

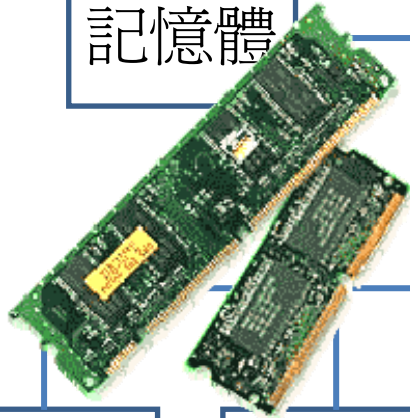
程式是什麼？

寫程式到底是怎一回事？



CPU 運作時儲存
資料與程式的裝置

記憶體



中央處理器

資料運算 $+$, $-$, $*$, $/$,
AND, OR, NOT

資料搬移
比對與控制移轉

光碟

硬碟

儲存大量資料的裝置



網路

資料傳遞



資料存放到記憶體/由記憶體取出

- **記憶體**是 CPU 不可缺少的夥伴, 沒有它 CPU 就像是重度失憶症患者, 剛做完的事、剛算出來的結果, 沒有一個記得 (其實也不知道該怎麼完成任何一件事, 所有的指令都不知道記在哪裡)

```
int x, y;
```

取名稱, 規劃存放內容 ----- **int**

```
x = 12;
```

存放資料 ----- **x =**

```
y = x + x / 3;
```

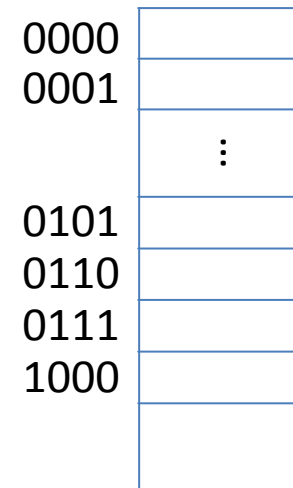
取出資料 ----- **x, y**

```
y = x - y;
```

```
int num = 35;
```

x, y, num 稱為**變數**

```
printf("我們班上有 %d 位同學\n", num);
```



要求 CPU 把資料顯示在螢幕上

```
#include <stdio.h>
```

```
printf("Hello World!\n");
```

```
printf("My name is Bob.\n");
```

```
q:\>testPrintf  
Hello World!  
My name is Bob.
```

```
int num = 35;
```

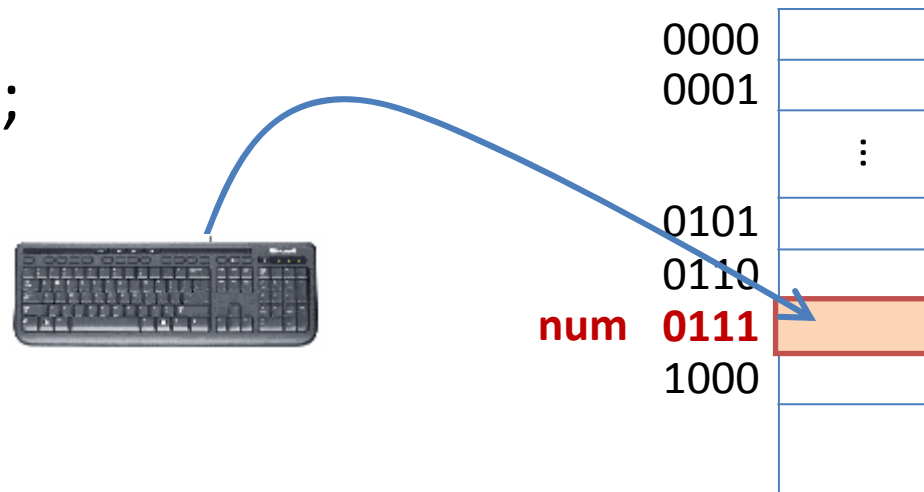
```
printf("我們班上有 %d 位同學\n", num);
```

```
我們班上有 35 位同學
```

要求 CPU 由鍵盤讀取資料

- 操作者由鍵盤輸入資料以後, CPU 把資料依照指定的格式放進記憶體 (變數) 裡

```
#include <stdio.h>  
  
int num;  
scanf("%d", &num);
```



CPU 的算術與邏輯運算

- +, -, *, /, AND, OR, NOT, XOR



```
int x = 12, y;
```

```
y = x + x / 3;
```

```
y = x - y;
```

```
double r, s = 12.5;
```

```
r = s * s;
```

CPU 的資料搬移

- 記憶體和記憶體之間, 記憶體和暫存器之間

```
int x = 12, y;
```

```
y = x + x / 3;
```

```
y = x - y;
```

- 記憶體和硬體裝置 (硬碟、鍵盤、網路...) 之間

測試比對與控制移轉

- CPU 沒有智慧, 但是這段程式有智慧行爲裡最基本的判斷

```
int x = 12, y;
```

```
y = x + x / 3;
```

```
y = x - y;
```

```
if (y < 0)
```

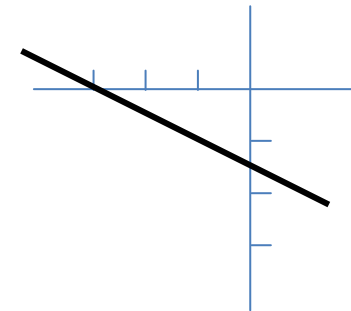
```
    printf("y 存放的數值是負數\n");
```

```
else
```

```
    printf("y 存放的數值大於或等於 0\n");
```

程式撰寫範例一

- 請寫一個程式計算 **直線** $ax + by + c = 0$ 和 **X 軸** 的交點
- 例如 $a = 1, b = 2, c = 3$



和 X 軸的交點為 **$(-3.0, 0.0)$**

```
#include <stdio.h>
int main() {
    double a=1, b=2, c=3, x, y=0;
    x = (b * y + c) / a;
    printf("和 X 軸的交點為 (%f, 0.0)\n", x);
    return 0;
}
```

範例一 (cont'd)

- 3.000000 格式和要求不太一樣, 調整一下

```
#include <stdio.h>
int main() {
    double a=1, b=2, c=3, x, y=0;
    x = (b * y + c) / a;
    printf("直線 %.1f x + %.1f y + %.1f = 0 ", a, b, c);
    printf("和 X 軸的交點為 (%.1f, 0.0)\n", x);
    return 0;
}
```

```
q:\>test02
直線 1.0 x + 2.0 y + 3.0 = 0 和 X 軸的交點為 (3.0, 0.0)
```

範例一 (cont'd)

- 係數 **a, b, c** 可不可以不要是固定的? **讓使用者輸入**

```
#include <stdio.h>
int main() {
    double a, b, c, x, y=0;
    printf("請輸入直線 a x + b y + c = 0 的係數: ");
    scanf("%lf%lf%lf", &a, &b, &c);
    x = (b * y + c) / a;
    printf("直線 %.1f x + %.1f y + %.1f = 0 ", a, b, c);
    printf("和 X 軸的交點為 (%.1f, 0.0)\n", x);
    return 0;
}
```

```
q:\>test03
```

```
請輸入直線 a x + b y + c = 0 的係數: 1.2 -2.5 5.9
```

```
直線 1.2 x + -2.5 y + 5.9 = 0 和 X 軸的交點為 (4.9, 0.0)
```

範例一 (cont'd)

- 當使用者可以自由輸入 **a**, **b**, 與 **c** 時, 前一頁的程式有可能發生執行時的錯誤

```
q:\>test03  
請輸入直線 a x + b y + c = 0 的係數: 0 1 2  
直線 0.0 x + 1.0 y + 2.0 = 0 和 X 軸的交點為 (1.$, 0.0)
```

- 仔細想一下, 發生問題的敘述應該是

$$x = (b * y + c) / a;$$

當係數 **a** 是 0 時, **除法** 是沒有定義的

範例一 (cont'd)

- 當 **a** 的數值很小的時候, 算出來的結果也會很可怕

```
q:\>test03
請輸入直線 a x + b y + c = 0 的係數: 1e-300 1 2
直線 0.0 x + 1.0 y + 2.0 = 0 和 X 軸的交點為 (199999999999999998000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
0000000.0, 0.0)
```

- 修改的方法
- ```
if (a == 0)
 printf("和 X 軸沒有交點\n");
else {
 x = (b * y + c) / a;
 printf("和 X 軸的交點為 (0.1f, 0.0)\n", x);
}
```

# 範例一 (cont'd)

- 如果改成

```
if (a == 0)
 printf("和 X 軸沒有交點\n");
else {
 x = (b * y + c) / a;
 printf("和 X 軸的交點為 (%.1e, 0.0)\n", x);
}
```

- 輸出的結果會變成

```
q:\>test03
請輸入直線 a x + b y + c = 0 的係數: 1 e-300 1 2
直線 1.0 x + 0.0 y + 0.0 = 0 和 X 軸的交點為 (2.1e-317, 0.0)
```

- 更謹慎的改法 `if (a < 1e-10 && a >-1e-10) ... else ...`

10<sup>-10</sup>

-10<sup>-10</sup>

- 看完前面的範例以後, 應該要很清楚地知道：
  - **程式** 是操控電腦完成某項工作時一步一步的指令
  - **寫程式** 是你在教電腦一步一步執行你想要它完成的工作
- 想要順利地寫出一個程式, 你應該要能夠手動完成程式的要求, 把動作的順序記錄下來, 完整地用 C 語言表示出來 ... 就是你的程式了



## 程式撰寫範例二

- 請寫一個程式計算直線  $ax + by + c = 0$  和直線  $dx + ey + f = 0$  的交點
- 重點在於手動算出這兩條直線的交會點

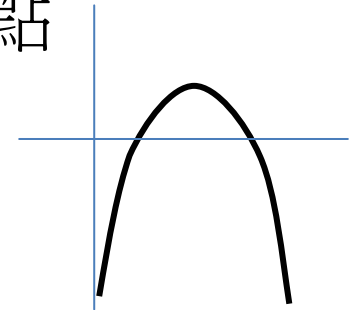
$$\begin{aligned} \text{if } (d \neq 0) \quad dx + ey + f = 0 &\Rightarrow x + e/d y + f/d = 0 && \dots \textcircled{3} \\ &\Rightarrow ax + ae/d y + af/d = 0 && \dots \textcircled{1} \\ &ax + by + c = 0 && \dots \textcircled{2} \end{aligned}$$

$$\textcircled{1} - \textcircled{2} \Rightarrow y = (c - af/d) / (ae/d - b)$$

$$y \text{ 代回 } \textcircled{3} \Rightarrow x = -e/d (c - af/d) / (ae/d - b) - f/d$$

# 程式撰寫範例三

- 讓我們寫一個程式計算一元二次方程式 (拋物線)  
 $a x^2 + b x + c = 0$  的兩個根
- 重點在於手動算出 拋物線 與 X-軸 的交會點



$$d = b^2 - 4 a c$$

$$\text{if } (d > 0) \quad x_1 = (-b + \sqrt{d}) / (2 a)$$

$$x_2 = (-b - \sqrt{d}) / (2 a)$$

if (d==0) 只有一個交點  $x = -b / (2 a)$

if (d<0) 沒有交點

# 程式撰寫範例四

- 請寫一個程式計算等差級數 (等差數列的和)
- 重點在於怎樣手動算出 等差級數 的結果

計算  $a_0 + a_1 + a_2 + a_3 + \dots + a_n$

其中  $a_i = a_0 + i * d$ ,  $d$  是公差

$$\frac{(a_0 + a_n) (n+1)}{2}$$

$$\begin{array}{cccccccc} a_0 & + & a_1 & + & a_2 & + & \dots & + & a_n \\ a_n & + & a_{n-1} & + & a_{n-2} & + & \dots & + & a_0 \\ \hline a_0+a_n & & a_0+a_n & & a_0+a_n & & & & a_0+a_n \end{array}$$