

Expressions and Variables

0307, hydai
2015 資訊之芽 語法班

程式的架構

架構概念

- 程式 = 資料(變數) + 任務(函式)
- 函式 = 多行的動作(指令)
- 一切的起點 = 由主線任務(main)開始執行

基本規則

- C++ 程式碼結尾通常使用 .cpp
- ex: helloworld.cpp
- 每行指令的後面都由分號(;)
- 像是指令的句點

程式碼基本架構

```
1 #include <iostream>
2
3 int main() {
4     // Write your program here
5     return 0;
6 }
```

- 第一行先不理他
- 目前的主線任務是空的，只做回傳 0 的動作

hello world

```
1 #include <iostream>
2
3 int main() {
4     std::cout << "Hello, world" << std::endl;
5     return 0;
6 }
```

- 第一行先不理他
- 目前的主線任務多了一個指令：
 - 印出 Hello, world
 - cout 我們也先不理他

看見程式在說話
- cout

再看一次 hello world

```
1 #include <iostream>
2
3 int main() {
4     std::cout << "Hello, world" << std::endl;
5     return 0;
6 }
```

表示要輸出

輸出的字串

表示要換行

cout 的結構

- `std::cout << "String";`
 - 輸出 String
- `std::cout << "String" << std::endl;`
 - 輸出 String 然後換行

Example

- 0J - 202 / Hello, Sprout!
- 馬上來動手寫第一個程式吧！

數值運算

還記得小時候的數學

- 四則運算：+，-，*，/

- $3+4-2*6/3$

- 小括弧：()

- $(3+((4-2)*6)/3)$

- 取餘數：%

- $6 \% 3 = 0$

- $7 \% 2 = 1$

電腦的數學好像不太一樣？

- $22/7 = 3.142857.....$
- 我們馬上寫一個程式印出 $22/7$ 看看吧！

$$22/7 = ?$$

```
1 #include <iostream>
2
3 int main() {
4     std::cout << 22/7 << std::endl;
5     return 0;
6 }
```

- 把剛才 "hello, world" 改成 22/7 就可以囉
- `std::cout << 運算式;`
- `std::cout << 運算式 << std::endl;`

$$22/7 = 3.\dots?!!!$$

難道電腦錯了嗎？

- 電腦沒有錯，這裏就會牽涉到 屬性(型態)
- 在數學上我們會說整數、小數、有理數、無理數
- 程式裡也有：int, double, etc
- 預設的數字是 int 的型態
- 當整數遇到除法時，會自動幫你無條件捨位

還記得不等式嗎？

- 電腦世界裡常常需要邏輯判斷
- 存在著一些邏輯運算符號：
 - `==` 等於，`!=` 不等於
 - `>`，`>=`，`<`，`<=`
 - `&&` 且，`||` 或者
 - `!` 非

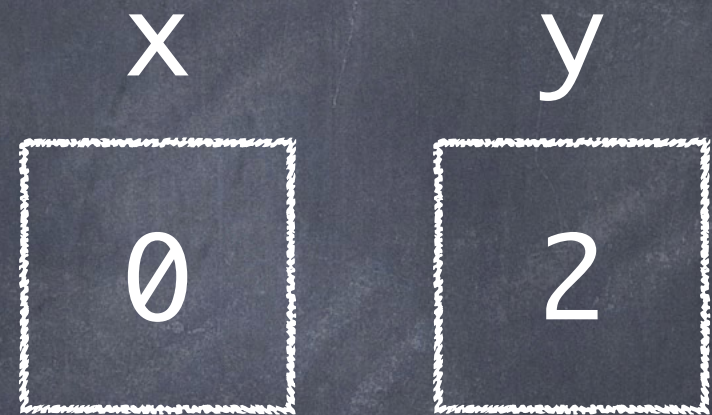
判斷的結果 (DEMO)

- 一個邏輯判斷結果就會是 True or False
- 在 C++ 裡面
 - True 通常回傳 1
 - False 通常是 0
- $(1 == 1) = 1$
- $(1 != 1) = 0$

變數

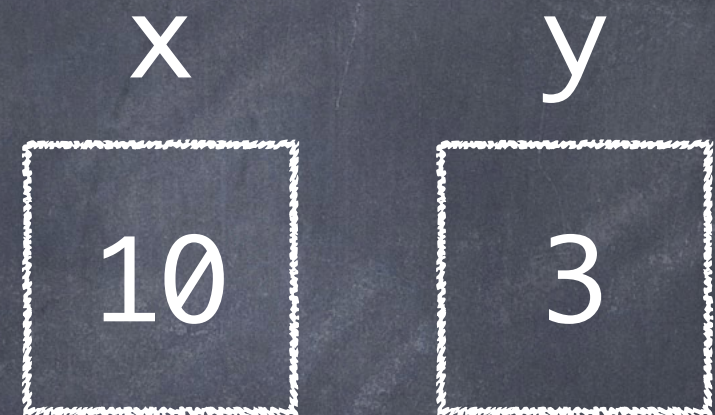
用來存資料的背包

- `int x = 0;`
- `int y = x + 2;`



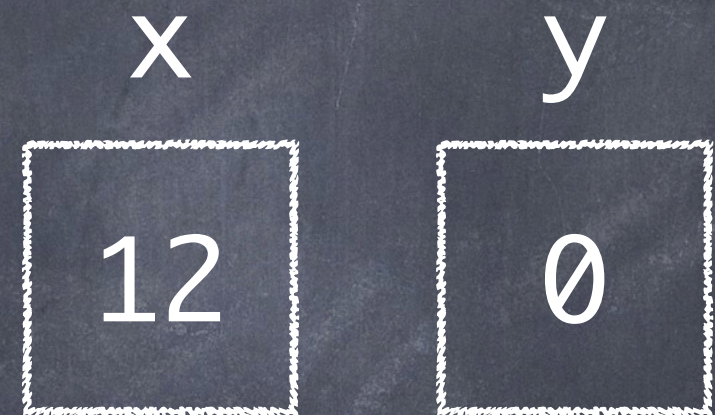
用來存資料的背包 (cont)

- `int x = 0;`
- `int y = x + 2;`
- `x = 10;`
- `y = 3;`



用來存資料的背包 (cont)

- $x = 10;$
- $y = 3;$
- $x = x + 2;$
- $y = y - 3;$



解釋 $y = y - 3$

- 這時候你應該會發現 "=" 並不代表等於
- 而是把右邊的數值放到左邊的變數中
- 嘗試看看 $y = y - 3$; 和 $y - 3$; 比較最後的 y 值

宣告變數的語法

- `int name; // 沒有初始值`
- `int score = 0; // 初始值為零`
- `int x = 3, y = 2, z = x + y;`
- `int` 是型態，還有：`bool`, `double`, `char`
- `name`, `score`, `x.....` 這些都只是一個名稱

要怎麼互動
- cin

讀取 a 輸出 a 的平方

```
1 #include <iostream>
2
3 int main() {
4     int a;
5     std::cin >> a;
6     std::cout << a*a << std::endl;
7     return 0;
8 }
```

表示要輸入

cin 的結構

- `std::cin >> 變數A;`
 - 把資料從鍵盤讀入到變數A裡面
- `std::cin >> 變數A >> 變數B;`
 - 讀取多個變數

Example

- OJ - 198 / 交換測試 - 2015
- 馬上測試一下自己的想法吧！

Example

- OJ - 200 / 閏年問題 - 2015
- 有時間就講解

分析結構

- `#include <iostream>`
 - 標準輸出入放在 `iostream` 裡面
 - 為了能利用這些功能，需要包含這個工具箱
- 之後隨著要用到更多的功能（函式）