

實驗項目- 製作遞迴式階層函式(recursive-factorial 函式)

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
- 練習以遞迴概念撰寫程式

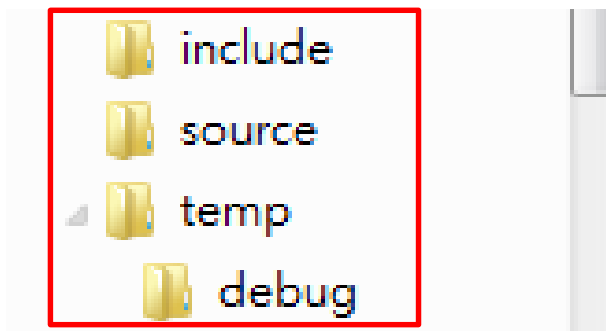
二、 設計重點：

- C 語言的函式應用

三、 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“Ch3_Lab2-2”的資料夾，再於 Ch3_Lab2-2 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\ Ch3_Lab2-2

2. 路徑設定、新增 .c 檔

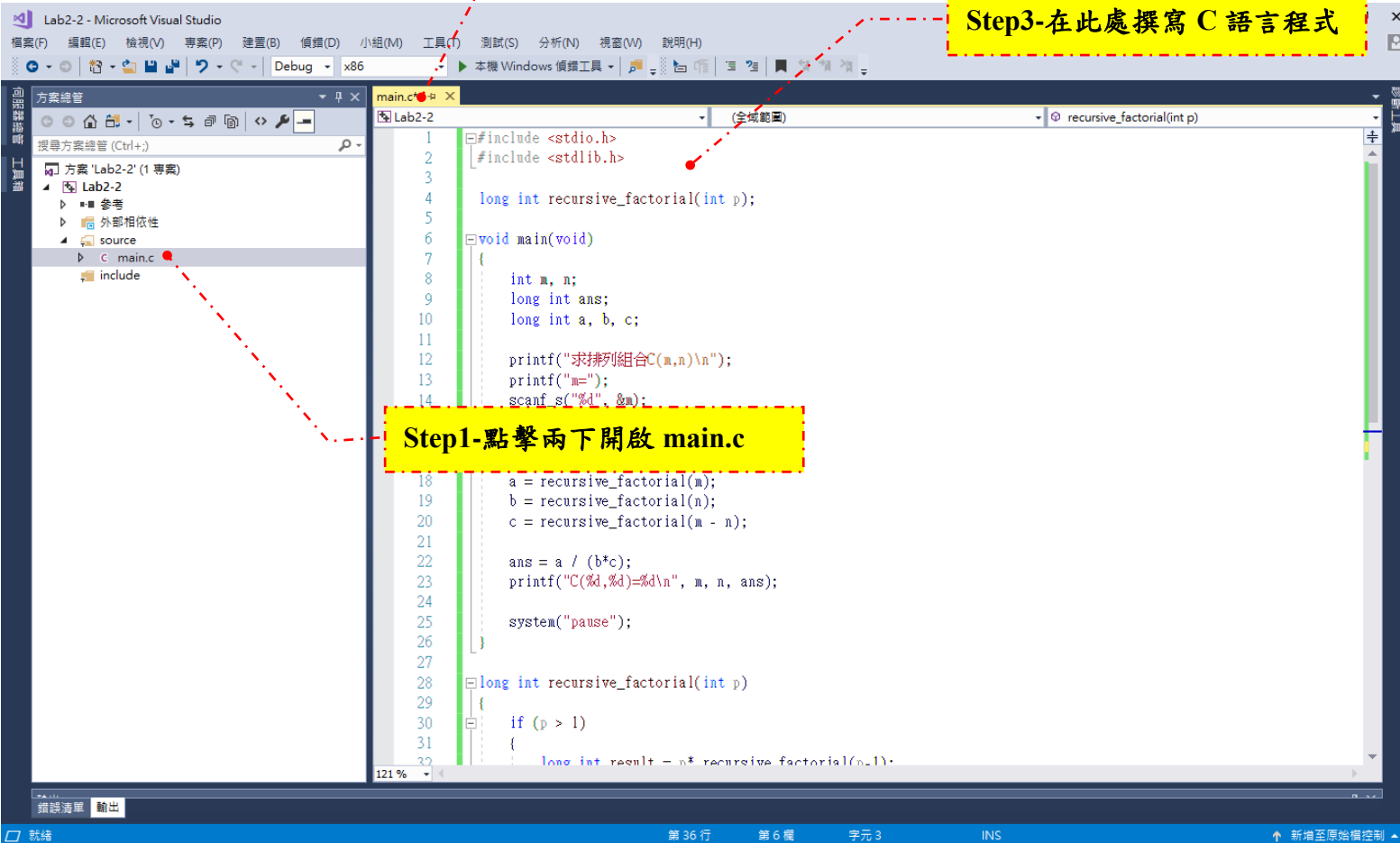
Step1-參照 Ch1_Lab3 中“2. 路徑設定、新增 .c 檔” Step1~Step8，新增 main.c 檔與設定相關屬性設定。

3. 撰寫 C 語言程式

Step2-於 main.c 頁面下撰寫程式

Step3-在此處撰寫 C 語言程式

Step1-點擊兩下開啟 main.c



main.c 程式碼：

```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  long int recursive_factorial(int p);
5
6  void main(void)
7  {
8      int m, n;
9      long int ans;
10     long int a, b, c;
11
12     printf("求排列組合C(m,n)\n");
13     printf("m=");
14     scanf_s("%d", &m);
15     printf("n=");
16     scanf_s("%d", &n);
17
18     a = recursive_factorial(m);
19     b = recursive_factorial(n);
20     c = recursive_factorial(m - n);
21
22     ans = a / (b*c);
23     printf("C(%d,%d)=%d\n", m, n, ans);
24
25     system("pause");
26 }
27
28 long int recursive_factorial(int p)
29 {
30     if (p > 1)
31     {
32         long int result = p* recursive_factorial(p-1);
33         return result;
34     }
35     else
36     {
37         return 1;
38     }
39 }
```

4. 執行與測試程式結果

Step1-點選開始偵測，進行偵測

