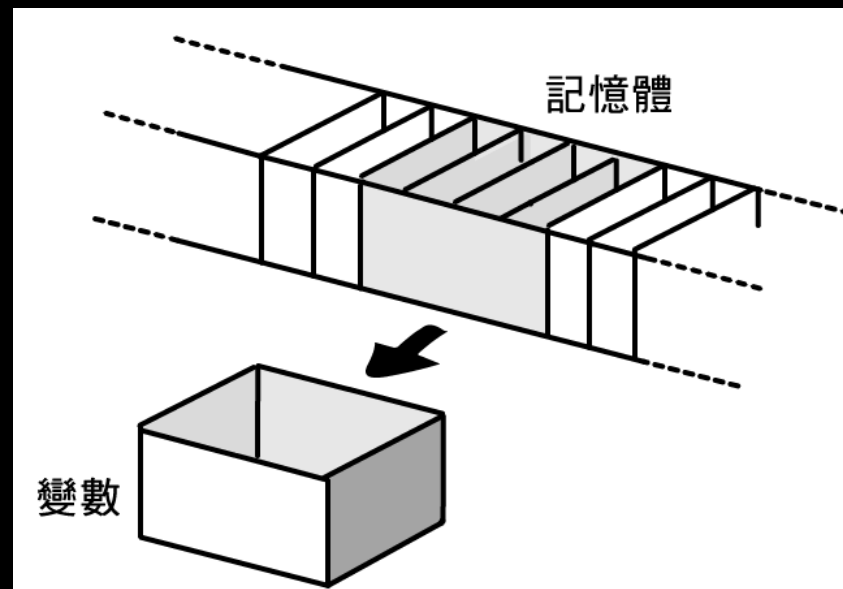


變數

3-1 變數

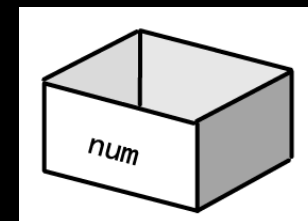
○ 關於變數

- 電腦在進行處理的過程中，也必須「記住」某些特定的資料，才能進行後續處理(例如進一步將結果顯示在螢幕上)，這種讓電腦記住特定資料的功能就稱為變數(variable)。
- 變數通常是存放在記憶體(memory)內。



3-2 識別字

- 在C語言程式中如果要使用某變數，必須先針對該變數進行下列2項設定工作：
 - 指定變數的「名稱」
 - 指定變數的「型態」
- 識別字專門用來做為變數的「名稱」。
- 建立新的識別字時仍然要遵照下列的規則：
 - 一般而言，我們會使用英文字母、阿拉伯數字、底線連接（_）、\$等符號作為識別字名稱。無法包含特殊記號。
 - 視環境而定，最長不得超過31個字元。
 - C語言已保留起來的關鍵字（keyword）不能做為識別字名稱。常見的關鍵字如return。
 - 識別字名稱不能用阿拉伯數字做為開頭。
 - 英文字母的大小寫有區別。



3-3 變數的資料型態

- 變數可以用來「存放」特定的文字或數值，這些文字或數值一般稱為資料型態（data type）。
- C語言的變數可以使用的基本資料型態：
 - 文字型--char、unsigned char
 - 整數型--short int、int、
unsigned int、long int
 - 浮點數型--float、double、
long double
- 電腦中的每個二進位數值都算是一個bit(位元)，它是電腦資料的基本單位，而八個bit可以組成一個byte。

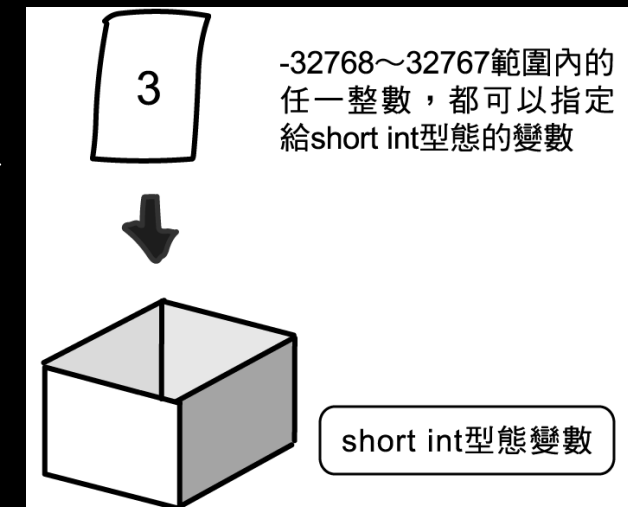


表 3-1：C 語言的變數可以使用的基本資料型態

種類	名稱	檔案大小	資料的範圍
文字型	char	1 byte	1 個英數字 -128 ~ 127
	unsigned char	1 byte	1 個英數字（無符號） 0 ~ 255
整數型	short int	2 byte	整數 -32768 ~ 32767
	unsigned short int	2 byte	整數（無符號） 0 ~ 65535
	int	4 byte	整數 -2147483648 ~ 2147483647
	unsigned int	4 byte	整數（無符號） 0 ~ 4294967295
	long int	4 byte	長整數 -2147483648 ~ 2147483647
	unsigned long int	4 byte	長整數（無符號） 0 ~ 4294967295
浮點數型	float	4 byte	單精度浮點數 3.4E-38 ~ 3.4E+38
	double	8 byte	雙倍精度浮點數 1.7E-308 ~ 1.7E+308
	long double	8 byte	四倍精度浮點數 1.7E-308 ~ 1.7E+308

3-4 宣告變數

- 宣告變數的語法：

變數的資料型態 識別字；

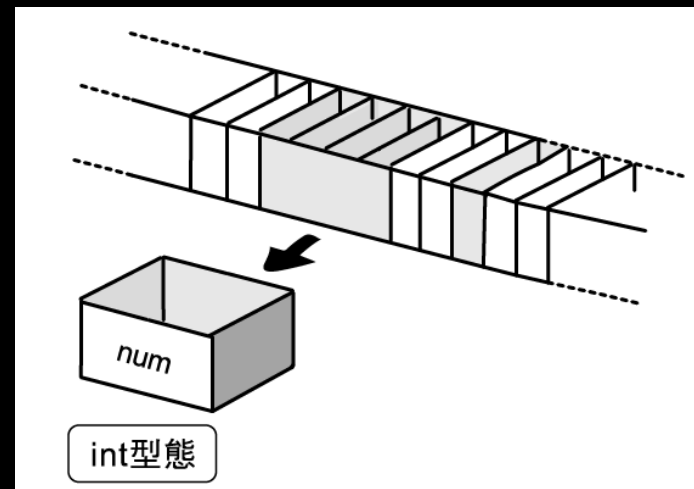
- 範例：

`int num;` ← **int**型態的變數 (變數名稱是**num**)

`char c;` ← **char**型態的變數 (變數名稱是**c**)

`double db, dd;`

← **double**型態的變數
(變數名稱是**db**、**dd**)



3-5 開始使用變數

- 指定「數值」或「其他資料」給某變數的語法為：

變數名稱 = 特定資料;

ex: **num = 3;**

- 等號(=)的意思並非是「等於」，而是「指定給」。

- 輸出變數所代表的值：

printf("變數num的值為%d。\\n", num);

- 變數的初始化(initialization)語法為：

資料型態 識別字 = 運算式;

ex: **int num = 3;**

- 初始化就是在宣告變數的同時指定值。

Sample1.c ▶ 使用變數

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

```
int main(void)
```

```
{
```

```
    int num; ●
```

①宣告變數 num

```
    num = 3; ●
```

②將 3 指定給變數 num

```
    printf(" 變數 num 的值是 %d 。 \n", num);
```

```
    return 0;
```

③輸出變數 num 代表的值，也就是 3

```
}
```

- 改變變數的值之範例：

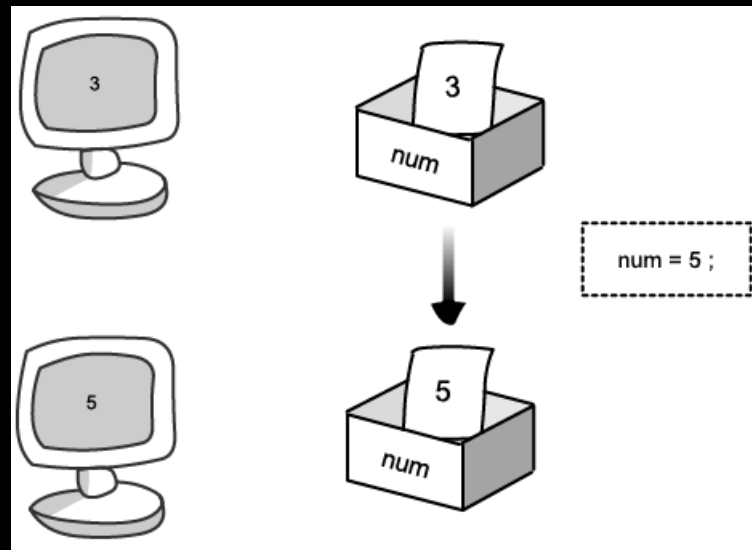
...

```
num = 3;  
printf("變數num的值是%d 。 \n",  
      num);  
num = 5;  
printf("變數num更新後的值是%d 。  
      \n", num);
```

- 執行結果：

變數num的值是3。

變數num更新後的值是5。



Sample2.c ► 改變變數的值

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

```
int main(void)
```

```
{
```

```
    int num;
```

```
    num = 3;
```

```
    printf("變數 num 的值是 %d 。 \n", num);
```

①輸出變數的值

```
    num = 5;
```

②重新指定變數的值

```
    printf("更新變數 num 的值。 \n");
```

```
    printf("變數 num 更新後的值是 %d 。 \n", num);
```

③輸出新的變數值

```
    return 0;
```

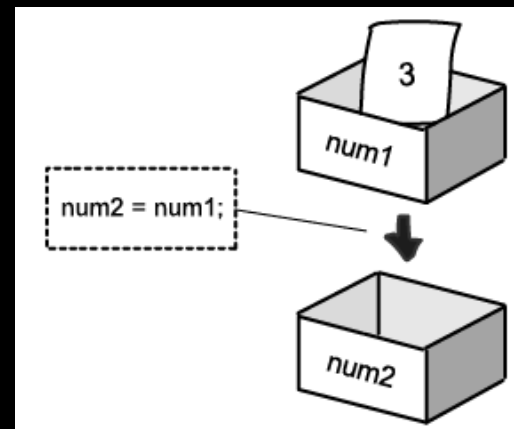
```
}
```

○ 變數值再指定

- 指定各種資料給變數、作為變數所代表的值，此時等號（=）右邊並不是只能使用3或5這種數字型的資料。
- 例如：

```
...  
int num1, num2;  
num1 = 3;  
num2 = num1;  
printf("變數num2的值是%d。\\n",  
      num2);
```

結果num2的值會是3，這表示既有的變數值也可以指定給新的變數。



Sample3.c ▶ 變數值再指定

```
#include <stdio.h>

int main(void)
{
    int num1, num2;

    num1 = 3;

    printf("變數 num1 的值是 %d 。 \n", num1);

    num2 = num1; ●————— 變數 num1 的值再指定給變數 num2

    printf("把變數 num1 指定到變數 num2 之中。 \n");

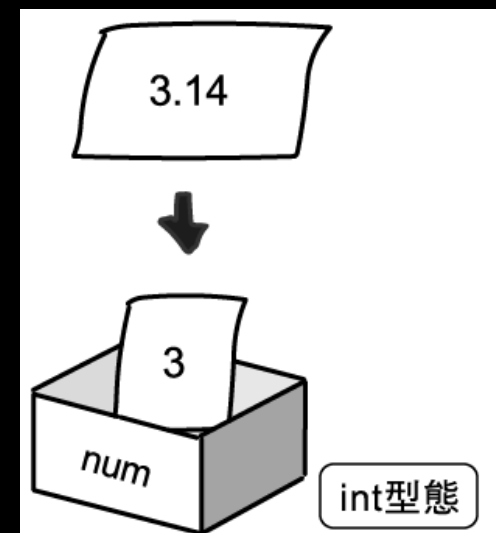
    printf("變數 num2 的值是 %d 。 \n", num2);

    return 0;
}
```

- 指定變數值時應注意的事項

- 變數會根據宣告的型態來決定可以記憶的數值種類。如果將帶有小數的數值指定給整數值的變數，變數的資料型態就會自動被轉換。

```
int num;  
num = 3.14;  
printf("變數num的值是%d 。  
\\n", num);
```



Sample4.c ▶ 指定變數的值

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

```
int main(void)
```

```
{
```

```
    int num;
```

```
    num = 3.14;●
```

將小數指定給 int 型態的變數...

```
    printf(" 變數 num 的值是 %d 。 \n", num);
```

```
    return 0;
```

```
}
```

- 留意變數的宣告位置

原則上變數都是在main()函數的程式區塊（block）內宣告，在C語言當中，變數的宣告必須放在區塊內的最前頭。

```
...
```

```
int main(void)
```

```
{
```

```
    int num;
```

```
    printf("歡迎來到C語言！\n");
```

```
    ...
```

```
}
```

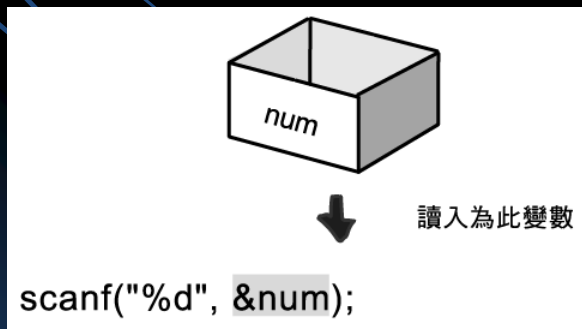
宣告變數的部分

3-6 從鍵盤輸入變數值

- 接受使用者從鍵盤輸入變數值

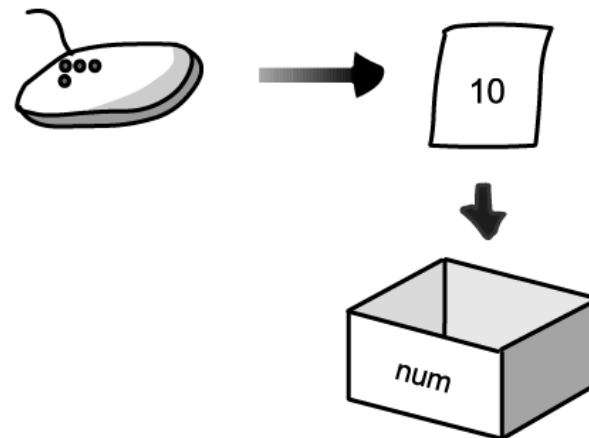
- 使用scanf便可讀入從鍵盤輸入的數值，其範例如下：

```
#include <stdio.h>
int main(void)
{
    宣告變數;
    scanf("轉換規格", &變數);
    ...
}
```



當程式處理到scanf時，會呈現等待由鍵盤輸入的狀態

輸入完畢後按下Enter鍵便會讀入為變數



Sample5.c ▶ 從鍵盤輸入整數

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

```
int main(void)
```

```
{
```

```
    int num;
```

```
    printf(" 請輸入 1 個整數 。 \n");●
```

顯示請使用者利用鍵盤輸入的訊息

```
    scanf("%d", &num);●
```

從鍵盤輸入的數值被讀入變數 num

```
    printf(" 輸入的是 %d 。 \n", num);
```

顯示已輸入的數值

```
    return 0;
```

```
}
```

○ 輸入小數

- 使用「%lf」這個指令來當做轉換規格，範例如下：

```
scanf("%lf", &num);
```

○ 輸入文字

- 輸入文字所使用的是getchar，其語法如下：

```
變數名稱 = getchar();
```

○ 標準輸入與標準輸出

- 正如同printf代表的是「標準輸出」一樣；scanf和getchar所指的就是從標準輸入（standard input）這個地方來進行讀入數值和文字的處理。通常所謂的「標準輸入」指的就是電腦的鍵盤。

Sample6.c ▶ 輸入小數

```
#include <stdio.h>

int main(void)
{
    double num;

    printf(" 請輸入小數。 \n");

    scanf("%lf", &num);

    printf("%f 已輸入。 \n", num);

    return 0;
}
```

從鍵盤讀入小數

Sample7.c ▶ 輸入文字

```
#include <stdio.h>

int main(void)
{
    char ch;

    printf(" 請輸入 1 個英數字。 \n");

    ch = getchar(); ●———— 從鍵盤讀入文字

    printf(" 輸入的是 %c。 \n", ch);

    return 0;
}
```