

第二單元

變數的宣告與算術運算

■ 程式設計

相信大家都有使用掌上型計算機計算長方形的面積與周長的經驗，那可要按一長串的按鍵，且邊長在算面積時輸入一次，算周長還要再輸入一次。又例如，使用計算機計算一堆數字的平均，也是要不斷鍵入一長串按鍵，且若有按鍵錯誤的疑慮，還要重複按一次。程式設計就是要解決這些輸入的不便。其次，若使用程式設計工具，那資料只要輸入一次，我們還可繼續深入求這些資料的平均、極大值、不及格個數、排序等統計資料。還有，此一程式還可儲存，下次只要再輸入資料，就可進行此計算。例如，上一單元已經完成以上長方形面積計算程式如下：

```
a=4
b=5
c=a*b
d=2*(a+b)
print(c)
print(d)
```

只要指派 a,b 的值一次，就可計算面積與周長，且此程式還可存檔，任何時機都可再拿來做相同的計算。本單元即要開始介紹如何編寫這些程式，體驗『程式設計』的神奇功效。

變數的命名

前面我們已經用 `a,b` 等符號，代表長方形的邊長，這樣就可求得長方形的面積與周長，『`a,b`』這些符號，在程式設計的領域，我們就稱為變數，電腦會安排記憶體儲存這些『值』。為什麼這些符號稱為變數呢？那是因為這些符號的『值』，在程式執行階段，都可以再任意改變，所以就稱為變數。數學通常用 `a,b,c` 代表已知數，`x,y,z` 代表未知數，物理通常以 `t` 代表時間，`v` 代表速率。Python 的應用領域涵蓋所有科學、工程與商業計算，所以變數種類也多，以下則是 Python 變數命名取用規則。

1. 變數僅能以大小寫的英文字母或底線『`_`』開頭。例如，以下是合法的變數名稱。

```
a
i
sum
_sum
```

以下是一些非法的變數識別字。

```
7eleven    # 不能由數字開頭
%as        # 不能由符號開頭
```

2. 變數由字母、底線開頭後，僅可由字母、底線及數字組合而成，但不得包含空白。例如，以下是一些合法的變數。

```
a123
AB5566
_a_b
```

以下是一些非法的變數。

```
A=         # 不能含有 = 號
sum!       # 不能含有 ! 號
```

```
Age#3      # 不能含有 # 號
a c        # 不能含空白
c+3        # 不得含有加號
```

- 變數的大小寫均視為不同，例如 Score、score 及 SCORE 皆代表不同的變數。
- 變數不得使用保留字，如 if、for 等。以下是 Python 的保留字：（附註：所有程式語言都有保留字，也就是某些單字已經事先定義一些功能，為了避免混淆，當然不能再使用這些單字。就如同現實社會裡，『電視』、『電腦』已經有特定意義，所以就沒有人取名為『電視』或『電腦』。

False	await	else	import	pass
None	break	except	in	raise
True	class	finally	is	return
and	continue	for	lambda	try
as	def	from	nonlocal	while
assert	del	global	not	with
async	elif	if	or	yield

- 變數雖然可用中文，但因為所有程式指令都是英文，若自己使用中文當變數，那還要切換輸入法，真是自找麻煩了。
- 變數要盡量取有意義的英文字。例如，num、total、average，但是本書為避免使用者打字錯誤，那就使用數學老師的習慣，使用 a,b,c,d，這樣反而很簡潔。

變數的宣告

大部分的程式語言變數宣告的同時也要指派其型態是整數、實數或字串，但是 Python 只要指派其初值就好，型態與精密度就通通不用管，反正其精密度通通使用最高標準，這樣就可以滿足所有需求了。請鍵入以下程式，並觀察執行結果。

```
a=2 #整數
print(2**100)
b=3.4 #實數
print(b**100)
```

以上都是一個敘述佔用一列，若要將兩個敘述放在同一列，請用分號『;』，例如：

```
a=1;b=2;print(a+b)
```

但是以下程式，雖然沒有錯，但卻是脫褲子放屁，因為這不是 C 或 Java，每一列不用以分號『;』做結束。

```
a=1;
b=2;
print(a+b);
```

變數不用宣告的困境

變數只要放在左邊，就等於宣告，這樣就可在右邊使用，這樣也會有缺點。請鍵入以下程式，並觀察執行結果。

```
student=0
student=stutent+1
print(student)
```

以上第二列程式指派運算子『=』右邊的 `stutent` 拚字錯誤，Python 可以馬上幫忙找出錯誤（請留意解譯器會出現錯誤提示），很快就可以修正如下：

```
student=0
student=student+1
print(student)
```

得到正確結果。但以下程式您就哭不完了。

```
student=0
stutent=student+1
print(student)
```

請問錯在哪了？剛剛是錯在指派運算子『=』的右邊，Python 幫您找出錯誤，這次錯在指派運算子左邊，那就麻煩了。因為指派

運算子『=』右邊的變數要先宣告，這樣可預防打字拚字錯誤，左邊就是您說了算，沒有幫忙除錯，所以指派運算子左邊的變數要自己特別留意了。又例如，

```
Score=1
score=Score+1
print(Score)
```

也是沒有錯誤信息，但結果也是錯誤，請特別留意運算子左邊的變數名稱不能錯誤。

算術運算子(Arithmetic operators)

小型計算機有加、減、乘、除等鍵，這樣就可計算加、減、乘、除的結果，程式語言要能計算加減乘除，也要有這些算術運算子。下表是 Python 的算術運算子列表：

運算子	定義	優先順序	結合律
**	次方	1	由左至右
+/-	正負號	2	由右至左
*	乘法運算	3	由左至右
/	除法，得實數商	3	由左至右
//	除法，得整數商	3	由左至右
%	求餘數 (Modulus)	3	由左至右
+/-	加法 / 減法運算	4	由左至右
=	指派	14	由右至左

指派運算子 (Assignment operators)

指派運算子的符號為 (=)，其作用為將運算符號右邊運算式的值指派給運算符號左邊的運算元。所以，以下敘述 `sum=a+b` 是將 `a+b` 的值指派給 `sum`。

```
a=5;b=4;sum=0
sum=a+b
```

上式與數學的等號是不同的，所以不要一直糾結為什麼 0 會等於 9。其次，你是不能將常數放在指派運算子的左邊，例如，

```
8 = a
```

此為一個不合法的敘述，但以下敘述，將常數 8 指派給變數 a 是合法的。

```
a = 8
```

算術運算

以下是一些簡單算術運算，請自行鍵入，並觀察與寫出結果。

```
a=5;b=4
print(a+b)
print(a-b)
print(a*b)
print(a/b)
print(a//b)
print(a%b)
print(a**2)
print(9**(1/2))
print(27**(1/3))
print(10**2)
print(10**-2)
print(1/10**2)
```

整數除法與取餘的應用

我們平常的算術運算，對於整數除法與取餘並沒有特別留意。但在程式設計的領域，這兩個運算子有其獨到用途。因為整數除法與取餘可以將一個整數分解為數個數字，且很多應用都會用到此『分解』運算。例如，若要在顯示器顯示 152，那就要將此數字先分解為 1,5,2 三個數字，請自行鍵入以下程式，並觀察結果。

```
a=152
a3=a%10 #2 個位數
a=a//10
a2=a%10 #5 十位數
a=a//10
a1=a #1 百位數
print(a1,a2,a3)
```

數值函式

Python 的算術運算子只有以上 `+, -, *, /, //, %, **`，若所需運算沒有運算子，例如，取絕對值運算，那要如何處理呢？答案是要使用數值處理函式。例如，若要執行取絕對值運算，則可使用 `abs()` 函式，程式如下：

```
x=-3
print(abs(x))
```

自我練習

1. 請指派一個四位數，並將其反向輸出此四個數字且求其數字和。例如，輸入 1234，那可以輸出 4 3 2 1 與 10。（數字之間留空白，且先不要用迴圈）
2. 請寫一個程式，可以指派 4 個數字，然後將其兜成一個 4 位數輸出。例如，指派 `a=3,b=4,c=5,d=6`，那可以輸出『3456』。
3. 請寫一個程式，可以指派 1 個數字，並將其轉為 2 進位。例如，指派 11，則輸出 1 1 0 1（本例先從右邊輸出）。
4. 請寫一個程式，可以指派 1 個數字，並將其轉為 3 進位。例如，指派 11，則輸出 2 0 1（本例先從右邊輸出）。

補充說明

1. 以上方法不只一種，方法沒有優劣，只要自己想出來的都是最好的。

運算子的優先順序(Precedence)

同一敘述，若同時含有多個運算子，此時即需定義運算子的優先順序。例如：

```
x=a+b*c
```

在數學裡，我們定義先乘除再加減，程式語言也是相同，必須定義這些運算子的優先順序，請自行鍵入以下程式，並觀察結果。

```
print(3**2)
print(-3**2)
print(4+3**2%2+1)
```

運算子的結合律(Associativity)

當同一敘述，相鄰的運算子擁有相同優先順序的運算子，此時即需定義運算子是『由左至右』的左結合或『由右至左』的右結合。例如：

```
x=a-b-c
```

同樣是減號(-)，優先順序相同，此時就要靠定義結合律，減法結合律是由左至右，所以以上同義於

```
x=((a-b)-c)
```

而

```
a=b=c=2
```

指派運算子的結合律是由右至左，所以以上式子同義於：

```
(a=(b=(c=2)))
```

以上式子，a、b、c的結果都是2。請鍵入以下程式，並觀察結果。


```
print(6/2*3%2+1)
a=b=c=3
print(a,b,c)
```

變數同時多重指派

Python 允許使用者同時多重指派。請鍵入以下程式，並觀察結果。

```
a,b,c=1,2,3
print(a,b,c)
```

變數同時交換內容

變數同時交換內容是程式設計常需使用的運算，例如，變數 a,b 要交換內容，要先找一個臨時變數 t, 程式如下：

```
a=1;b=2
t=a
a=b
b=t
print(a,b) #2 1
```

Python 就允許使用者同時交換變數內容。請鍵入以下程式，並觀察結果。

```
a,b=1,2
a,b=b,a
print(a,b) #2,1
a,b,c=1,2,3
a,b,c=b,c,a
print(a,b,c) #2,3,1
```

註解 (Comments)

適當的程式註解才能增加程式的可讀性，Python 的註解有兩種方式。第一，使用『`'''`』當註解開頭，直到遇到『`'''`』，兩個『`'''`』中間的文字通通是註解，註解是僅給給人看，電腦不予理會。

```
"""我是註解
我也是註解"""
```

上式『`"""`』符號以後的字串視為註解，編譯程式不予處理，直到遇到『`"""`』為止。其次，同一列中，井號『`#`』後面的也視為註解，編譯器均不予處理。例如：

```
# 我是註解
```

前者，因為可超過兩列，較適合編寫較長的註解，後者，則僅能寫在同一列。

範例2a

請寫一程式，滿足以下條件。

1. 可以指派兩個座標。
2. 計算此兩點座標距離。
3. 輸出此兩點距離。

演算法

1. 已知兩點座標分別是 $(x_1, y_1), (x_2, y_2)$ ，則其距離的公式的數學語言是：

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

程式列印

Python 有次方運算子，所以開根號就是 $1/2$ 次方。其次，因為有運算子優先順序問題，所以請留意括號不能省略。

```
x1=3;y1=0
x2=0;y2=4
d=((x1-x2)**2+(y1-y2)**2)**(1/2)#此為Python語言表示法
print(d)#5.0
```

補充說明

1. 本例所有變數內容先使用『指派』，等待學完下一單元，再改為輸入。
2. 因為有運算子優先順序問題，所以請留意括號不能省略。

自我練習

指派三角形三邊長 a 、 b 、 c ，求其面積。(提示：先計算 $d=(a+b+c)/2$ ，則三角形面積 $=\sqrt{d(d-a)(d-b)(d-c)}$ ，本例假設所指派的三角形三邊長可圍成三角形，例如輸入 3, 4, 5 則得三角形面積 6。其次，並不是任意三條線都可圍成三角形，若要判斷是否可圍成三角形，請繼續研讀第 7 單元)

資料的抽象化

若有一直線方程式是 $ax+by+c=0$ ，點 $p(m,n)$ 到直線的距離是

$$d = \frac{|am+bn+c|}{\sqrt{a^2+b^2}} \quad (\text{提示：絕對值函式是 } \text{abs}())$$

這件事要由電腦作，那就是要將資料從人類約定的書寫習慣 $ax+by+c=0$ 獨立抽象出來。例如，直線方程式是 $3x+4y+5=0$ ，求點 $p(1,2)$ 到直線的距離，首先，以

$$a=3; b=4; c=5$$

約定此為直線

$$3x+4y+5=0$$

然後以

$$m=1; n=2$$

約定此為 $p(1,2)$ ，所以全部程式如下：

```
a=3;b=4;c=5
m=1;n=2
d=abs(a*m+b*n+c)/((a*a+b*b)**(1/2))
print(d)
```

又例如，您要判斷 $q(1,-2)$ 是否在直線上，也是將資料抽象出來，程式如下：(if 請看第七單元)

```
a=3;b=4;c=5
m=1;n=-2
if (a*m+b*n+c)==0:
    print("yes")
else:
    print("no")
```

以上 $3x+4y+5=0$ 是人類在數學上書寫的約定，電腦只要給 a,b,c ，我們就約定此為 $ax+by+c=0$ ，此稱為資料的抽象化或資料的數位化。

範例2b

寫一個程式，可以指派一個一元二次方程式，並求其解（本例假設所輸入的方程式恰有二解）。

運算思維

國中解一元二次方程式是先用十字交叉乘法，十字交叉乘法需要一點判斷，可減少計算，但是本範例使用公式法，公式法雖計算較多，但完全不用判斷，最適合由電腦完成。這種強調多計算少判斷，就是電腦與人類不同的運算思維，人類計算能力較差，所以希望使用一些技巧來簡化計算，但是電腦則是計算能力強，判斷能力差，所以強調用計算來簡化程式。解一元二次方程式的公式法，演算步驟如下：

(1) 設有一元二次方程式如下：

$$ax^2 + bx + c = 0$$

(2) 資料的抽象化，本例指派 a, b, c 三個整數（本例假設整係數方程式），我們就約定此 a, b, c 代表 $ax^2 + bx + c = 0$ 。

(3) 令 $d = \sqrt{b^2 - 4ac}$ 。提示：此為數學語言，Python 是 $d = (b*b - 4*a*c)**(1/2)$ ，括號、乘號均不能省。

(4) 則其二解分別為 $x_1 = \frac{-b+d}{2a}$ ， $x_2 = \frac{-b-d}{2a}$ 。（提示：此為數學語言，Python 是 $x_1 = (-b+d)/(2*a)$ ，括號、乘號均不能省。）

(5) 例如， $2x^2 - 7x + 3 = 0$ 。其解為 $x_1 = 3.0$ ， $x_2 = 0.5$ 。

程式列印

```
a=2;b=-7;c=3#變數先指派，可簡化問題。
d=(b*b-4*a*c)**(1/2)
x1=(-b+d)/(2*a)
x2=(-b-d)/(2*a)
print(x1)#3.0
print(x2)#0.5
```

補充說明

1. 本例先假設所指派的係數一定有實數解，待下一單元再判斷所輸入係數有沒有實數解。
2. 運算思維：近來運算思維廣為被大家討論，我的淺見是，人有人的特性與優勢，電腦有電腦的特性與優勢，所謂運算思維，就是要學習電腦有哪些特性與優勢，然後使用電腦的運算思維寫程式。以本範例為例，人類可能用十字交叉法較快，但是電腦是使用公式法較快，此即為電腦的運算思維，往後各單元還有更多電腦運算思維，學習這些運算思維，就可增加程式設計功力，請繼續研讀以下各單元，即可瞭解。

自我練習

寫一個程式，可以指派一個二元一次方程式，並求其解（本例假設所指派的方程式恰有一解）。

運算思維

1. 解二元一次方程式，國中會先教代入消去法與加減消去法，這都需要一點判斷，但可以簡化計算，本題使用公式法，可完全不用判斷，直接計算而得，這樣最適合計算機了，這也是電腦的運算思維。解二元一次方程式的克拉馬公式演算過程如下：

- (1) 設二元一次方程式如下：

$$a_1x + b_1y = c_1$$

$$a_2x + b_2y = c_2$$

- (2) 指派 $a_1, b_1, c_1, a_2, b_2, c_2$ 六個整數。（本例假設整係數方程式）

- (3) 令 $d = \begin{vmatrix} a_1 & b_1 \\ a_2 & b_2 \end{vmatrix} = a_1b_2 - a_2b_1$ 。（此稱為行列式表示法）

- (4) 則其解分別是 $x = \frac{\begin{vmatrix} c_1 & b_1 \\ c_2 & b_2 \end{vmatrix}}{d} = (c_1b_2 - c_2b_1) / d$

（x 的地方用 c_1, c_2 代替）

$$y = \frac{\begin{vmatrix} a_1 & c_1 \\ a_2 & c_2 \end{vmatrix}}{d} = (a_1c_2 - a_2c_1) / d$$

（y 的地方用 c_1, c_2 代替）

- (5) 例如， $3x + y = 5$

$$x - 2y = -3$$

則其解為 $x=1 \ y=2$

2. 請寫一個程式，可以將時速 (km/h) 轉為每秒幾公尺 (m/s)。

3. 請寫一個程式，可以將每秒幾公尺 (m/s) 轉為時速 (km/h)。
4. 找零問題。請寫一程式，可以輸入消費金額與付款鈔票面值，計算所需找回的錢，並以最節省鈔票與硬幣數量的方式找錢，例如，輸入

26,1000

可以計算所需找 500,100,50,10,5,1 元的鈔票數量。(APCS105 試題，本例請先不要用迴圈)

範例2c

假設某次考試成績資料如下：

55、66、77、88、99

- (1) 請寫一程式儲存以上資料。
- (2) 計算總和。
- (3) 輸出總和與平均。

程式列印

本例 5 筆資料，先用 5 個變數儲存以上資料，所以程式如下：

```
a1=55;a2=66;a3=77;a4=88;a5=99
s=a1+a2+a3+a4+a5
print(s)#385
print(s/5)#77.0
```

補充說明

本例只有 5 筆資料，程式就落落長，那如果有 50 筆，甚至 10000 筆，那不就操死程式設計師，請大家放心，待學習後面單元的迴圈與 list，就會豁然開朗。

自我練習

1. 每一個人請挑一個物理或數學公式，設計一個程式，計算其結果。例如，攝氏轉華氏溫度的公式是『華氏 = 攝氏 * (9/5) + 32』，請寫一程式，可以指派攝氏溫度，並輸出華氏溫度。
2. 以下程式，可以讓您認識 Python 有哪些關鍵字，它會逐一輸出每一個關鍵字，請跟著輸入，最後會統計正確題數。

```
import keyword
r=0
for a in keyword.kwlist :
    print (a)
    b=input("Please type in The word:")
    if b==a :
        r=r+1
print(r)
```

3. 請寫一程式，可以輸入三點座標，輸出其面積。
提示：三角形面積公式如下：

$$\frac{1}{2} \begin{vmatrix} x_1 & y_1 \\ x_2 & y_2 \\ x_3 & y_3 \\ x_1 & y_1 \end{vmatrix} = \frac{1}{2} |(x_1y_2+x_2y_3+x_3y_1-x_2y_1-x_3y_2-x_1y_3)|$$

例如，輸入 (0,0)、(3,0)、(0,4)，結果是 6。