4. function pointer

變數的指標指向變數的位址，同樣的，function pointer (函式指標) 也是指向函式的位址的指標。  
函式指標的加入，讓C/C++的符號更複雜，也使更多人望之而卻步。在說明函式指標的用途前，我們先直接由語法來看看函式指標該怎麼宣告、怎麼理解。  
假設有個函式長成下面的樣子：

void func1(int int1, char char1);

我們想宣告一個能指向func1的指標，則寫成下面這樣：

void (\*funcPtr1)(int, char);

這樣的寫法應理解成：funcPtr1是一個函數指標，它指向的函數接受int與char兩個參數並回傳void。如果今天有另一個函式長成

void func2(int int2, char char2);

則funcPtr1也能指向func2。  
指標指向的方法，寫成下面這樣：

funcPtr1 = &func1;

取址符號省略亦可，效果相同：

funcPtr1 = func1;

若欲在宣告時就直接給予初值，則寫成下面這樣：

void (\*funcPtr1)(int, char) = &func1; //&亦可省略

stdlib.h中提供的qsort函式是函式指標最常見的應用之一。此函式之prototype長得如下：

void qsort(void\* base, size\_t n, size\_t size, int (\*cmp)(const void\*, const void\*));

其中的int (\*cmp)(const void\*, const void\*) 就使用到函式指標。  
  
函式指標常見的使用時機是multithread時。函數指標負責把函數傳進建立執行緒的API中。  
另外，callback function也是常使用函式指標的地方。所謂callback function即：發生某事件時，自動執行某些動作。在event driven的環境中，便時常使用callback function來實現此機制。  
事實上，函式指標還能讓C語言實作polymorphism。但礙於篇幅，在此不再詳述。