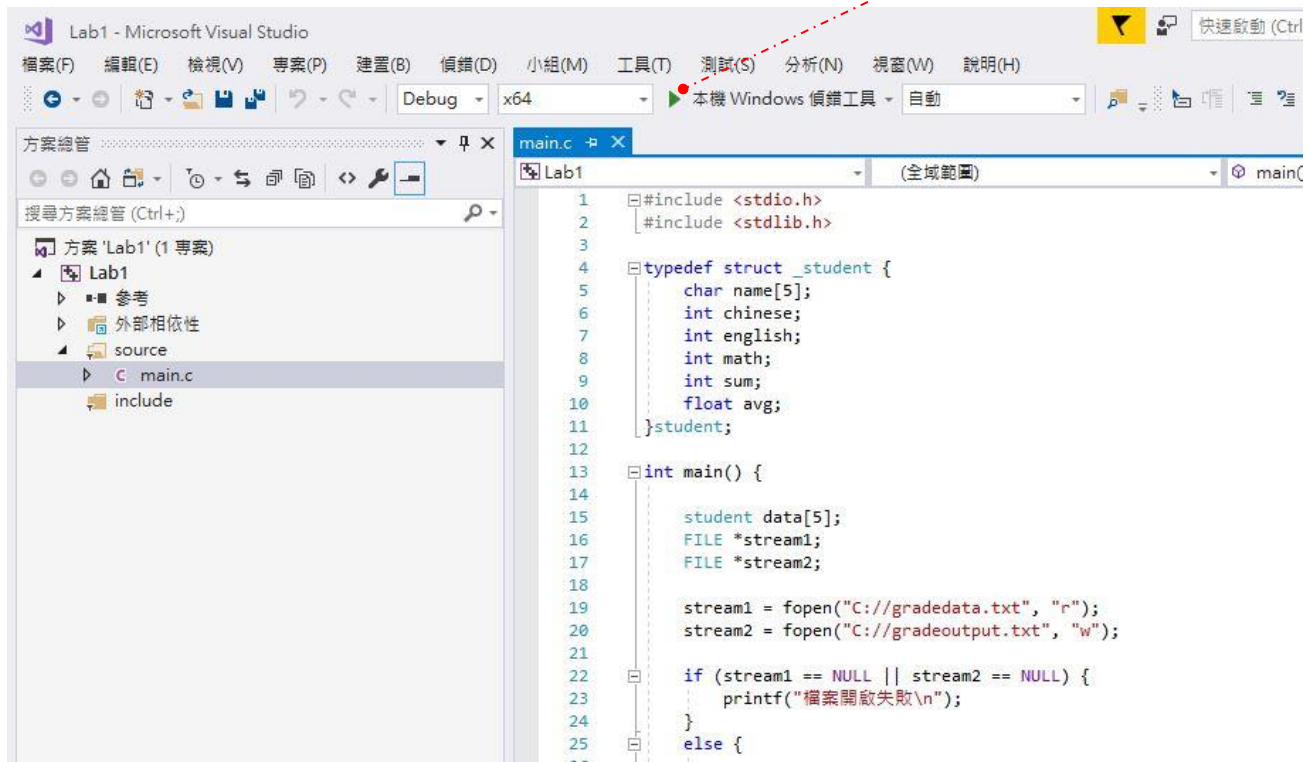
main.c 程式碼：

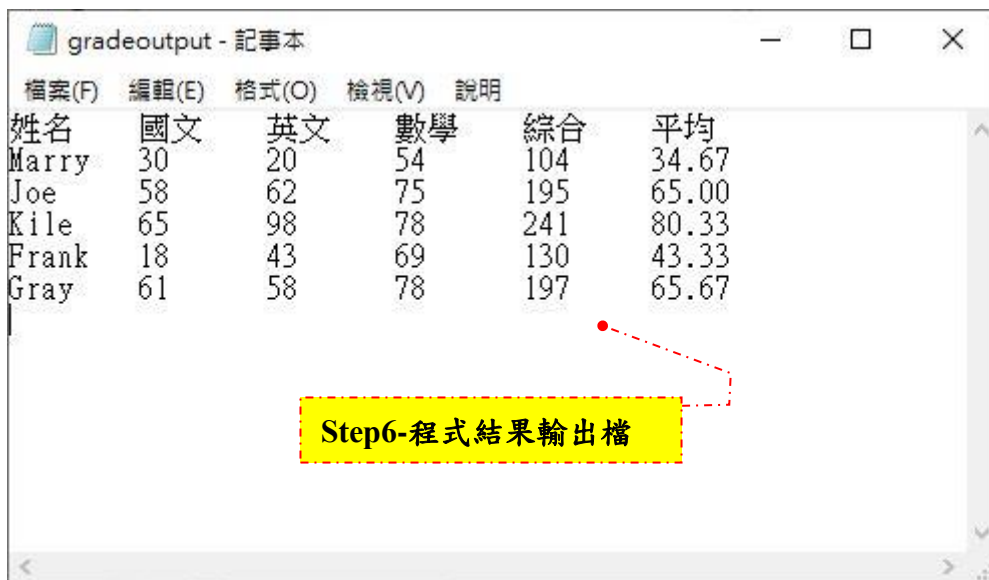
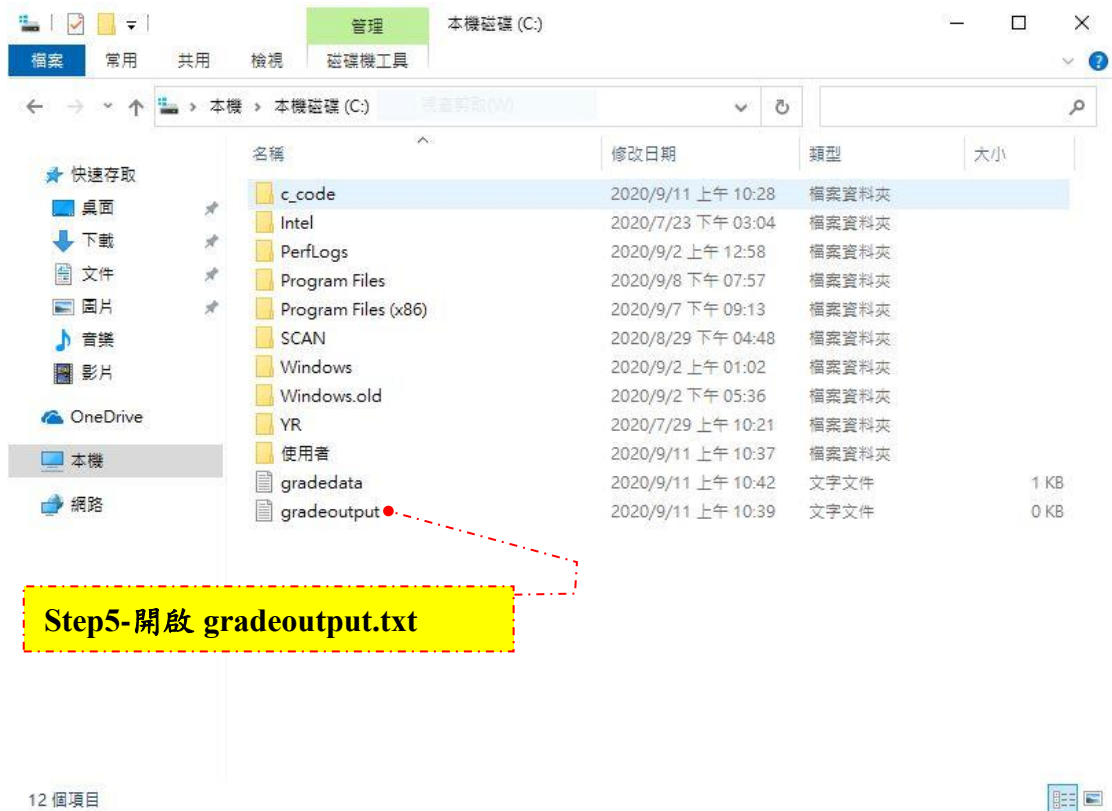
```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 typedef struct _student
5 {
6     char name[5];    //學生姓名
7     int chinese;     //國文成績
8     int english;     //英文成績
9     int math;        //數學成績
10    int sum;          //成績總和
11    float avg;        //成績平均
12 } student;
13
14 int main()
15 {
16     student data[5];
17     FILE *stream1;
18     FILE *stream2;
19
20     stream1 = fopen("C://c_code/Ch7_Lab1/gradedata.txt", "r");
21     stream2 = fopen("C://c_code/Ch7_Lab1/gradeoutput.txt", "w");
22
23     if (stream1 == NULL || stream2 == NULL) {
24         printf("檔案開啟失敗\n");
25     }
26     else {
27         fprintf(stream2, "姓名\t國文\t英文\t數學\t總和\t平均\n");
28         for (int i = 0; i < 5; i++) {
29             fscanf(stream1, "%s", data[i].name);
30             fscanf(stream1, "%d", &data[i].chinese);
31             fscanf(stream1, "%d", &data[i].english);
32             fscanf(stream1, "%d", &data[i].math);
33
34             data[i].sum = data[i].chinese + data[i].english + data[i].math;
35             data[i].avg = data[i].sum / 3.0;
36             fprintf(stream2, "%s\t%d\t%d\t%d\t%d\t%.2f\n", data[i].name,
37                 data[i].chinese, data[i].english, data[i].math, data[i].sum, data[i].avg);
38         }
39         fclose(stream1);
40         fclose(stream2);
41     }
42
43     system("pause");
44     return 0;
45 }
```

*若專案路徑與 Step1 不同
務必將 fopen 的路徑改為你的專案路徑
否則無法正確讀取檔案

4. 執行與測試程式結果

Step3-點選開始偵測，進行偵測





!!! 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。

