

實驗項目 - C 語言的條件運算

一、本節目的：

- 學習開發 C 語言程式
- 實現在 Visual Studio 2017 系統設計平台上

二、設計重點：

- C 語言的條件式應用

三、設計步驟：

1. 建立專案

方法 A. 透過 Github Classroom 下載並開啟專案

Step1. 請參考實驗 Lab0-2 的章節 0.1.1 連接 Github Classroom 將實驗專案 clone 至本地。

Step2. 開啟專案檔案

The first screenshot shows a file explorer with a table of files. A red dot is on the 'ch1-lab-template' folder, and a yellow callout box says: 開啟專案資料夾
資料夾名稱會是 ch1-lab-{你的 Github 帳號 ID}

名稱	修改日期	類型	大小
ch1-lab-template	2022/9/28 下午 0...	檔案資料夾	

The second screenshot shows a file explorer with a table of files. A red dot is on the 'ch1-lab3' folder, and a yellow callout box says: 開啟 ch1-lab3 資料夾

名稱	修改日期	類型	大小
ch1-lab1	2022/9/28 下午 0...	檔案資料夾	
ch1-lab2	2022/9/28 下午 0...	檔案資料夾	
ch1-lab3	2022/9/28 下午 0...	檔案資料夾	
.gitignore	2022/9/28 下午 0...	文字文件	1 KB
lab1.sh	2022/9/28 下午 0...	Shell Script	
lab2.sh	2022/9/28 下午 0...	Shell Script	
lab3.sh	2022/9/28 下午 0...	Shell Script	3 KB
Makefile	2022/9/28 下午 0...	檔案	1 KB
README.md	2022/9/28 下午 0...	Markdown 來源...	2 KB

The third screenshot shows a file explorer with a table of files. A red dot is on the 'Lab3' folder, and a yellow callout box says: 開啟 Lab3 資料夾

名稱	修改日期	類型	大小
Lab3	2022/9/28 下午 0...	檔案資料夾	
source	2022/9/28 下午 0...	檔案資料夾	
.gitignore	2022/9/28 下午 0...	文字文件	5 KB

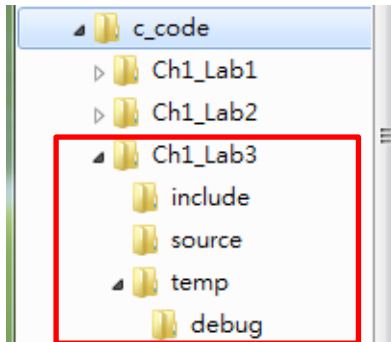
The fourth screenshot shows a file explorer with a table of files. A red dot is on the 'Lab3.sln' file, and a yellow callout box says: 點擊 lab3.sln 專案檔案

名稱	修改日期	類型	大小
Lab3	2022/9/28 下午 0...	檔案資料夾	
x64	2022/9/28 下午 0...	檔案資料夾	
Lab3.sln	2022/9/28 下午 0...	Visual Studio Sol...	2 KB

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

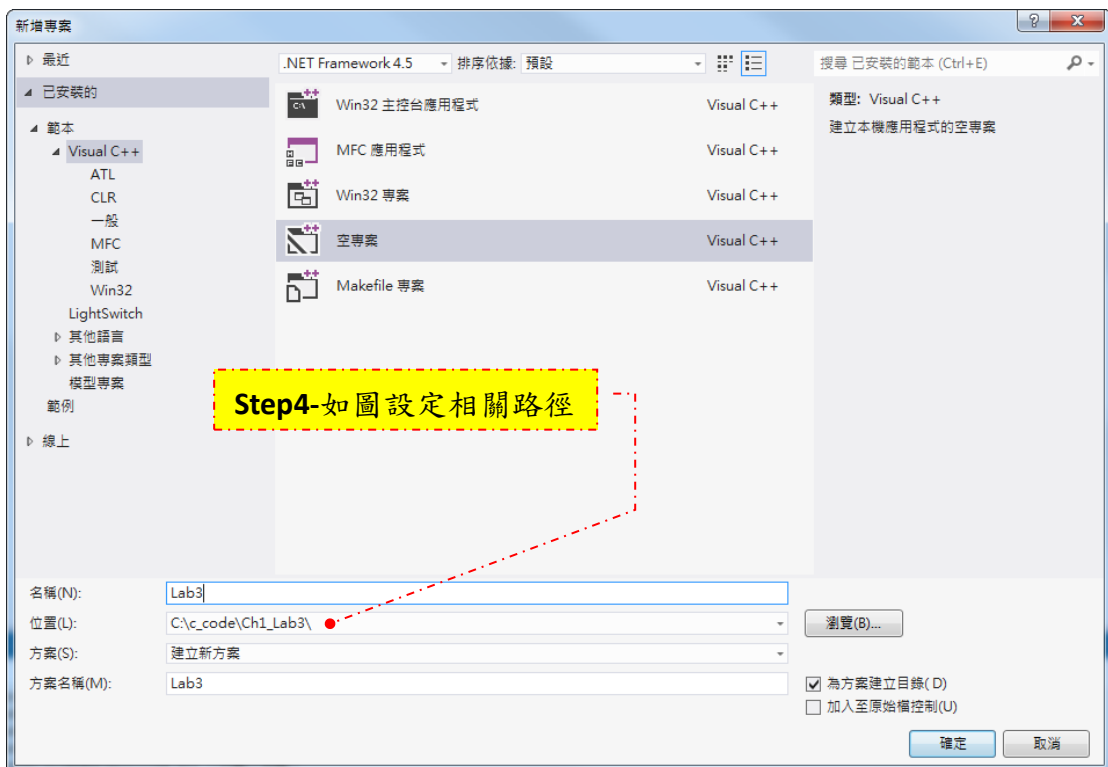
方法 B. 透過 Visual Studio 新建專案

Step1-在 C:\c_code 資料夾內新增名為“Ch1_Lab3”的資料夾，再於 Ch1_Lab3 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

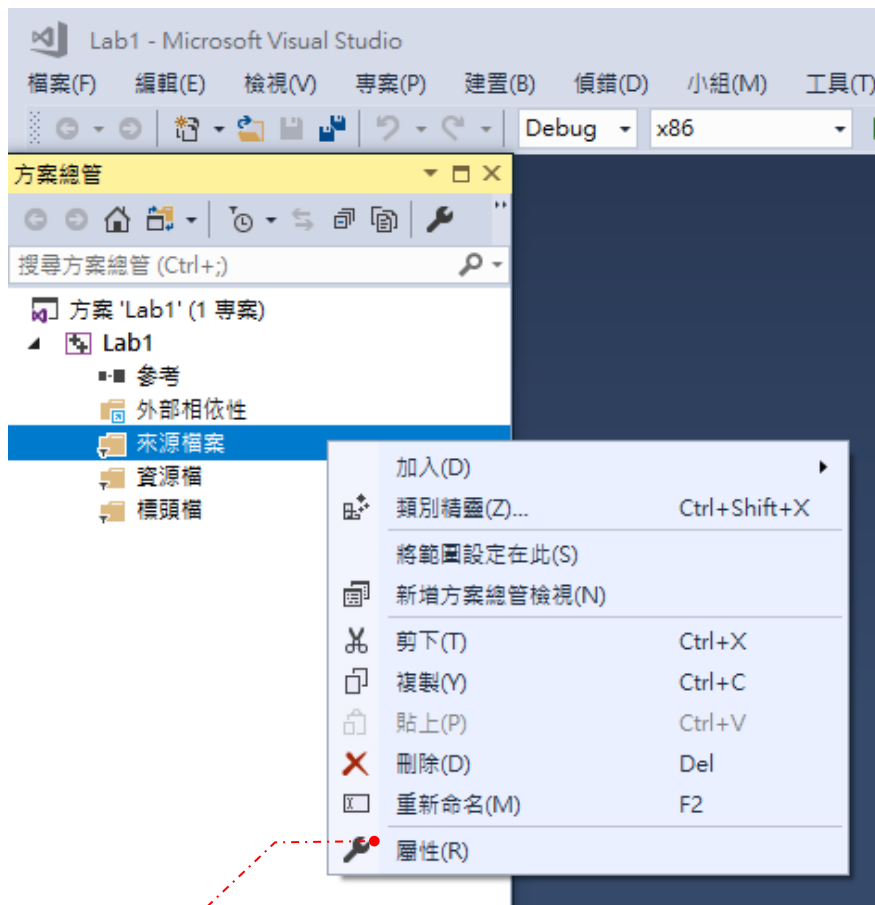


Step2-開啟 Microsoft Visual Studio 視窗畫面後點選左上角“開始(F)”

Step3-點選“新增(N)”再點選“專案(P)”

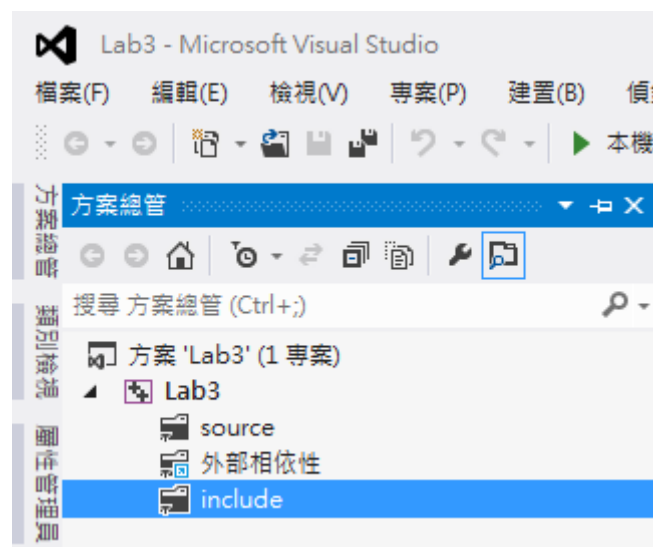


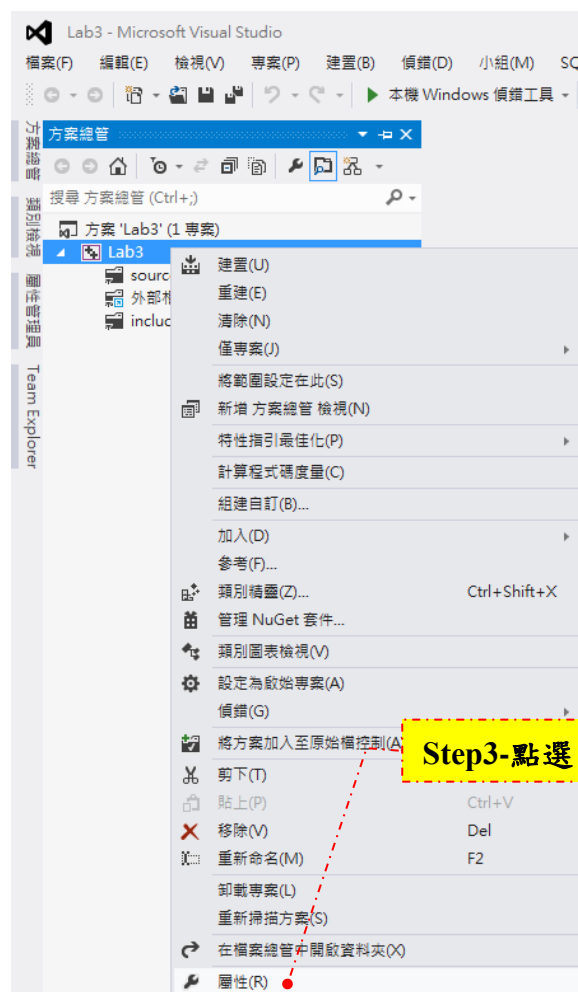
2. 路徑設定、新增 .c 檔

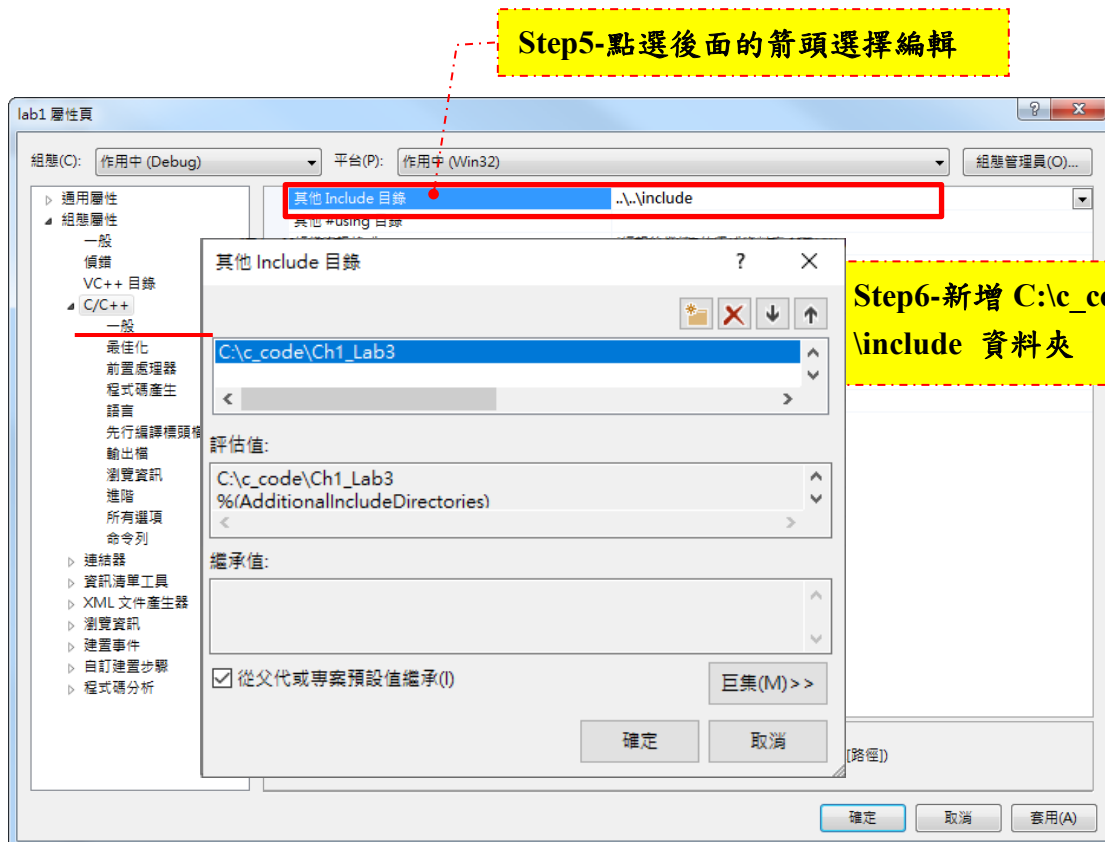
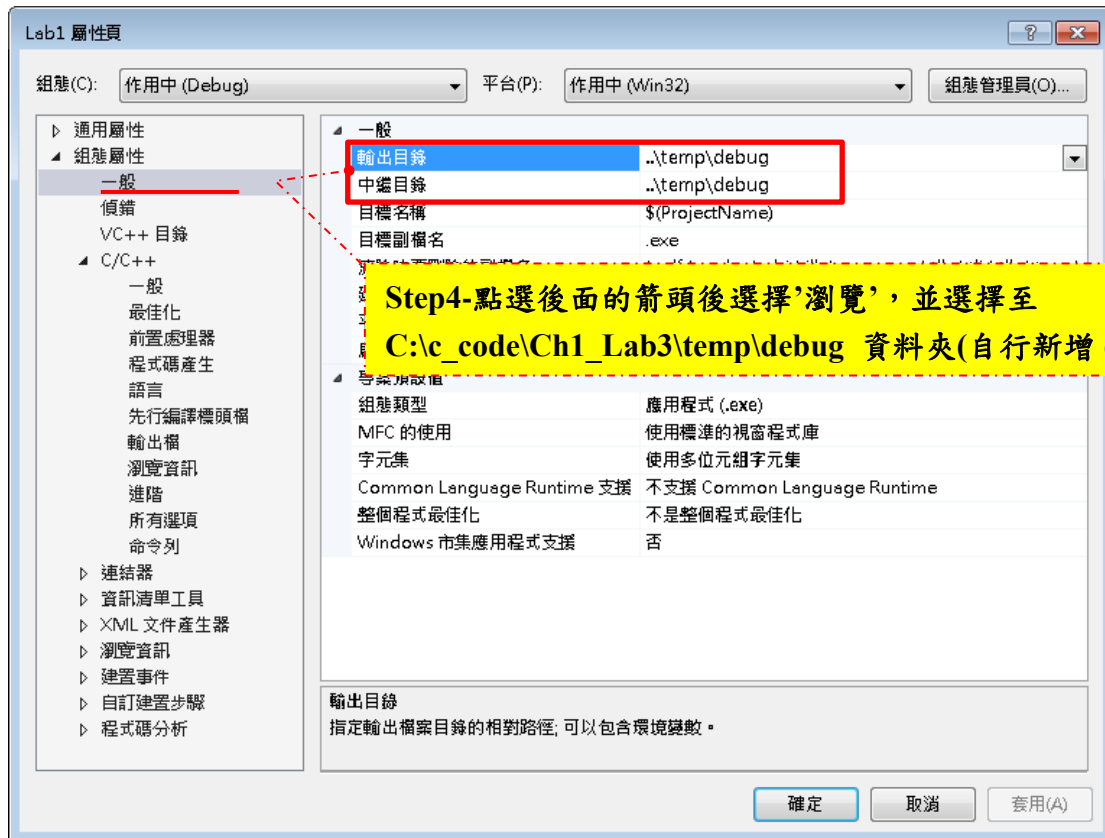


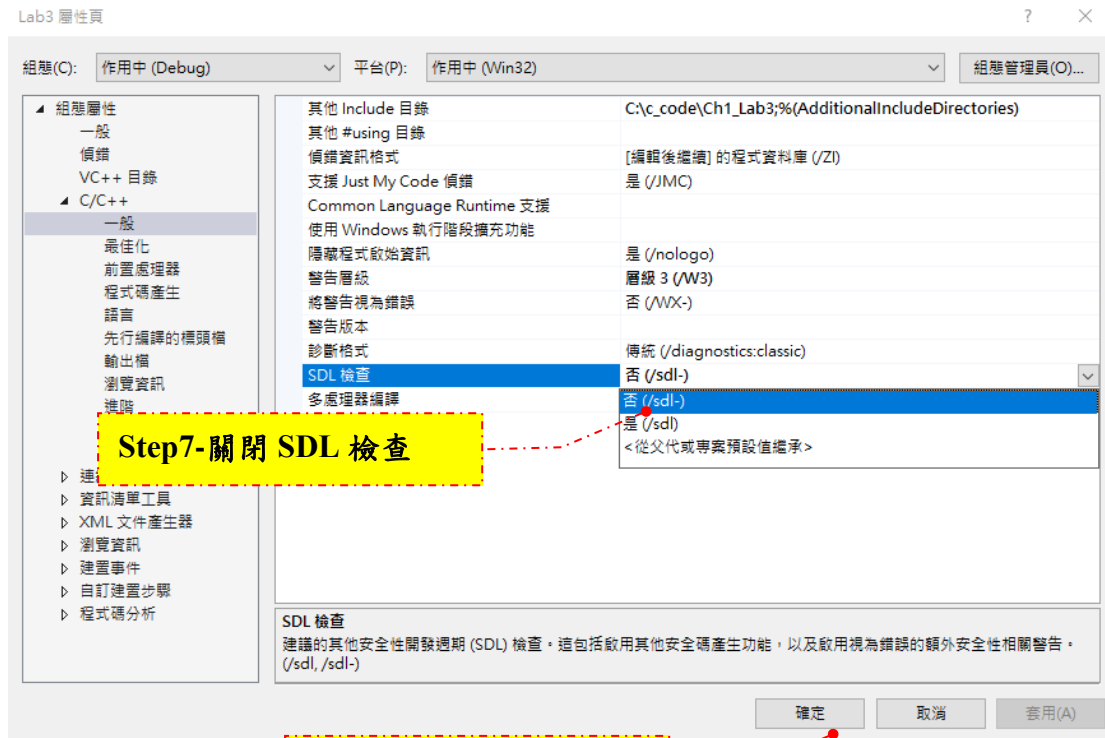
Step1-在左上角“方案總管”欄 Lab3 裡，“來源檔案”上按右鍵，選“重新命名(M)”，命名為“source”，再將“標頭檔”重新命名為“include”

點選“資源檔”，按右鍵選“刪除(D)”，完成後方案總管如下圖









3. 撰寫 C 語言程式

如果透過「方法 A. 透過 Github Classroom 下載並開啟專案」建立專案，直接從此步驟繼續實驗。

The screenshot shows the Microsoft Visual Studio interface for a project named 'Lab3'. The 'Solution Explorer' on the left shows a folder structure with 'include' and 'source' subfolders. The 'source' folder contains a file named 'Main.c', which is highlighted with a red dot. A red dashed line connects this dot to a yellow callout box labeled 'Step1-點擊兩下開啟 Main.c'. The 'Main.c' file is open in the editor, showing C code that includes 'stdio.h' and 'stdlib.h', and contains a 'main' function. The code prompts the user to enter two integers and checks if they are equal or not. A red dot is placed on the 'main' function definition, with a red dashed line connecting it to a yellow callout box labeled 'Step2-於 Main.c 頁面下撰寫程式'. Another red dot is placed on the 'printf' statement within the 'main' function, with a red dashed line connecting it to a yellow callout box labeled 'Step3-在此處撰寫 C 語言程式'. The bottom status bar indicates '第 6 行' (Line 6) and '第 14 欄' (Column 14).

Step1-點擊兩下開啟 Main.c

Step2-於 Main.c 頁面下撰寫程式

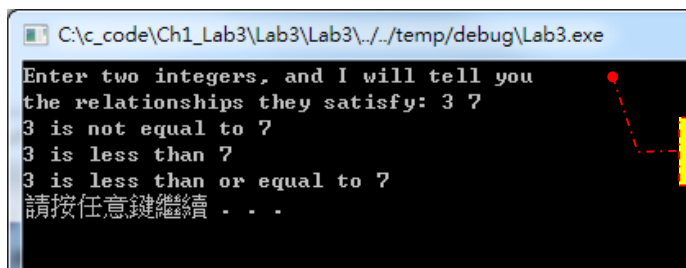
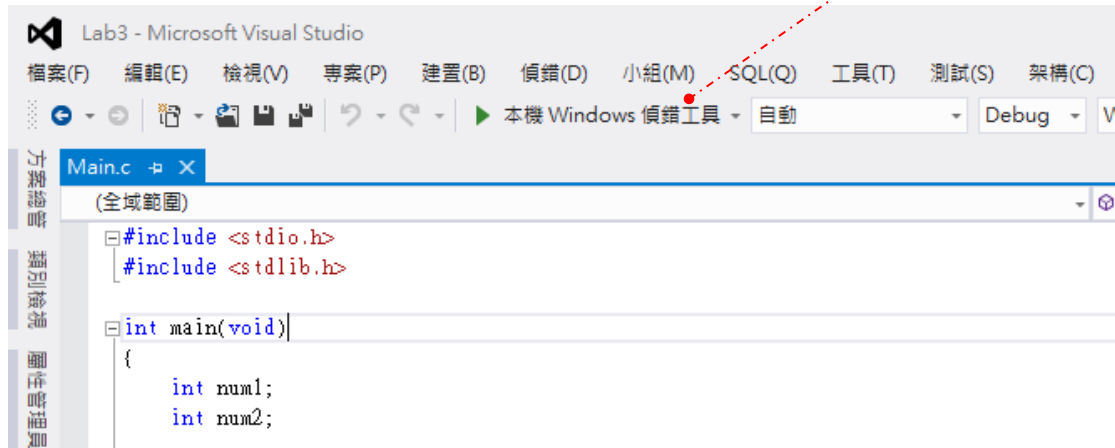
Step3-在此處撰寫 C 語言程式

Main.c 程式碼：

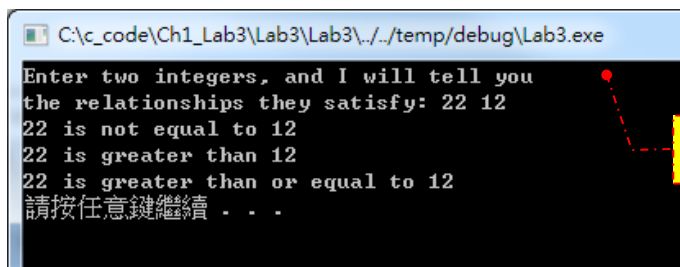
```
main.c X
Lab3 (全域範圍)
1  #include <stdlib.h>
2  #include <stdio.h>
3
4  int main(void)
5  {
6      printf("Enter two integers, and I will tell you\n");
7      printf("the relationships they satisfy:\n");
8
9
10     int num1;
11     int num2;
12
13     scanf("%d%d", &num1, &num2);
14
15     if (num1 == num2)
16     {
17         printf("%d is equal to %d\n", num1, num2);
18     }
19     if (num1 != num2)
20     {
21         printf("%d is not equal to %d\n", num1, num2);
22     }
23     if (num1 < num2)
24     {
25         printf("%d is less to %d\n", num1, num2);
26     }
27     if (num1 > num2)
28     {
29         printf("%d is greater to %d\n", num1, num2);
30     }
31     if (num1 <= num2)
32     {
33         printf("%d is less than or equal to %d\n", num1, num2);
34     }
35     if (num1 >= num2)
36     {
37         printf("%d is greater than or equal to %d\n", num1, num2);
38     }
39
40     system("pause");
41     return 0;
42
43 }
```


4. 執行與測試程式結果

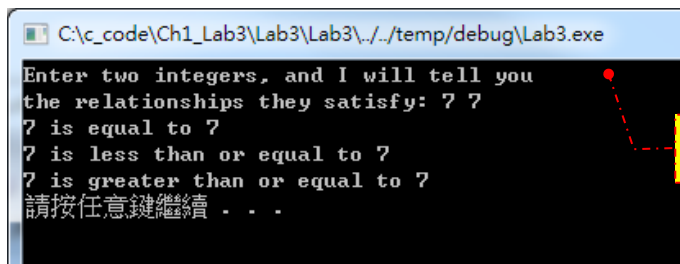
Step1-點選開始偵測，進行偵測



Step2-輸入樣本：3 和 7



Step3-輸入樣本：22 和 12



Step4-輸入樣本：7 和 7

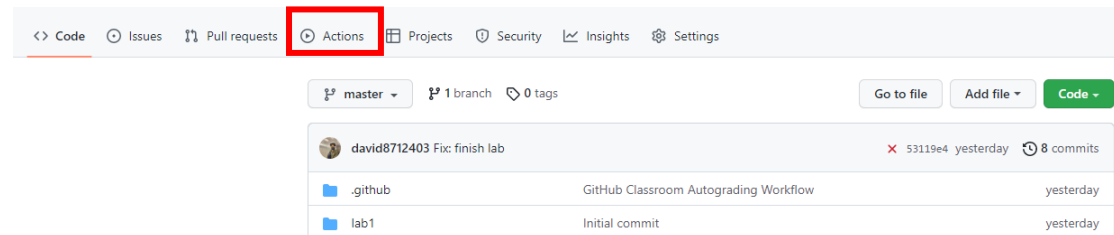
5. 上傳實驗至 Github Classroom

請參考從 Github Clone 專案中 README.md 檔案的**上傳專案說明**，將專案透過 Git 指令 push 到 Github classroom

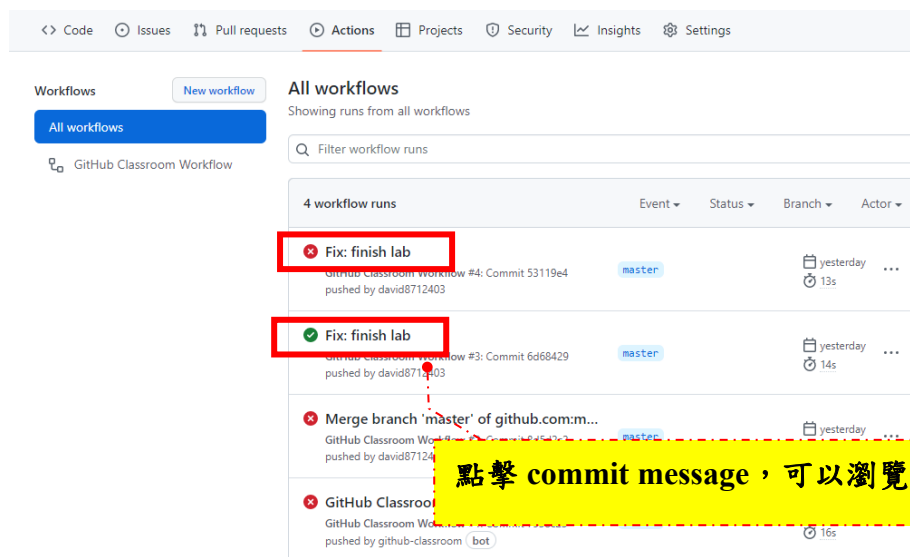
6. 觀察 Github Action 評分

在每次有新的 commit push 到 Github Classroom 時，會觸發定義好的 action 流程，會自動將程式編譯後執行，判斷是否執行正確。

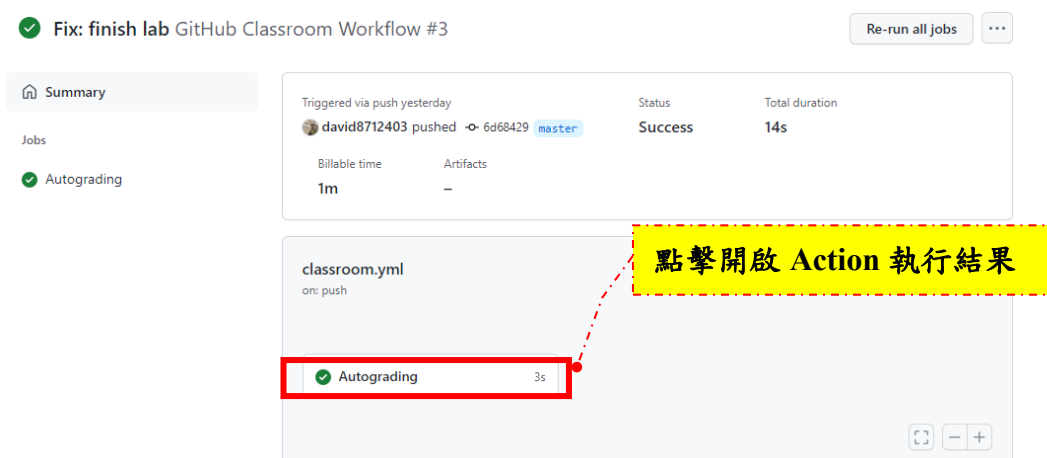
Step1. 進入 Github classroom 實驗的 repository，點擊 Action



Step2. 觀察自己每次 commit 時，action 的輸出及批改結果



Step3. 瀏覽 Action 執行結果



自動評分執行成功情況。本次 Lab 有 3 個 test case，全數通過會得到 100 分

Autograding
succeeded 1 hour ago in 11s

展開 Run education/autograding@v1

Run education/autograding@v1

```
38  the relationships they satisfy:
39  10 is not equal to 20
40  10 is less to 20
41  10 is less than or equal to 20
42  Pass: Output is correct
43
44  All tests passed.
45
46  ✓ test2
47
48  📄 test3
49
50
51  bash test3.sh
52  Running tests...
53
54  sh: 1: pause: not found
55  Pass: Program exited zero
56  Output:
57  Enter two integers, and I will tell you
58  the relationships they satisfy:
59  40 is not equal to 20
60  40 is greater to 20
61  40 is greater than or equal to 20
62  Pass: Output is correct
63
64  All tests passed.
65
66  ✓ test3
67
68
69  ::***::
70
71  All tests passed
72
73  🌟🌟💙💙👤💙🌟🌟💙🌟🌟👤💙🌟🌟
74
75  Points 100/100
76
```

編譯及輸出結果