

C++ builder 字串轉換

C++ builder 裡的字串是使用 `AnsiString` 類型，如果要和 C 的字串溝通，需要用 `c_str()` 將 `AnsiString` 轉成 C 的 `char*`，再用 `strcpy` 讓目前指標的內容賦予給目的指標，
ex. `#include <string>` 現在看 `strcpy` 範例。

Input :

Output: "test"

`strcpy` 是將第二參數
給第一參數。

1. `#include <string>`
⋮
2. `char ch1[20];`
3. `char ch2[20] = "test";`
4. `strcpy(ch1, ch2);`
5. `std::cout << ch1 << std::endl;`

第一和第二參數都
為 `char` 類型的
指標。

接著說明將 `String` or `AnsiString` 轉成 C 的 `char` (字元) 形式，使用 `<cstring>` 裡的 `c_str()`，它會回傳 `const char*`。

注意，呼叫 `c_str()` 是回傳只能讀的字元類型的指標。

ex.

Input :

Output = "test";

1. #include <cstring>

⋮

2. AnsiString ansiStr = "test";

3. char *ch;

4. ch = ansiStr.c_str();

5. cout << ch << endl;

如果在這行加

ansiStr = "123"; ,

cout 出來 ch 為 123。

上面第 4 項這樣讓 ansiStr 的指標給 ch 是不安全的，因為只要 ansiStr 一改內容，cout 出來的 ch 也會跟著改，所以為了 ch 的內容不會改變，可將

ch = ansiStr.c_str();

改成

strcpy(ch, ansiStr.c_str());