

實驗項目 - 結構(structure)中陣列,指標與函式的使用

一、本節目的：

- 學習使用結構(structure)方法
- 學習如何在結構中使用陣列、指標與函式

二、設計重點：

- 利用陣列、指標與函式應用在結構方法

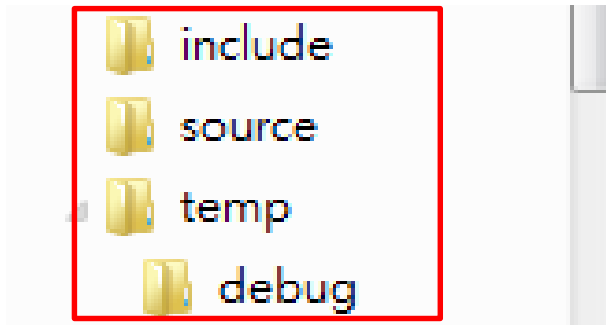
三、實驗題目：

- 參考 CH6 投影片-p15 程式，建立一結構 struct _student 內包含學號(num[7])、班級(clas[20])、姓名(name[20])及各科成績(score[3])，與一函式 sub()用來將學生成績加 10 分。

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“Ch6_Lab3”的資料夾，再於 Ch6_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



Step2-參照 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\ Ch6_Lab3

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

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

```
#include <stdio.h>
#include <stdlib.h>

typedef struct _student
{
    char num[7];        //學號
    char clas[20];      //班級
    char name[20];      //姓名
    int score[3];       //各科成績score[0]國文/score[1]英文/score[2]數學
} student;

void sub(student *p);

int main(void)
{
    {"970101", "四子一甲", "王小名", 61, 71, 81},
    {"970102", "四子一甲", "李中雄", 92, 82, 72},
    {"970103", "四子一甲", "張大成", 73, 63, 83}
};

student *ps;
//ps指向student陣列的開頭，也就是student的位址
```

Main.c 程式碼：

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 typedef struct _student
5 {
6     char num[7];    //學號
7     char clas[20];  //班級
8     char name[20];  //姓名
9     int score[3];   //各科成績score[0]為國文,score[1]為英文,score[2]為數學
10 } student;
11
12 void sub(student *p);
13
14 void main(void)
15 {
16
17     int i;
18     student stu[3]={{"970101","四子一甲","王小明",61,71,81},
19                     {"970102","四子一甲","李中雄",92,82,72},
20                     {"970103","四子一甲","張大成",73,63,83}};
21
22     student *ps;
23     ps=stu;    //ps指向stu陣列的開頭，也就是stu[0]的位址
24     printf("\n");
25     printf("===== \n");
26     printf(" 學號      班級      姓名      國文  英文  數學 \n");
27     printf("===== \n");
28     for (i=0;i<=2;i++)
29     {
30         printf("%6s %-10s %-8s  %4d  %4d  %4d \n", (ps+i)->num, (ps+i)->clas,
31              (ps+i)->name, (ps+i)->score[0], (ps+i)->score[1], (ps+i)->score[2]);
32     }
33     printf("    呼叫函式前 \n \n");
34     sub(ps);
35     printf(" \n    呼叫函式後 \n");
36     for (i=0;i<=2;i++)
37     {
38         printf("%6s %-10s %-8s  %4d  %4d  %4d \n", (ps+i)->num, (ps+i)->clas,
39              (ps+i)->name, (ps+i)->score[0], (ps+i)->score[1], (ps+i)->score[2]);
40     }
41     printf("\n");
42     system("pause");
43 }
44
45 void sub(student *p)
46 {
47     int i;
48     for (i=0;i<=2;i++)
49     {
50         (p+i)->score[0] +=10;
51         (p+i)->score[1] +=10;
52         (p+i)->score[2] +=10;
53     }
54 }
```

4. 執行與測試程式結果

Step1-點選開始偵測，進行偵測

