

實驗項目-環境建置、開新專案與撰寫程式

一、本節目的：

- 建置 VC 2017 的開發環境

二、設計重點：

- 教你如何於 Windows 下建置 VC2017 的開發境與撰寫程式

三、設計步驟：

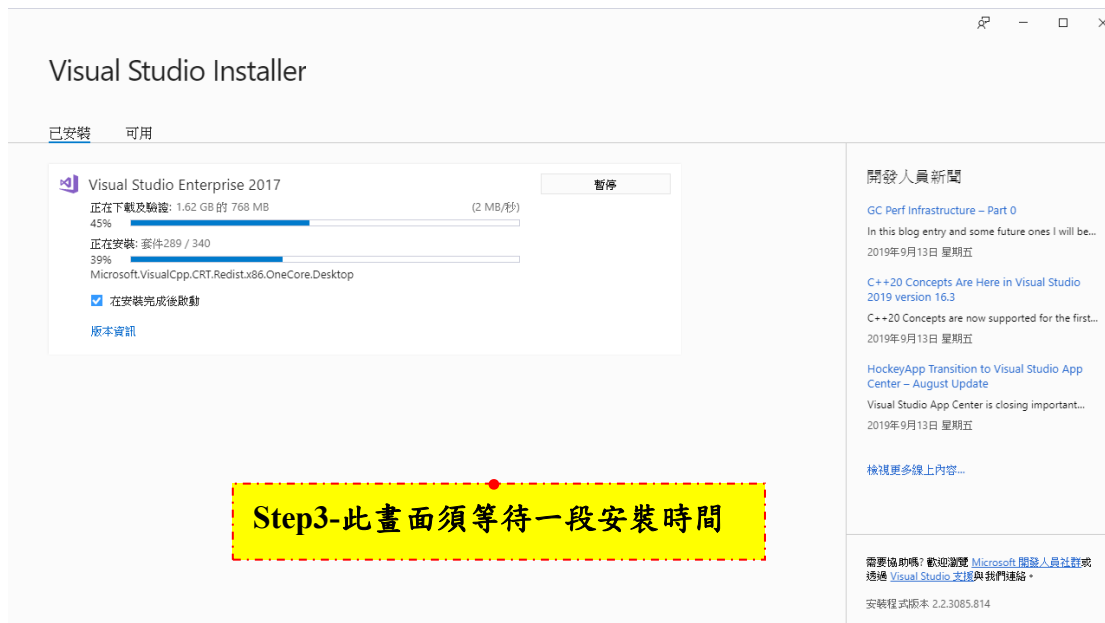
1. 安裝 Microsoft Visual Studio 2017
2. 下載此版本: [Trial] Visual Studio Enterprise 2017 (version 15.9)
3. 下載網址:

https://my.visualstudio.com/Downloads?q=visual%20studio%202017&wt.mc_id=o~msft~vscom~older-downloads

4. 軟體金鑰:

<http://140.124.73.33/109%20%E9%AB%98%E9%9A%8E%E7%A8%8B%E5%BC%8F%E8%AA%9E%E8%A8%80/Software/>







Step5-開發設定選擇“Visual C++”，色彩佈景主題依個人喜好選擇

Step6-點選“啟動 Visual Studio(S)”

5. 新增專案：

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. 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-lab1' folder. A yellow callout box says: 開啟 ch1-lab1 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...		
lab1.sh	2022/9/28 下午 0...	Shell Script	1 KB
lab2.sh	2022/9/28 下午 0...	Shell Script	1 KB
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 'lab1' folder. A yellow callout box says: 開啟 lab1 資料夾

名稱	修改日期	類型	大小
lab1	2022/9/28 下午 0...	檔案資料夾	
source	2022/9/28 下午 0...	檔案資料夾	
.gitignore		文件	5 KB

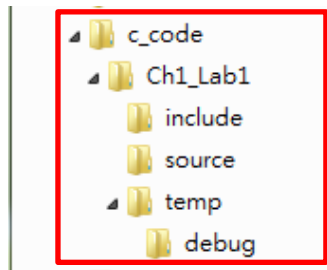
The fourth screenshot shows a file explorer with a table of files. A red dot is on the 'lab1.sln' file. A yellow callout box says: 點擊 lab1.sln 專案檔案

名稱	修改日期	類型	大小
.vs	2022/9/13 下午 0...	檔案資料夾	
lab1	2022/9/13 下午 0...	檔案資料夾	
x64	2022/9/13 下午 0...	檔案資料夾	
lab1.sln	2022/9/13 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 7.撰寫 C 語言程式

方法 B. 透過 Visual Studio 新建專案

Step1-在 C:\內建立一個資料夾“c_code”，於 c_code 資料夾內新增名為“Ch1_Lab1”的資料夾，再於 Ch1_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



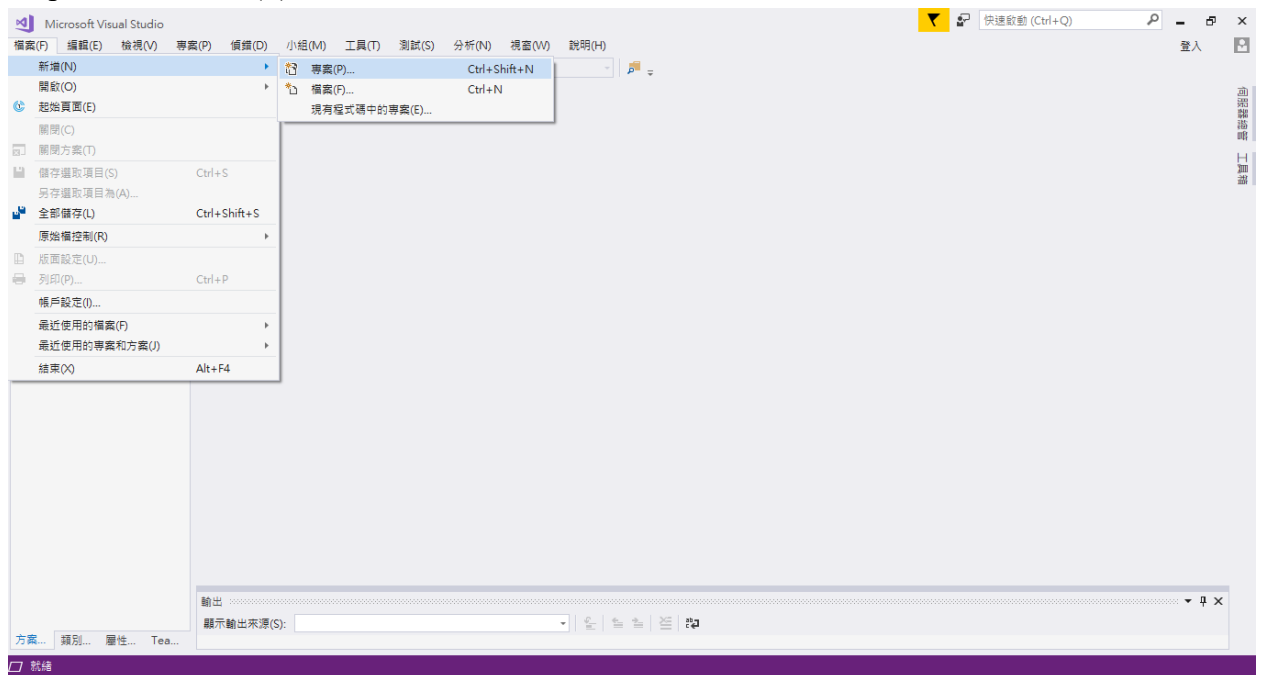
Step2-點選“開始”->“所有程式”尋找“Visual Studio 2017”資料夾中的“Visual Studio 2017”



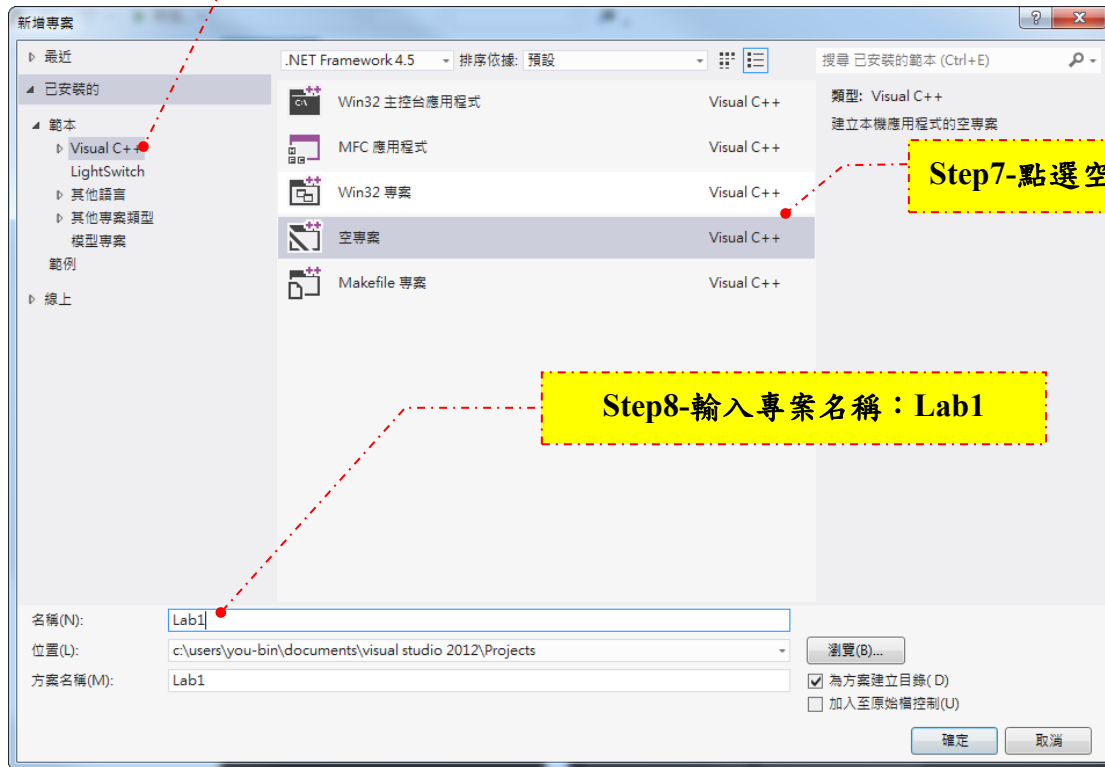
Step3-開啟 Microsoft Visual Studio 視窗畫面後點選左上角“開始(F)”

Step4-點選“新增(N)”

Step5-點選“專案(P)”

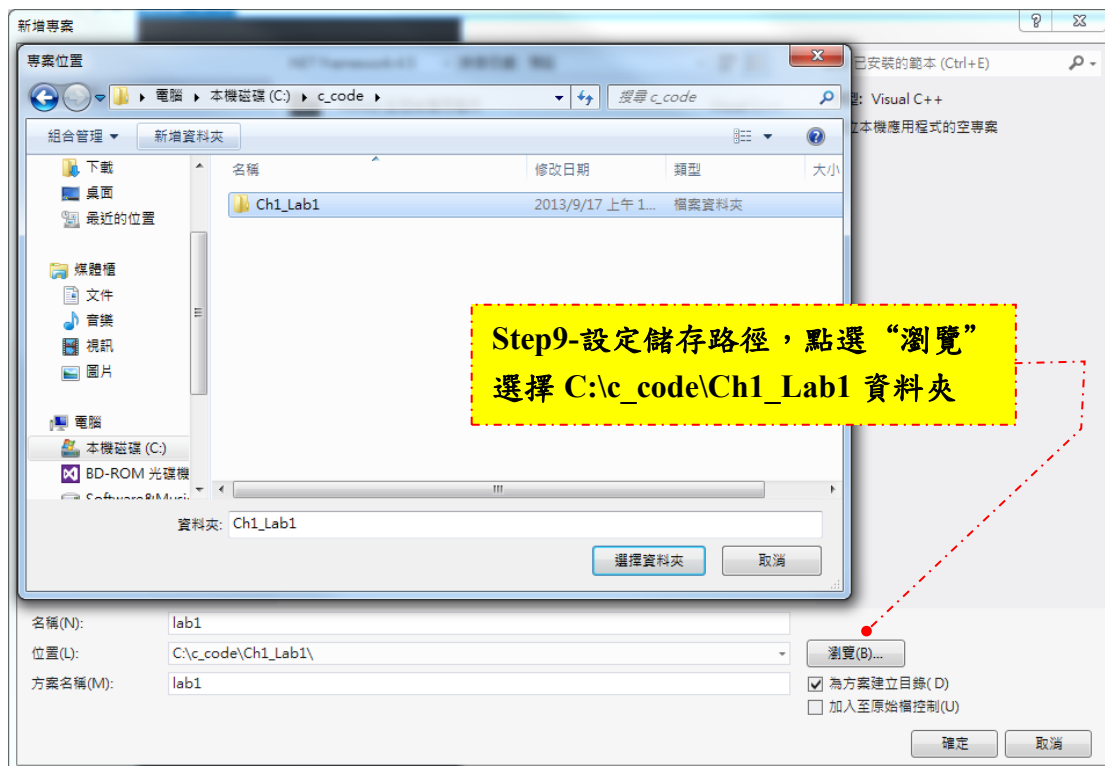


Step6-點選 Visual C++

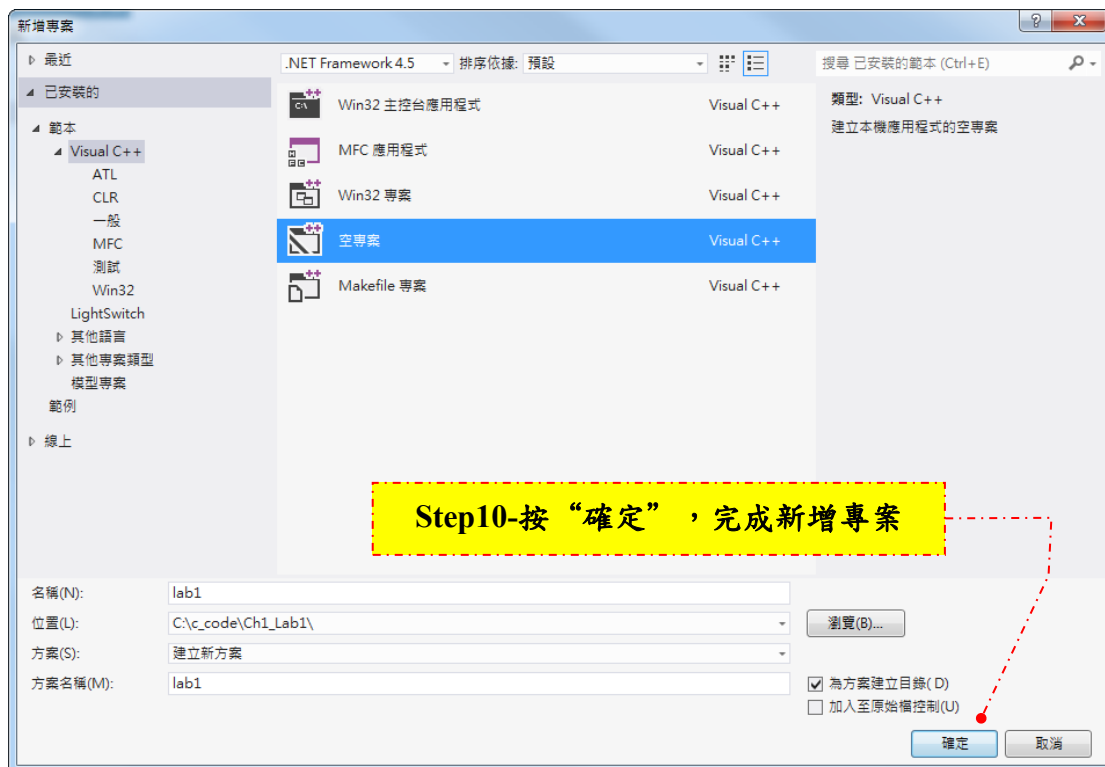


Step7-點選空專案

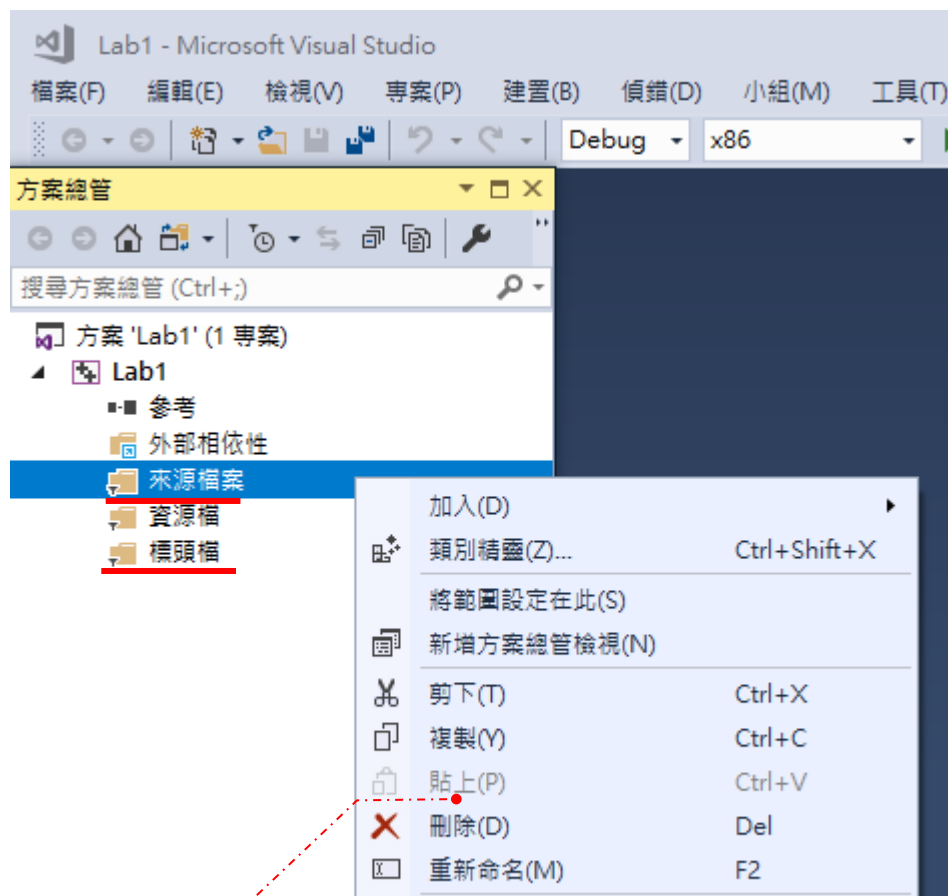
Step8-輸入專案名稱：Lab1



**Step9-設定儲存路徑，點選“瀏覽”
選擇 C:\c_code\Ch1_Lab1 資料夾**

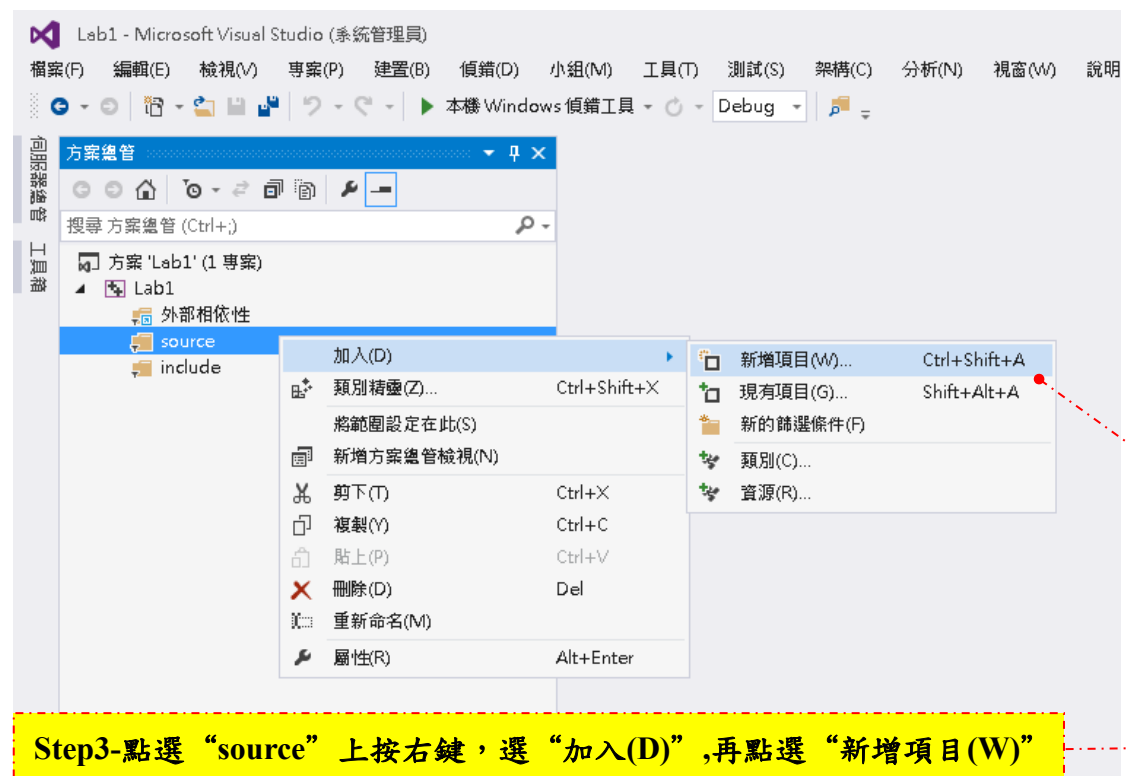
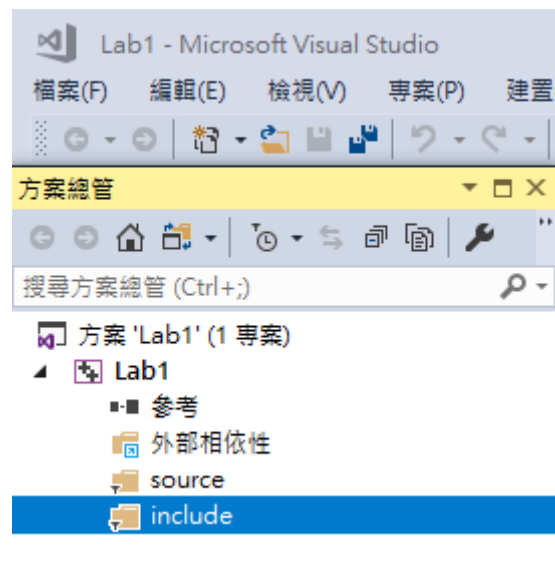


6. 新增 .c 檔、路徑設定

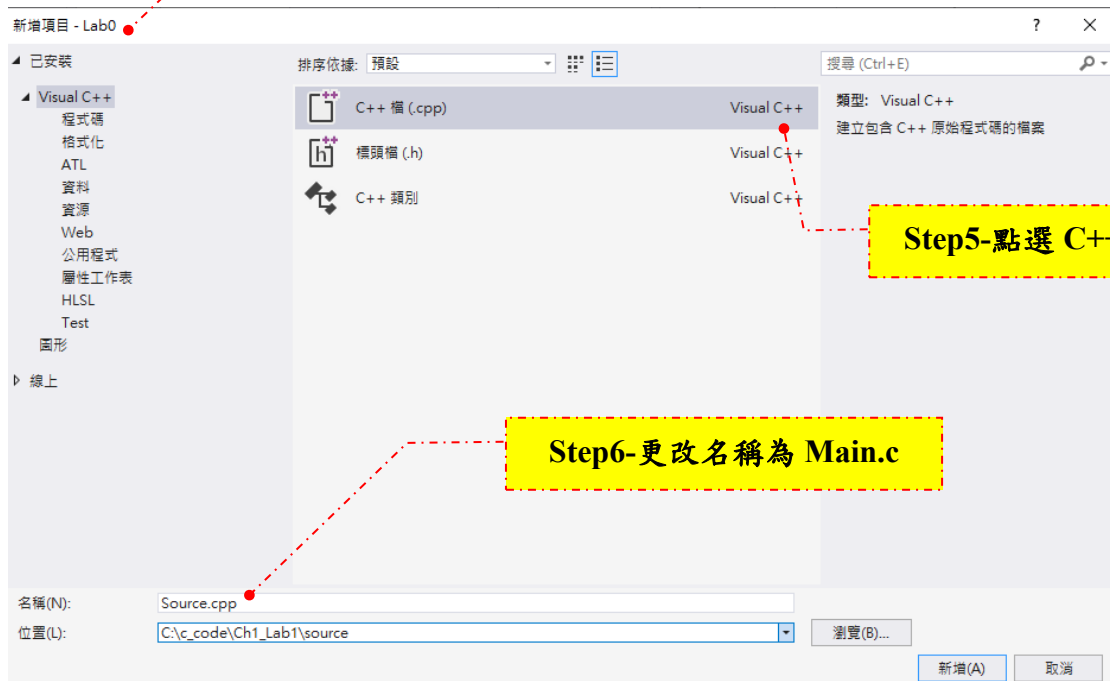


Step1-在左上角“方案總管”欄 Lab1 裡，“來源檔案”上按右鍵，選“重新命名(M)”，命名為“source”，再將“標頭檔”重新命名為“include”

Step2-點選“資源檔”，按右鍵選“刪除(D)”，完成後方案總管如下圖

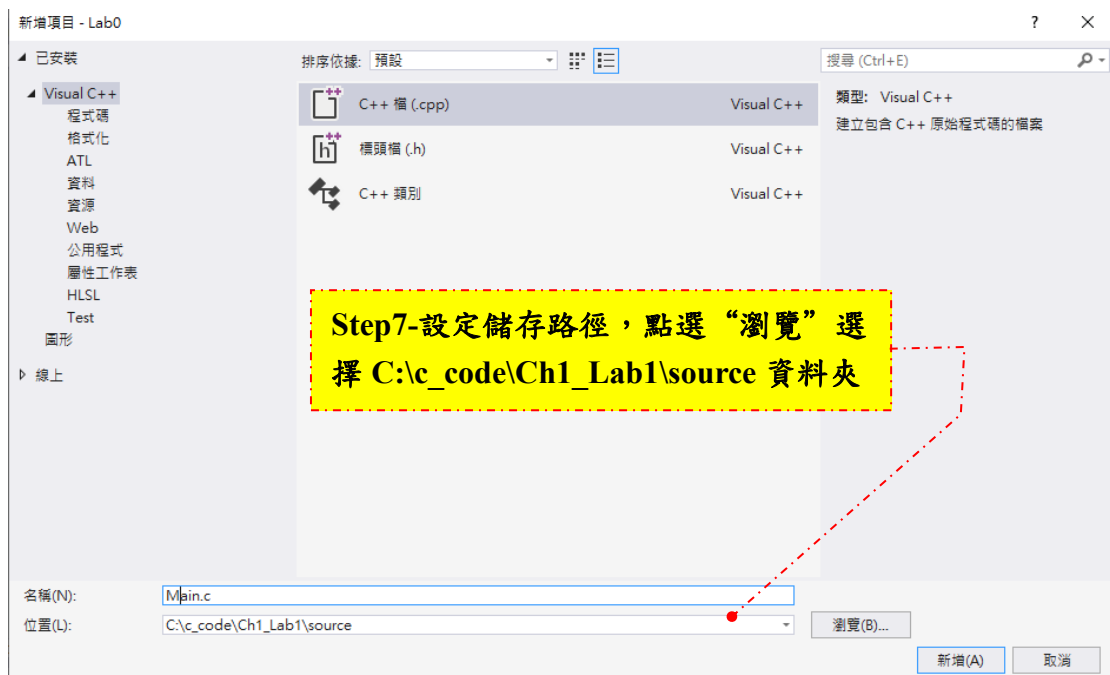


Step4-點選 Visual C++

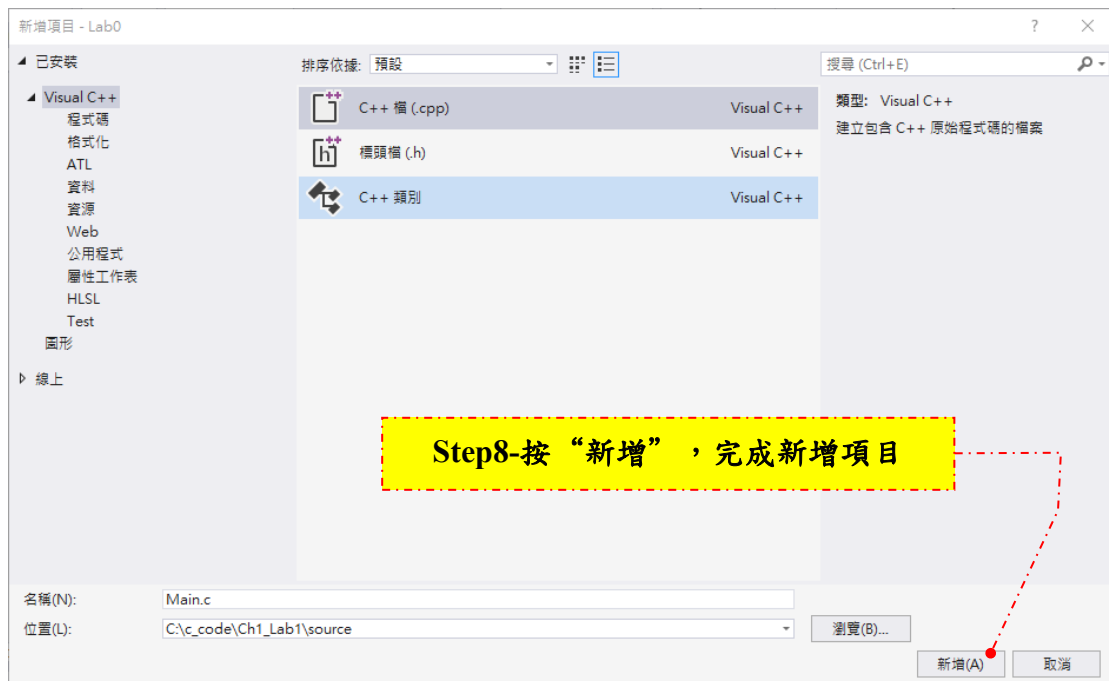


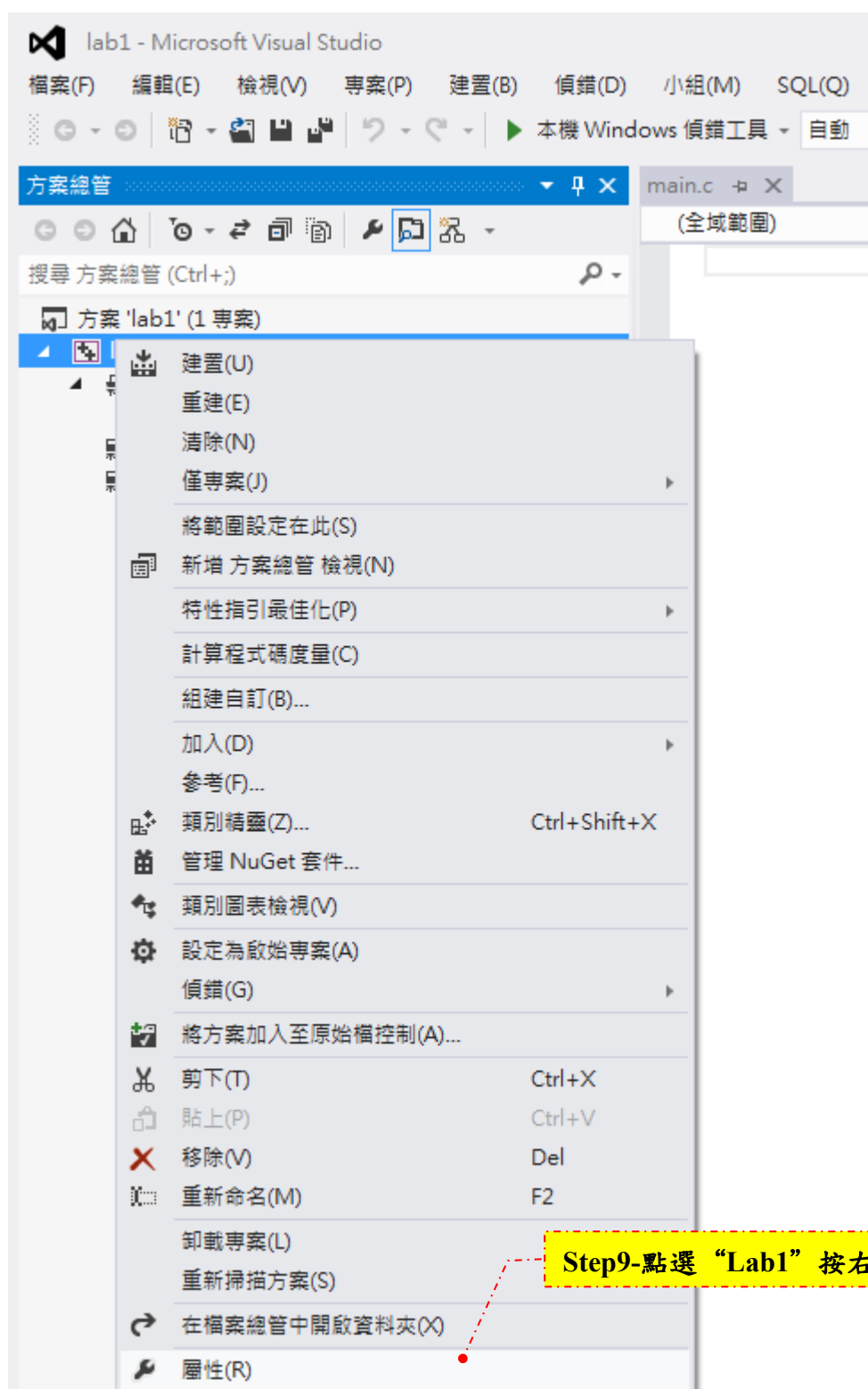
Step5-點選 C++

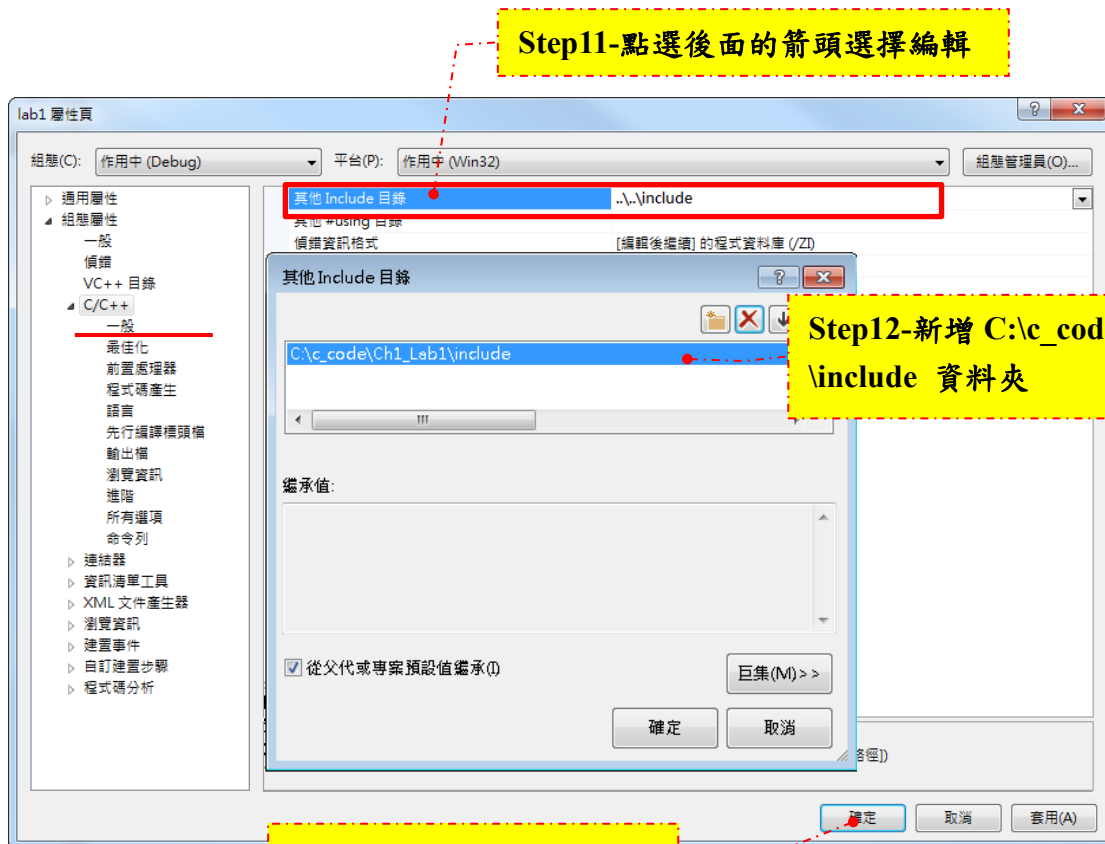
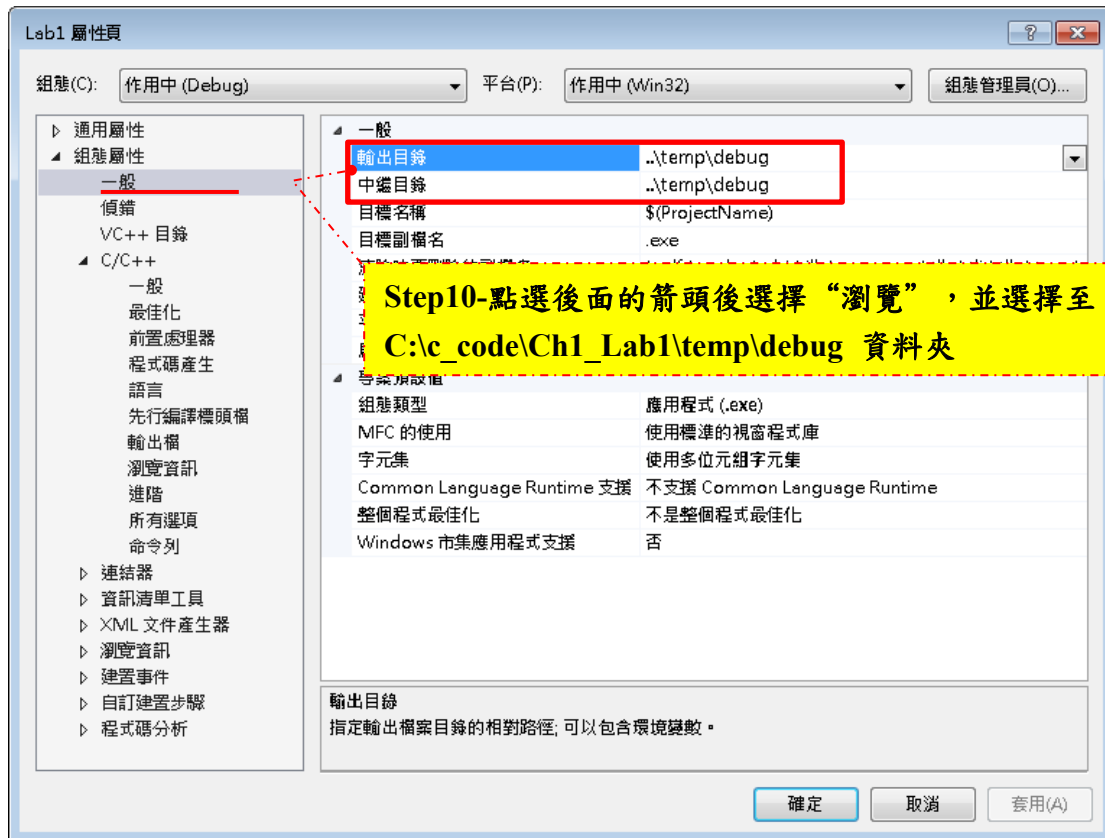
Step6-更改名稱為 Main.c



Step7-設定儲存路徑，點選“瀏覽”選擇 C:\c_code\Ch1_Lab1\source 資料夾

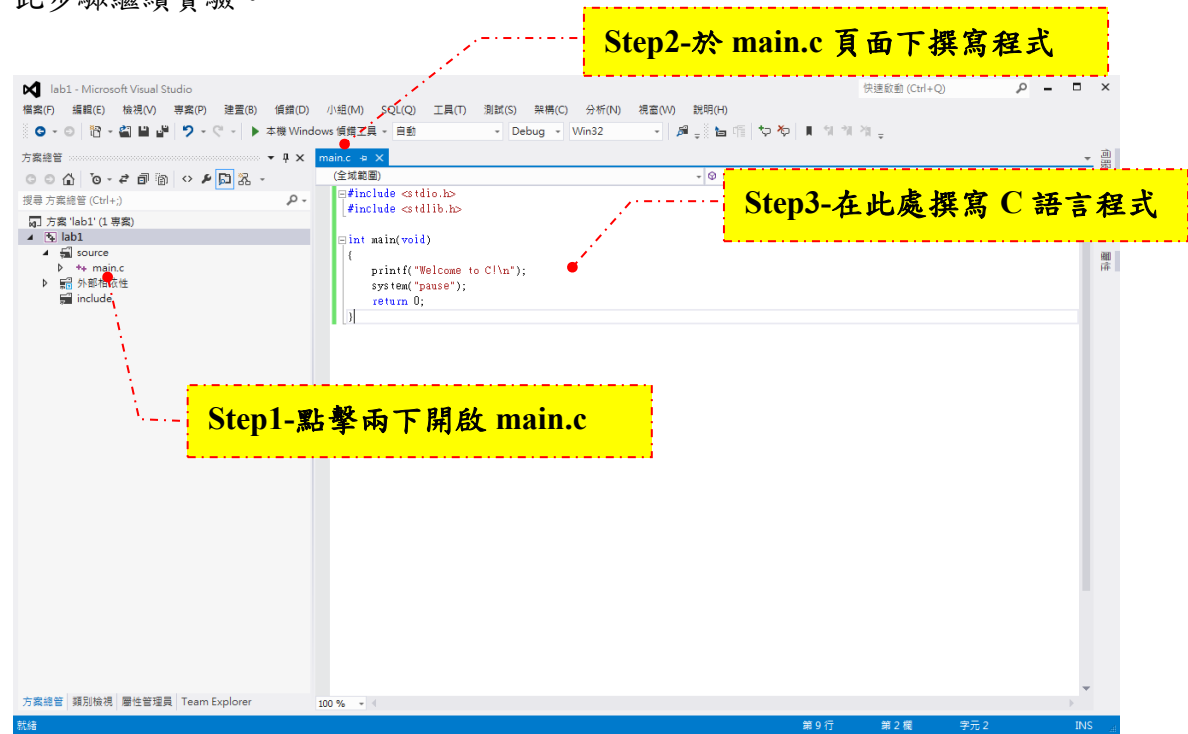




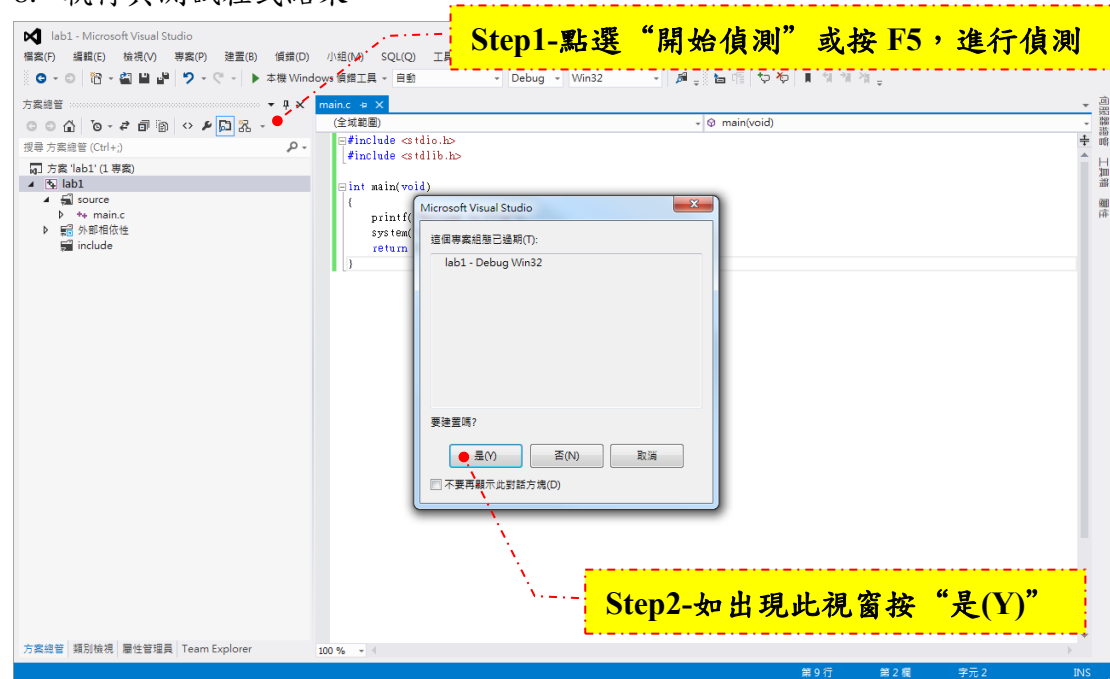


7. 撰寫 C 語言程式

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。



8. 執行與測試程式結果



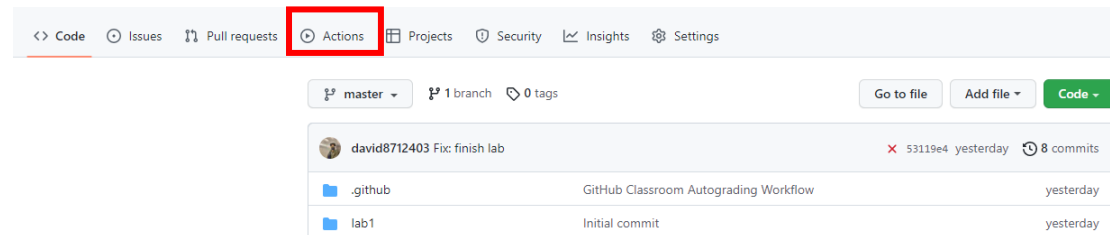
9. 上傳實驗至 Github Classroom

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

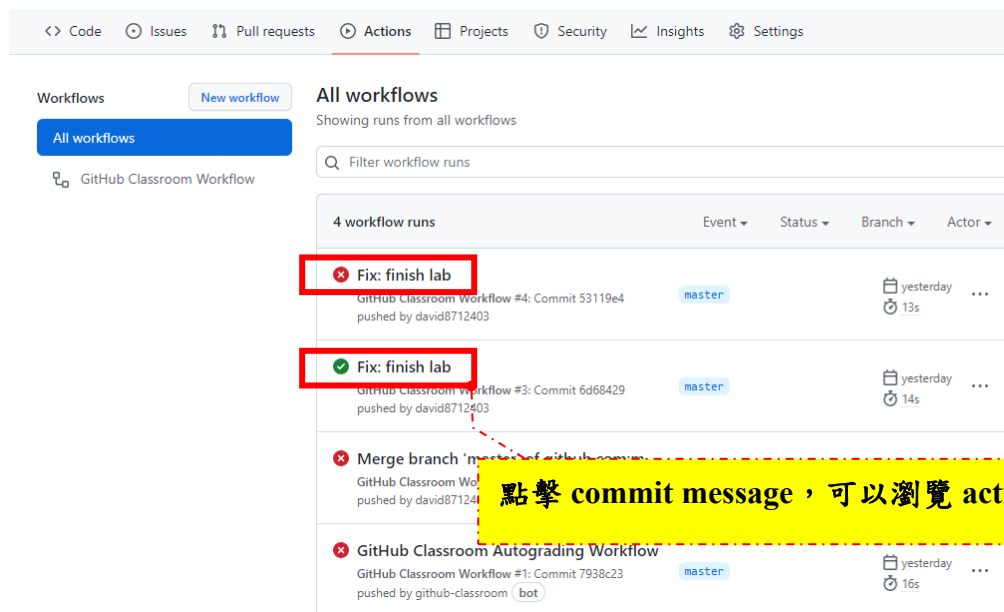
10. 觀察 Github Action 評分

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。

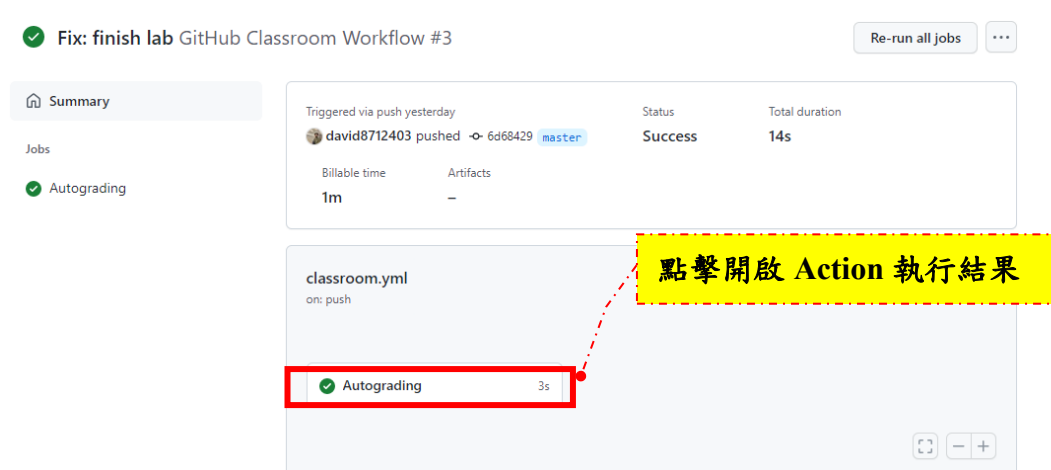
Step1. 進入 Github classroom 實驗的 repository，點擊 Action



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

✓ Fix: finish lab GitHub Classroom Workflow #3 Re-run all jobs ...

Summary

Jobs

✓ Autograding

Autograding
succeeded yesterday in 3s

Search logs

> ✓ Set up job

> ✓ Run actions/checkout@v2 0s

✓ **Run education/autograding@v1** 0s

```
1 ▶ Run education/autograding@v1
4
5 ::stop-commands::***
6
7 test
8
9
10 gcc source/Main.c
11 bash test.sh
12 sh: 1: pause: not found
13 Running tests...
14
15 Pass: Program exited zero
16 Output:
17 welcome to c
18 Pass: Output is correct
19
20 All tests passed.
21
22 ✓ test
23
```

展開 Run education/autograding@v1

編譯及輸出結果

自動評分執行失敗情況

✗ GitHub Classroom Autograding Workflow GitHub Classroom Workflow #1 Re-run jobs ...

Summary

Jobs

✗ Autograding

Autograding
failed 1 hour ago in 5s

Search logs

> ✓ Set up job

> ✓ Run actions/checkout@v2 1s

✗ **Run education/autograding@v1** 0s

```
1 ▶ Run education/autograding@v1
4
5 ::stop-commands::***
6
7 test
8
9
10 gcc source/Main.c
11 bash test.sh
12 Running tests...
13
14 sh: 1: pause: not found
15 Pass: Program exited zero
16 Output:
17 Expected 'Welcome to C!' but got: ''
18 make: *** [Makefile:5: test] Error 1
19
20
21 ✗ test
22 ::error::Error: Exit with code: 2 and signal: null
23
24 ::***::
25
```

展開 Run education/autograding@v1

編譯及輸出結果

失敗原因，輸出內容與預期不符
預期輸出: Welcome to C!
實際輸出: (空字串)