

# 第七週

王子銓, 陳毅軒, 吳尚龍

電機通訊程式設計

April 1, 2024

## 1 字串

## 2 連假加分作業預告

## 什麼是字串

- 用來儲存多個字元的資料型態。
- 是一種陣列。
- 由字元組成的陣列。
- 字串的名字是一個指標，指向 index 為 0 的位置，並且不可更改。

# 字串

	0	1	2	3	4	5	6	7	8	9	10	11	12
str1	H	e	l	l	o		w	o	r	l	d	!	\0

注意: 字元陣列結尾是 \0

# 字串宣告

字串由字元陣列組成。

```
1 // 語法
2 char str1[5];
3 char str2[] = "C is better";
4 char str4[3244] = {'e','x','c','e','e','d'};
```

**注意:** 字串宣告必須是雙引號

# 字串輸出

```
1 // 語法
```

```
2 char str1[] = "Hello World";  
3 printf("%s\n", "ccuee ccucomm");  
4 printf("%s", str1);
```

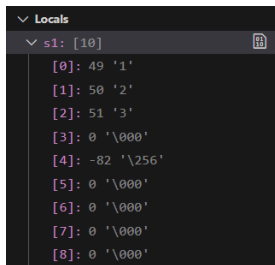
```
1 // output
```

```
2 ccuee ccucomm  
3 Hello World
```

# 字串輸入

```
1 // 語法
2 char str[10];
3 scanf("%s",str);
```

程式會以空白、換行中斷輸入，並把輸入結尾的部分替換成 `\0`  
例如這裡輸入 123，會把 `str[3]` 的地方替換成 `\0`。



**注意:** 如果輸入超過宣告的字串長度，會發生 **run error**

# 字串函式庫

```
1 // 語法
```

```
2 #include <string.h>
```

```
1 char str1[20], str2[10];
```

```
2 strcpy(str1, str2); // 將 str2 的內容複製到 str1
```

```
3 strncpy(str1, str2, 10); // 將 str2 的內容複製到 str1，且限制長度 10
```

```
4 strcat(str1, str2); // 將 str2 的內容接在到 str1 後面
```

```
5 strncat(str1, str2, 5); // 將 str2 的前 5 個字元接在到 str1 後面
```

```
6 strlen(str1); // 回傳 str1 字串的長度
```

**注意:** 如果以上字串函式沒有指定長度，會讀到 `\0` 為止，因此可能會超過 `s1` 陣列長度，會建議配合 `strlen` 來指定長度。



```
1 // 語法
2 char str1[20], str2[10];
3 strcmp(str1, str2);    // 比較 str1 和 str2
```

## 依照字典序逐個比較

- 若 str1 字典序小於 str2，則回傳  $< 0$  的值。
- 若 str1 字典序大於 str2，則回傳  $> 0$  的值。
- 若 str1 和 str2 相等，則回傳 0。

```
1 // 語法
2 char str1[20], str2[10];
3 strncmp(str1, str2, 5);           // 比較 str1 和 str2 的前五個字元
```

## 依照字典序逐個比較

- 若 str1 字典序小於 str2，則回傳  $< 0$  的值。
- 若 str1 字典序大於 str2，則回傳  $> 0$  的值。
- 若 str1 和 str2 相等，則回傳 0。
- 與 strcmp 不同，strncmp 會限定比較的字串長度。

```
1 // 語法
2 char str[10]="abc";
3 strlen(str); // 回傳 str 字串的長度
```

**注意:** 使用陣列時，要初始化，避免出現預期外的結果

## strcspn

- `size_t strcspn(const char *str1, const char *str2)`
- `str1` 為要被檢查的字串
- `str2` 為該字串包含了要在 `str1` 中進行匹配的字元列表
- 回傳 `str1` 開頭連續都不含字串 `str2` 中字元的字元數量

```
1 // 語法
2 char str1[20];
3 size_t index = strcspn(str1, "a");
4 //回傳 str1 開頭連續都不含"a" 字符的字元數量，也就是第一個 "a" 的 index
```

# 加分作業預告

連假會出幾題額外的加分作業，約 3 到 4 題，每題加總成績一分，敬請期待 (這不是愚人節玩笑)