

Code: <https://github.com/jack2012aa/sorting>

schedule.h: 定義 structure schedule

data\_generator.c: 隨機生成 30 筆 schedule 資料，不同欄間用空格區隔，不同筆資料間以換行區隔，寫入 data.txt 中

dataio.c: 讀取 data 和寫入 result

compar.c: 比較 struct schedule 的方法

sorting\_algorithm.c: 不同 sorting 的函式

sorting.c: 讀取資料進行排序後再寫入 result

由於這次需要寫的函式比較多，分開寫成不同文件會讓程式清楚很多。

因為 compar 函式的參數是 void 指標型式，所以 arg 不能直接拿來 dereference。可以宣告新的 schedule 指標指向 arg 即可使用。除此之外記得 a -> begin\_date 中的箭頭本身就帶有 dereference 的涵義，所以不需要再加星號。

swap 函數採用直接複製記憶體的策略，當 structure 的的大小非常大時每排序一次就要移動一次很耗時間。可以用兩種方法來改善：

1. 將 structure 中的變數全部宣告成指標型式，減少 structure 本身的體積
2. 製造一個 structure pointer array 指向原本 array 的內容，將 pointer array 排序後再依照其順序輸出排序好的內容

第二種方式在很多地方都會用來提升運算的速度，實作時只要將 compar() 函式中原本只像 void arg 的指標變成 pointer of pointer，後面再稍作修改就行。