2.1 輸出

使用 cout 輸出資料

在 Code::Blocks 裡建立一個專案後，它會自動產生這樣一個程式架構。

```
#include <iostream>

using namespace std;

int main()
{
    cout << "Hello world!" << endl;
    return 0;
}
```

main function

其中的 `main()` 稱為主函數(main function)，它是程式的起點。程式在啟動後，會由 main 裡的第一行開始依序執行下去。

`return 0;` 表示帶著回傳值 0，返回呼叫 main() 的地方，以這個例子來說就是返回作業系統。

```
int main()
{
    // 要做的事情寫在這裡面
    return 0;
}
```

剩下的那行就是真正在做的事情，程式執行後會在畫面上看到。



單行註解：雙斜線 // 開始到該行結尾都屬於「註解」，用來對程式碼做說明。這是給人類看的，對執行完全沒影響。

標準輸出

`cout` 是用來將資料輸出到標準輸出(standard output / stdout)，而一般的標準輸出指的是螢幕。

我們可以把它想像成，資料沿著 `<<` 的方向流到螢幕那邊。



至於 `endl` 則是 end of line，即「換行」的意思。如果把程式修改一下。

```
cout << "Hello" << endl << "world!";
```

那麼輸出就會變這樣。

```
Hello
world!
```

> 提醒：程式碼內容有更動後，都要[save]->[build]，才能[run]。 因為 [build] 會根據你目前的程式碼做出新的執行檔。若沒這麼做，你執行的還是之前舊的執行檔。

字串 和 運算式

比較一下這兩行程式碼有什麼不同。

```
cout << "2+3" << endl;
cout << 2+3 << endl;
```

看起來都是 2+3，執行結果卻大不相同。

```
2+3
5
```

第一行輸出的是 `"2+3"`，前後有雙引號。雙引號框起來的內容都會被視為文字，整個被當作字串(string)來處理，所以你寫什麼樣子，它就輸出什麼樣子。

第二行輸出的是 `2+3`，前後**沒有**雙引號。在這種情況下，cin 會等待 `2+3` 這個運算式的結果被計算出來，再將其結果輸出。因此我們會看到第二行的輸出結果是 5。

練習

請推測以下程式執行後的輸出結果為何？

自己寫下結果後，再實際輸入這個程式，觀察執行結果。

```
#include <iostream>

using namespace std;

int main()
{
    cout << "Hello world!" << endl;
    cout << 1 + 2 << endl;
    cout << 1 + 2 * 3 – 4 << endl;
    cout << (1 + 2) * (3 – 4) << endl;
    cout << "(1 + 2) * 5 / 3 = " << (1 + 2) * 5 / 3 << endl;

    return 0;
}
```

Revision #44
★Created 2023-05-02 23:57:45 UTC by huihui
Updated 2024-03-04 12:08:42 UTC by huihui