

第 11 章

指標

敬告！這個章節包含了進階的觀念並且會處理到可能會嚇到初學者的底層 C 語言觀念。如果您現在完全無法理解它，不用擔心。感謝上蒼，一般而言 - 雖然了解指標 (pointer) 如何運作很有幫助 - 它對於開始用 Objective-C 來進行程式設計並不是必要的。

當您定義一個變數後，您的 Mac 就會把這個變數與它記憶體內的某些空間連結起來，以儲存這個變數的值。

舉例來說，請查驗以下的述句：

[1]

```
int x = 4;
```

為了要執行它，您的 Mac 會在它的記憶體中找出一些還沒有被使用的空間，然後標明這塊空間是變數 x 的值所儲存的地方（當然，我們在這邊可以也應該使用其他的變數名稱）。請再看一次指令 [1]。標明這個變數的型別（在這裡是 *int*）以讓您的電腦知道在記憶體中需要多少的空間來儲存 x 的值，而如果這個值的型別是 *long long* 或 *double*，那麼需要保留的記憶體就更多了。

這個指定指令 " $x = 4$ " 把數字 4 儲存在這個保留的空間中。當然，您的電腦記得這個名為 x 的變數的值儲存在它記憶體的何處，或者換句話說，它知道 x 的位址 (address) 為何。如此一來，每次您在您的程式中使用 x 的時候，您的電腦就可以查看正確的地方（在正確的位址）然後找到 x 實際的值。

一個指標 (pointer) 是一個含有其他變數的位址的變數。

給定一個變數，您可以藉由在變數前面寫個 $&$ 來取得它的位址。舉例來說，要取得 x 的位址，您要寫 $&x$ 。

當電腦對算式 (expression) x 求值 (evaluate) 的時候它會傳回變數 x 的值（在我們的例子中，它會傳回 4）。相對的，當電腦對算式 $&x$ 求值時，它會傳回變數 x 的位址，而非儲存在此的值。位址是一個表示電腦記憶體中特定位置的數字（就像房間號碼表示在旅館中特定的一間房間一樣）。

您可以像下面這樣宣告一個指標：

[2]

```
int *y;
```

這個述句定義了一個名為 y 的變數，它會含有一個型別為 *int* 的變數的位址。要在變數 y 中儲存變數 y 的位址（正式的怪胎話：指定 x 的位址給 y ），您要這樣作：

[3]

```
y = &x;
```

給一個指標，您可以在這個指標前寫上星號來取得它所指向的變數。例如，對這個算式求值

```
*y
```

會傳回 4。這與對算式 "x" 求值相等。執行這個指令

```
*y = 5
```

與執行指令 "x = 5" 是相等的。

指標是有用的，因為有時候您不要處理變數的值，函式要處理這個變數的位址。舉例來說，如果您要設計一個函式來對變數加 1，您需要傳送給它這個變數的位址。這是因為您要修改儲存在變數中的東西，而不是單純的使用目前的值。因此，您使用一個指標作為引數：

[4]

```
void increment(int *y)
{
    *y = *y + 1;
}
```

然後您可以像這樣呼叫它：

[5]

```
int x = 4;
increment(&x);
// 現在 x 等於 5 了
```