



字元字串與陣列

**ABCDEFGH
IJKLMNOP
QRSTUVWXYZ
1234567890
abcdefgh
hijklmnop
qrstuvwxyz
!&?/=+-.@**

字元與字串

- 字元：
一個英文字母、數字或
其他的符號
- 字串：
一段文字



C語言如何處理字元
字串？

陣列(array)



資料型態的簡單分類

資料型態	型別	輸出格式指定碼
整數	int	%d
浮點數	float	%f
字元	char	%c
字串		%s



字元的輸入與輸出

- 輸入或指定

- 當指定一個字元時，我們可以用一對單引號 ' 把該表字元夾起來，例如：

```
char a='a' ;
```

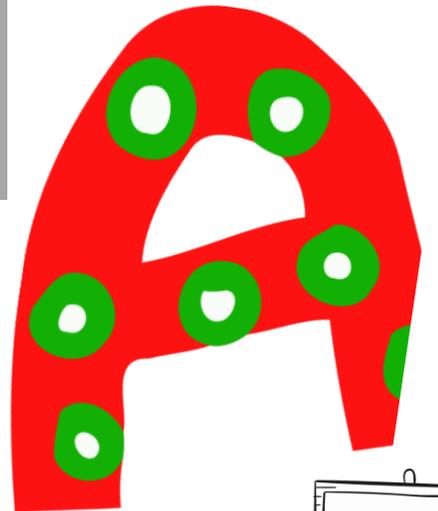
- 以下則是輸入一個字元

```
scanf( "%c" ,&a)
```

- 輸出

- `printf` 讀入及印出一個字元則是使用 `%c`

```
#include <stdio.h>
int main(){
    char a;
    a= 'a' ;
    printf( "%c\n" ,a);
    return 0;
}
```



Hello!

- 指定一個字串變數資料
`char char1[255]= "Hello"`
- 輸出如下：
`printf("%s\n", char1);`

字串的輸入與輸出



陣列中的字串-關於長度的宣告

- 上頁的範例：

- char

- char1[255]= "Hello"

- printf("%s\n", char1);

- 字串長度

- 上方的例子宣告了 char 1 是一個字串，長度有 255 個字元。

- 我們在宣告字串的時候要注意它的長度，以免為 255 位數不夠造成程式錯誤。

- 到底要多長？



陣列中的字串-關於字串的**內容**

- 上頁的範例：
 - `char char1[255]= "Hello"`
 - `printf("%s\n", char1);`

• \0

- 每個字串後面都有一個 '\0' 的字元，
- "Hello" 字串，共用了 6 Bytes

H	e	l	l	o	\0
---	---	---	---	---	----



陣列中的字串-關於字串的輸入

- 字串輸入範例：
 - `char char2[255];`
 - `scanf("%s" , char2)`
 - `printf("%s\n", char2);`

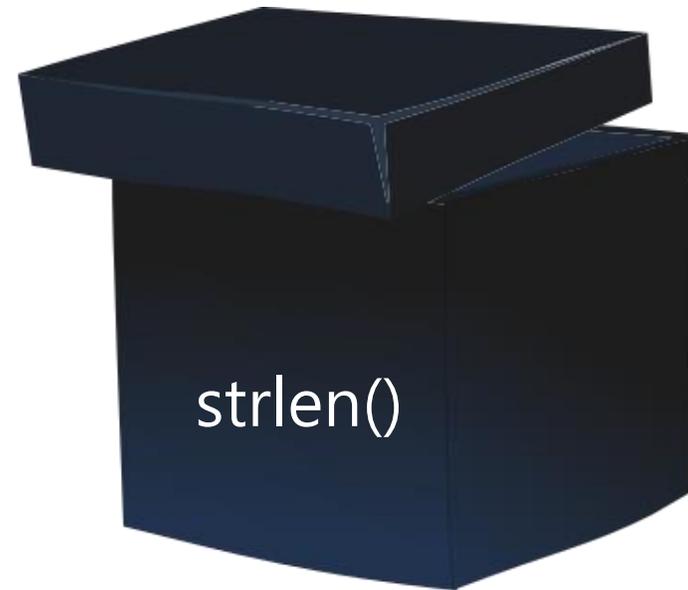
輸入不帶&

- 由於 `char2` 本身就是這個字元陣列的位址，所以在 `scanf` 裡我們不需要加上 `&` 符號。



試著使用字串函數

- 輸入一個字串，然後計算字串的長度？
- 不用函數怎麼寫？
- 有函數可以計算字串長度喔，如下圖提示



一些字串函式

函式格式	函式功能
strcat(字串名1, 字串名2)	將字串2連接到字串1後邊，返回字串1的值
strncat(字串名1, 字串名2, 長度n)	將字串2前n個字串連接到字串1後邊，返回字串1的值
strcpy(字串名1, 字串名2)	將字串2複製到字串1，返回字串1的值
strncpy(字串名1, 字串名2, 長度n)	將字串2前n個字串複製到字串1，返回字串1的值
strcmp(字串名1, 字串名2,)	比較字串1和字串2的大小，比較的結果由函數帶回； 如果字串1>字串2，返回一個正整數 如果字串1=字串2，返回0 如果字串1<字串2，返回一個負整數
strncmp(字串名1, 字串名2, 長度n)	比較字串1和字串2的前n個字串進行比較，函數返回值的情況同strcmp函數
strlen(字串名)	計算字串的長度，終止符'\0'不算在長度之內
strlwr(字串名)	將字串中大寫字母換成小寫字母 註：只能在 windows (VC、MinGW等) 環境下呼叫
strupr(字串名)	將字串中小寫字母換成大寫字母 註：只能在 windows (VC、MinGW等) 環境下呼叫

