

# 由片語學習C程式設計

台灣大學資訊工程系劉邦鋒著

台灣大學劉邦鋒老師講授

August 19, 2016

## 第二單元

# 基本架構

# 程式骨架

## 片語 1: 程式骨架

```
1 main()  
2 {  
3 }
```

- **main** 主程式。將來程式會由這裏開始執行。
- **main** 後面有一組小括號 **()**，小括號中間沒有任何東西。代表 **main** 不需要任何參數。
- 一組大括號 **{ }**。左大括號 **{** 代表主程式 **main** 的開始。右大括號 **}** 代表 **main** 的結束。

# 要點

## 特殊字元

小括號 ( 及 ) 代表一個函式的參數的開始和結束。

## 特殊字元

大括號 { 及 } 代表一個函式開始和結束。

# 變數

## 片語 2: 整數變數宣告

```
1 int i;
```

- 在 C 程式語言 中變數須經過宣告才能使用。
- 你可以想像這些變數是存放資料的紙片，我們可以在裡面存放以後要使用的資料。

# 變數屬性

## ① 類別

一個變數的類別就是可以放進這個變數的資料類別，例如整數類別是 `int`。

## ② 名字

一個變數必須有一個名字。變數的名字可以用大小寫英文字母，數字，以及底線 `_` 組成，但是不能由數字開頭。同時大小寫字母會被視為不同。

## ③ 值

一個變數可以存放符合其類別的資料，而現在這個變數所存放的資料就稱為它的值。

## ④ 位址

一個變數是存放在計算機的記憶體裏，而計算機的記憶體是由位址來存取，所以一個變數就有一個相對應的記憶體位址。

## 學習要點

C 程式語言 的變數必須先宣告再使用。

## 學習要點

C 程式語言 的句子，例如宣告等，必須使用一個分號；作為這個句子的結束。

## 特殊字元

底線 `_` 可以用作變數名稱的一部分，方便將多個英文字組合成變數名稱。

## 特殊字元

分號 `;` 代表程式句子的結束。



# 設初始值

## 片語 3: 整數變數宣告且設初始值

```
1 int i = 0;
```

- 宣告一個整數變數，同時設置初始值，所以變數 *i* 的值一開始即設為 0。
- 如果未設置初始值，則變數的初值可能是任意整數。

# 要點

## 學習要點

一個變數如果沒有給定初始值，則變數的值可以是任意值，這是很多程式錯誤的原因。

## 片語 4: 多個整數變數宣告

```
1 int i, j, k;
```

- 一次宣告數個整數變數。

## 風格要點

相關的變數宣告在一行，這樣程式會比較具有可讀性。

# 輸出

## 片語 5: 輸出一整數

```
1 printf("%d", i);
```

- 以上程式會將變數 `i` 的值顯示在螢幕上。
- `printf` 是一個會輸出資料到螢幕的函數。
- `printf` 的 **參數** 用小括號組 `()` 包起來。
  - 參數 `"%d"` 中的百分號 `%` 代表 `printf` 要印的資料，也就是後面的 `i`。
  - `%d` 中的 `d` 是以十進位 (decimal) 的形式印出變數值。
  - 後面的參數是 `i`。意即 `printf` 會顯示 `i` 這個變數的數值。

# 要點

## 特殊字元

雙引號 " 代表一個字串的開始與結束。

## 特殊字元

百分號 % 代表 `printf` 要處理的參數。

## 特殊字元

逗號 , 用來分隔一個函式的的衆多參數。

## stdio.h

## 片語 6: 引入 stdio.h 標頭檔

```
1 #include <stdio.h>
```

- 將系統的標準輸出入程式庫 (standard I/O library) 的標頭檔 `stdio.h` 加進你的程式中。
- 程式中如果有使用輸出入的函數，例如 `printf`，就需要把 `stdio.h` 加進程式中。
- 必須出現在程式的一開始。

## stdio.h

## 片語 7: 有引入stdio.h的主程式

```
1 #include <stdio.h>
2 main()
3 {
4 }
```



# 設定並顯示變數

## 範例程式 8: (init-print.c) 設定並顯示變數 i 的值

```
1 #include <stdio.h>
2 main()
3 {
4     int i = 100;
5     printf("%d", i);
6 }
```

## 輸出

1	100
---	-----

# 要點

## 學習要點

程式的執行是由上往下，一行一行執行。

## 風格要點

主程式 `main` 的句子都向右空兩格，這樣比較容易看出 `main` 主程式的起點與終點。

# 顯示兩個變數的值

## 範例程式 9: (init-two-print.c) 顯示兩個變數的值

```
1 #include <stdio.h>
2 main()
3 {
4     int i = 100;
5     int j = 200;
6     printf("%d", i);
7     printf("%d", j);
8 }
```

## 輸出

1 100200

# 換行

## 片語 10: 以十進位輸出一整數並換行

```
1 printf("%d\n", i);
```

- 為了解決顯示完 100 後馬上顯示 200 的問題，我們可以使用一個**換行字元**，強迫程式在顯示完 *i* 的結果後，換下一行才顯示 *j* 的結果後。
- 作法是在顯示 *i* 的結果時，在 `%d` 後面加上 `\n` 這個換行字元。

# 要點

## 特殊字元

`\n` 會強迫 `printf` 換行。

# 換行

範例程式 11: (init-two-print-newline.c) 將變數的值顯示在螢幕上並換行

```
1 #include <stdio.h>
2 main()
3 {
4     int i = 100;
5     int j = 200;
6     printf("%d\n", i);
7     printf("%d\n", j);
8 }
```



## 輸出

1	100
2	200

# 輸入

## 片語 12: 輸入一整數

```
1 scanf("%d", &i);
```

- 從鍵盤讀取一個十進位整數，並將其設定為變數 `i` 的值。
- `scanf` 和 `printf` 類似，都使用第一個參數 `"%d"` 表示處理的是一個十進位整數。
- 第二個參數 `&i` 把從鍵盤讀到的十進位整數指定給 `i`。既然要指定給 `i`，我們就必須知道 `i` 在記憶體中的位址。在 C 程式語言中，變數 `i` 的位址是用 `&i` 來表示。

# 要點

## 學習要點

使用 `scanf` 從鍵盤讀資料設定變數時，要使用 `&` 表示變數的位置。

## 特殊字元

`&` 代表變數的位址。

## 範例程式 13: (scan-print.c) 從鍵盤讀入一整數再將其顯示在螢幕上

```
1 #include <stdio.h>
2 main()
3 {
4     int i;
5     scanf("%d", &i);
6     printf("%d\n", i);
7 }
```

輸入

1 100

輸出

1 100