

你要記得的是 qsort 的使用模型:

1. 需要排序的所有資料要放在塊連續的記憶體中
2. 每一筆資料的長度都是固定的
3. 需要寫一個函式來比對兩筆資料的大小

這才是你在練習過以後應該要知道的東西 (你花一些時間練習好英文打字以後, 這種程式片段需要直接思考, 直接寫出來, 不需要邊看編寫, 我們也會慢慢地把關鍵程式碼從網頁中移除)

也許你從來沒有看過上面範例的用法, 或是覺得這樣子的 C 程式的語法上有點生疏, 分成三部份解釋一下:

### 1. 指標型態的強制轉換

```
int x;  
int *ptr = &x;
```

上面的兩列程式中, 變數 ptr 裡面放的是整數變數 x 的記憶體位址 &x

下面的敘述 \*ptr + 1 因為 ptr 存放的是整數變數的記憶體位址, 所以編譯器知道 \*ptr 需要由那個位址拿出連續的四個位元組, 然後用二的補數(整數)來解釋那四個位元組, 最後執行整數的加法來加 1

如果程式改成 `*(double *)ptr + 1.0`, ptr 前面加上 (double \*) 使得編譯器認為 (double \*)ptr 是一個 double 變數的記憶體位址, \*(double \*)ptr 則要求由那個位址取出連續的八個位元組, 以 IEEE 754 浮點數格式解釋那八個位元組, 然後執行浮點數的加法來加上 1.0;

請注意, (double \*)ptr 這個型態的強制轉換敘述其實沒有更改位址的數值, 但是改變了後續的「間接資料存取運算子」\* 的運作方法

2. **qsort 的第四個參數是一個函式指標變數** (請注意一個函式的參數如果是變數的話, 呼叫這個函式的時候可以藉由這個參數來調整函式處理的資料內容, 如果是一個函式指標變數的話, 可以直接調整函式的處理方法)

```
void qsort(void* base, size_t num, size_t size,  
          int (*compare)(const void*, const void*));
```

在運用 qsort 的時候需要撰寫函式 compare 來協助 qsort 分辨任意兩筆資料的大小 (回傳負值代表第一比資料比較小, 回傳 0 代表相等, 回傳正值代表第二筆資料比較小), qsort 在執行的過程中會不斷地呼叫這個函式來判斷兩筆資料的大小

3. **為什麼不直接寫一個比較兩個整數變數大小的函式 int compare(int \*, int \*) 或是 int compare(int, int) 呢?**

請注意這個 compare() 函式不是給你自己的程式使用的, 是給 qsort() 使用的, qsort() 是 stdlib 函式庫裡面別人在許多年前幫你寫好的一個工具, 在設計的時候當然完全不知道你想用它來排序的每一筆資料到底是 float, int, double 還是其它的型態, 也不曉得每一筆資

料用了幾個位元組, 所以只能夠運用 void \*element1, void \*element2 這兩個沒有特別型態的記憶體位址變數來傳入想要比對的兩筆資料 (qsort 演算法可以排序的不限於整數型態的資料), 來呼叫 compare 函式

我們在運用 qsort 時當然知道是要比對整數還是浮點數還是其它的型態, 但是 compare 函式需要根據 qsort 的要求來寫, 也就是第四個參數是上面的 int (\*compare)(const void \*, const void \*) 的函式指標

那麼在撰寫 compare 這個函式內容的時候, 我們知道要比對的兩筆資料是什麼型態, 所以就可以強制由 const void \* 型態轉換為你知道的型態, 例如上例的 int \*, 然後再進行比對

這是一個很有趣的程式重用機制, 其實就是一種多型的機制, qsort() 函式裡面在處理資料的時候其實完全不管實際想要排序的資料型態是什麼, 只知道每一個單位所佔用的記憶體是一樣的, 只知道運用 qsort 的人會提供給他判斷兩個單位誰大誰小的函式 compare() 就夠了, 在 C 程式裡面這種多型的運用不算太多, 但是到這學期後面你會看到 C++ 裡面的最常運用的多型機制就長這個樣子

[qsort 的各種應用方法 \(視訊\)](#)



[回 C++ 物件導向程式設計課程 首頁](#)

最近更新日期: Sat Mar 03 2018 21:54:56 GMT+0800 (台北標準時間)

製作日期: 02/20/2013 by 丁培毅 (Pei-yih Ting)

E-mail: [pyting@mail.ntou.edu.tw](mailto:pyting@mail.ntou.edu.tw) TEL: 02 24622192x6615

[海洋大學](#) [電機資訊學院](#) 資訊工程學系 Lagoon