

資訊工程學系	資料結構應用	文件編號：	LAB2
		發佈日期：	2022/03/04

LAB 2

陣列應用範例 – 氣泡排序法

長榮大學
資訊工程學系

班級：資工 2B

姓名：郭智榮

學號：109B30612

日期：2022/03/04

版本校定紀錄：

版本	更新紀錄	發佈日期
0.0.0.0	初版完成	2022/03/04

資訊工程學系	資料結構應用	文件編號：	LAB2
		發佈日期：	2022/03/04

一. 實驗需求：

(一). 題目說明

使用氣泡排序法製作一個將 5 個整數資料由小到大排序的程式。

程式執行時同時輸入 5 個整數，每個整數間以空白隔開。

輸出 5 個整數由小到大的排序的結果。

(二). 演算法

```

print("== 氣泡排序法(由小到大排序) ==\n\n") //題目要求格式
print("... 請連續輸入五個數值(資料間空一格)...\n\n") //題目要求格式

print(" 排序前:")
int number_arrays[5] //建立一個大小為 5 的陣列，存放輸入的資料
for(int i = 0 ; i < 5 ; i++){
    number_arrays[i] = input number //將輸入的整數依序存放到陣列
}

for(int i = 4 ; i > 0 ; i++){ //每次執行都不做前一次的最後一格
    for(int j = 0 ; j < i ; j++){ //依照 i 的值決定 j 的停止點
        if(number_arrays[j] > number_arrays[j+1]){ //比較大小
            int stay = number_arrays[j] //用於交換數值
            number_arrays[j] = number_arrays[j+1]
            number_arrays[j+1] = stay
        }
    }
}

print("\n 排序後:")
for(int i = 0 ; i < 5 ; i++){
    print(number_arrays[i]) //輸出排序結果
}

```

資訊工程學系	資料結構應用	文件編號：	LAB2
		發佈日期：	2022/03/04

二. 完整程式碼：

```

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

int main(){
    int number[5];
    printf("== 氣泡排序法(由小到大排序) ==\n\n");
    printf("...請連續輸入五個數值(資料間空一格)...\n\n");

    int n;
    printf(" 排序前：");
    for(int i = 0 ; i < 5 ; i++){
        scanf("%d", &n);
        number[i] = n;
    }

    for(int i = 4 ; i > 0 ; i--){
        for(int j = 0 ; j < i ; j++){
            if(number[j] > number[j+1]){
                int stay = number[j];

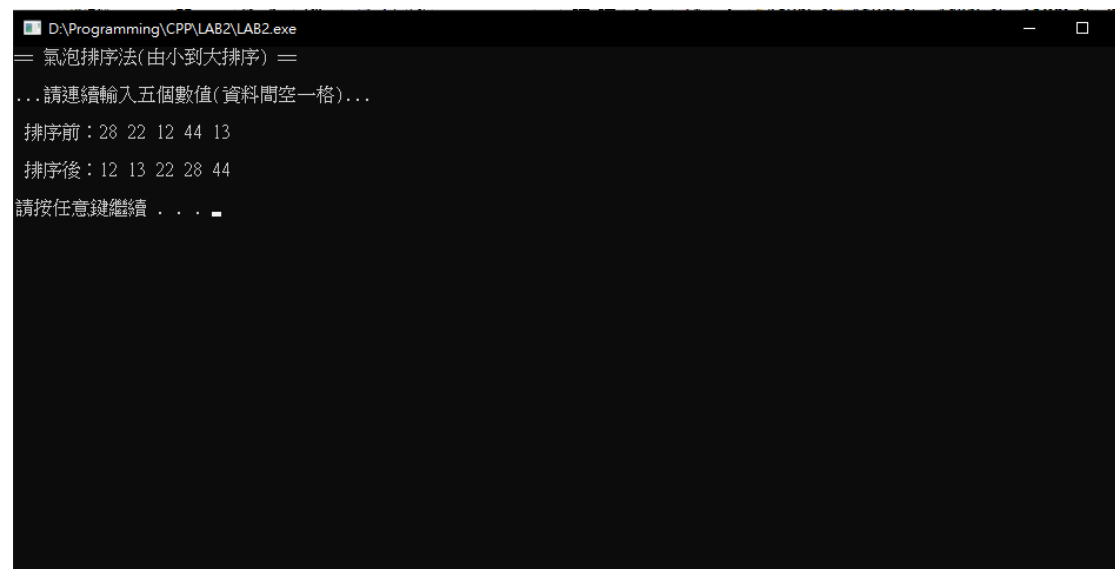
                number[j] = number[j+1];
                number[j+1] = stay;
            }
        }
    }

    printf("\n 排序後：");
    for(int i = 0 ; i < 5 ; i++){
        printf("%d ", number[i]);
    }
    printf("\n\n");
    system("PAUSE");
    return 0;
}

```

資訊工程學系	資料結構應用	文件編號：	LAB2
		發佈日期：	2022/03/04

三. 輸入及輸出結果：



```

D:\Programming\CPP\LAB2\LAB2.exe
== 氣泡排序法(由小到大排序) ==
...請連續輸入五個數值(資料間空一格)...
排序前：28 22 12 44 13
排序後：12 13 22 28 44
請按任意鍵繼續 . . .

```

四. 心得與討論：

本次氣泡排序法主要用 C 語言撰寫，之前就有稍微學過一點點基本的語法，加上上學期的物件導向程式設計時就有學過氣泡排序法，因此這次 LAB2 的作業並不是太困難；希望之後有機會多學習其他種排序方式，能更增加基礎演算法的觀念。