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.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。