



陣列 Array

Chun-Jung Lin



OUTLINE

- 一維陣列
 - 一維陣列的程式實作
- 二維陣列
 - 二維陣列的程式實作



Think (1/2)

- 本校將各教學大樓教室借出做為國家考試考場，並在每一棟教學大樓一樓張貼考生座位表。
- 請問在五層樓的教學大樓裡，考生座位表可以如何表現出來，讓考生可以很快的知道考場位置？



Think (2/2)

- 某位病患須按日按時吃藥，以控制病情。
- 若條件許可，您有什麼想法可以讓這位病患可以知道自己是否吃過哪幾次的藥。



陣列 (1/3)

■ 定義

- 陣列是指一群具有相同名稱及資料型態的變數之集合。

■ 特性

- 佔用連續記憶體空間。
- 用來表示有序串列之一種方式。
- 各元素的資料型態皆相同。
- 支援隨機存取(Random Access)與循序存取(Sequential Access)。
- 插入或刪除元素時較為麻煩。因為須挪移其他元素。
- 程式內若需要宣告多個變數，且需要依序存取每個變數，此時適合使用陣列取代多個變數。
- 使用迴圈可以存取陣列每一個元素，會比存取多個變數的程式碼簡潔。



陣列 (2/3)

- 陣列可稱為循序串列，因為每一元素的排列是依序的。
- 線性串列又稱**循序串列**(sequential list)或**有序串列**(ordered list)。
- 每一項依據它在串列的位置，可以形成一個線性的排列次序，所以 $x[i]$ 在 $x[i + 1]$ 之前。
- 線性串列經常發生的操作如下：
 - 取出串列中的第 i 項； $0 \leq i \leq n-1$ 。
 - 計算串列的長度。
 - 由左至右或由右至左讀此串列。
 - 在第 i 項加入一個新值，使其原來的第 $i, i+1, \dots, n$ 項變為第 $i+1, i+2, \dots, n+1$ 項。
 - 刪除第 i 項，使原來的第 $i+1, i+2, \dots, n$ 項變為第 $i, i+1, \dots, n-1$ 項。

陣列 (3/3)

- 假設我們需要5個整數變數來存放資料時，那就必須要宣告一個A陣列為整數型態，其註標(index)是按照順序排列從0~4共有5項，其含義如下：

int A[5];

陣列名稱	→	A				
陣列註標	→	0	1	2	3	4
陣列元素	→	A[0]	A[1]	A[2]	A[3]	A[4]



Practice

- 假設班上有55位同學坐在教室裡，老師每次以5位同學一個梯次叫到臺前領取資料。
 - 可以用什麼**準則**每次請5位同學到臺前。
 - 總共要叫幾個梯次？
 - 若要減少到4個梯次叫完全部的同学，每次要叫幾位同學？
 - 請以文字敘述、流程圖或虛擬碼表示做法。



一維陣列的概念 (1/4)

- 陣列(Array)是將**相同資料型別**的多個變數結合在一起，每個陣列中的元素皆可視為變數使用。
- 陣列佔有**連續的記憶體空間**，提供**索引值(Index)**存取陣列內個別元素。
- **陣列第一個元素其索引值為0**，第二個元素其索引值為1，第三個元素其索引值為2，依此類推， n 個元素的陣列，存取陣列最後一個元素其索引值為 $n-1$ 。
- 每個索引值對應唯一一個陣列元素，因此只要指定陣列名稱與索引值，就可以存取陣列中指定的元素。
 - 存取成績陣列索引值為0的元素，就可存取成績陣列的第一個元素。



一維陣列的概念 (2/4)

- 若程式中要計算全班30位同學的資訊科成績的總分，不使用陣列時，需宣告30個變數(例如：`score1`、`score2`、...、`score30`)儲存30個資訊科成績，使用「`sum=score1+score2+...+score30`」加總，獲得資訊科全班總分。
- 若使用陣列，可以使用迴圈控制陣列索引值存取與累加陣列內每一個元素，達成加總的功能。
- 在樣本空間放大時，兩種方式之間的差異會更明顯，如果要計算全年級國文科總分，全年級有500位同學，使用宣告500個變數(例如：`score1`、`score2`、...、`score500`)的方式，就加總而言需寫成「`sum=score1+score2+...+score500`」，這樣的程式非常不易閱讀與撰寫。

一維陣列的概念 (3/4)

- 使用一維陣列info儲存全班資訊科期末考成績，再利用迴圈與陣列索引值概念可以存取陣列中所有元素，計算出資訊科成績的全班總分。





一維陣列的概念 (4/4)

- 利用迴圈變數結合陣列索引值，經由控制陣列索引值可以存取陣列中所有元素。
- 使用一維陣列儲存資訊科成績所撰寫程式碼較簡潔，不需要每一個同學宣告一個變數儲存成績，且新增學生時只需要增加陣列的元素個數與修改迴圈變數的數值範圍。

一維陣列的操作 (1/3)

■ 宣告與初始化

- 程式中使用陣列需先宣告，宣告為指定陣列名稱與陣列的元素個數。
- 初始化意指設定陣列每個元素的值，在程式中指定陣列元素的值。
- 使用 $A[0]=1$ 就可以將陣列 A 的第一個元素設定為 1，也是將數值 1 寫入陣列 A 的第一個元素。

陣列宣告語法 (一)	程式範例
以宣告 k 個元素的陣列為例。 陣列名稱 $= [0] * k$ 陣列名稱 $[0]$ = 陣列第一個元素的值 陣列名稱 $[1]$ = 陣列第二個元素的值 陣列名稱 $[2]$ = 陣列第三個元素的值 ... 陣列名稱 $[k-1]$ = 陣列第 k 個元素的值	以宣告 5 個元素的陣列 A 為例。 $A = [0] * 5$ $A[0] = 1$ $A[1] = 2$ $A[2] = 3$ $A[3] = 4$ $A[4] = 5$ 以上程式碼宣告了陣列 A ，有五個元素，並初始化第一個元素為 1，第二個元素為 2，第三個元素為 3，第四個元素為 4，第五個元素為 5。

一維陣列的操作 (2/3)



除了可以在宣告陣列程式中初始化，也可以於宣告時同時初始化，如下表。

陣列宣告語法 (二)	程式範例
陣列名稱 = [陣列第一個元素的值 , 陣列第二個元素的值 , 陣列第三個元素的值 , ... , 陣列第 k 個元素的值]	<pre>A = [1, 2, 3, 4, 5]</pre> <p>以上程式碼宣告了陣列 A，有五個元素，並初始化第一個元素為 1，第二個元素為 2，第三個元素為 3，第四個元素為 4，第五個元素為 5。</p>

一維陣列的操作 (3/3)

■ 讀取一維陣列

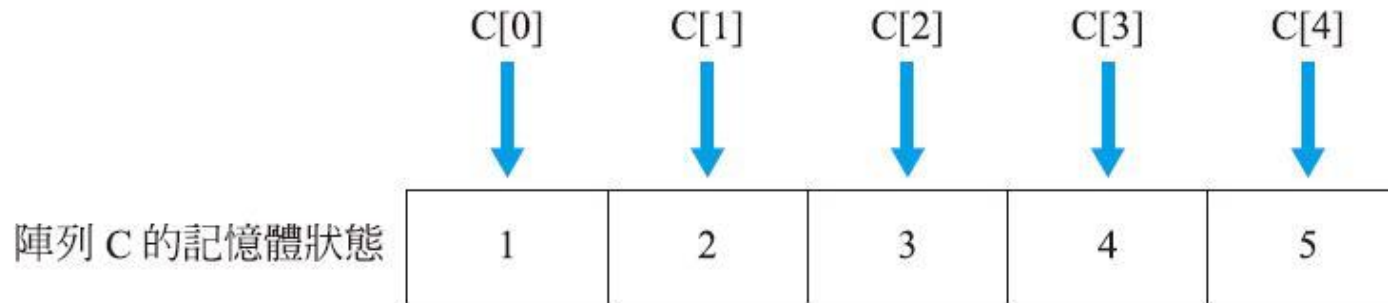
- 使用A[0]讀取陣列A的第一個元素，A[1]讀取陣列A的第二個元素，以此類推。
- 利用迴圈變數與陣列索引值結合，經由控制陣列索引值可以存取陣列中所有元素，陣列元素A[i]的i值就是陣列索引值。
- 當i等於0，就指向陣列A的第一個元素，其值為1；當i等於1，就指向陣列A的第二個元素，其值為2，依此類推。

程式範例	程式執行結果	說明	
<pre>A=[1,2,3,4,5] for i in range(5): print(A[i])</pre>	1	i=0	A[i] 此時值為 1
	2	i=1	A[i] 此時值為 2
	3	i=2	A[i] 此時值為 3
	4	i=3	A[i] 此時值為 4
	5	i=4	A[i] 此時值為 5

以上為印出陣列所有元素的演算法，每一個元素都需要讀取與顯示在螢幕上，演算法效率為 $O(n)$ ， n 為陣列的元素個數。

插入元素到一維陣列 (1/4)

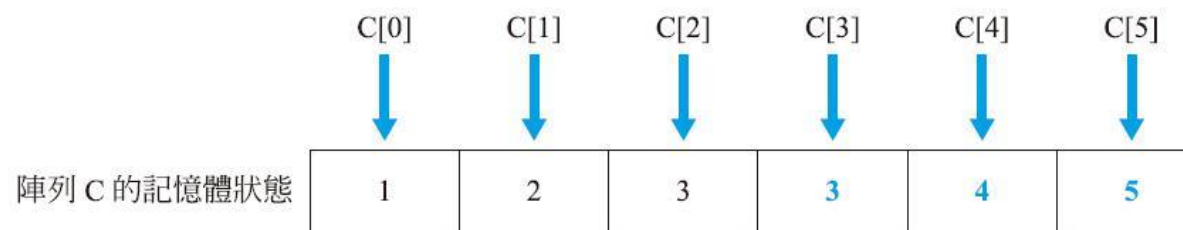
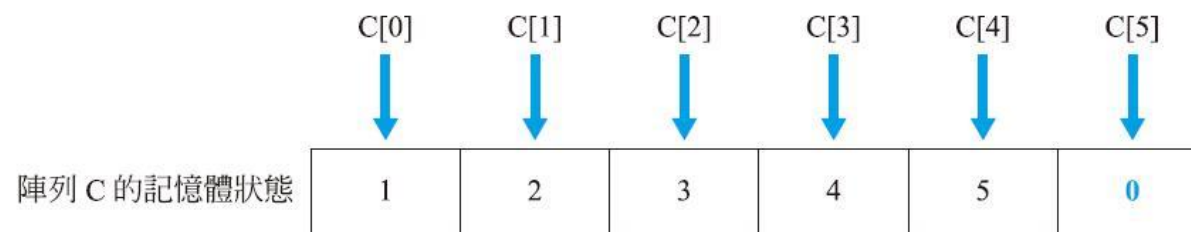
- 當要插入數值“6”到陣列C的第3個元素，則陣列C需要增加一個元素，變成6個元素。



插入元素到一維陣列 (2/4)

■ 方法一 (1/2)

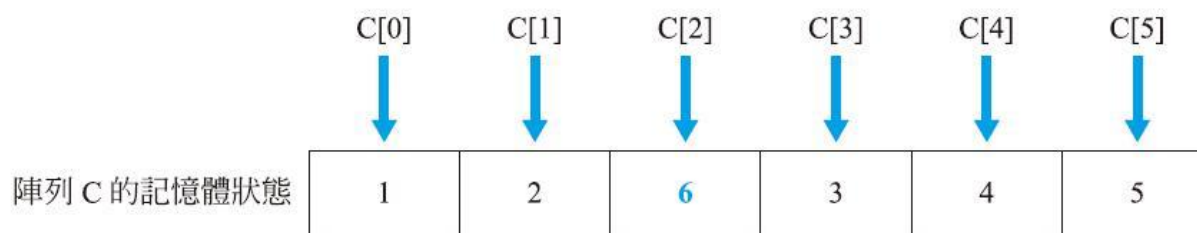
- 1. 使用 `C.append(0)`，新增一個數值 `0` 到陣列 `C` 的最後。
- 2. 使用迴圈 `for` 與 `C[i] = C[i-1]` 每個元素向右移一格。



插入元素到一維陣列 (3/4)

■ 方法一 (2/2)

- 3. 使用 $C[2] = 6$ 設定陣列 C 第3個元素為6。
- 演算法最花時間在每個陣列元素右移一格時，平均需要花 $O(n)$ ， n 為陣列的元素個數。



```
C = [1, 2, 3, 4, 5]
C.append(0)
for i in range(len(C)-1, 2, -1):
    C[i] = C[i-1]
C[2] = 6
print(C)
```



插入元素到一維陣列 (4/4)

■ 方法二：使用insert()

■ 串列名稱.insert(索引值, 串列元素)

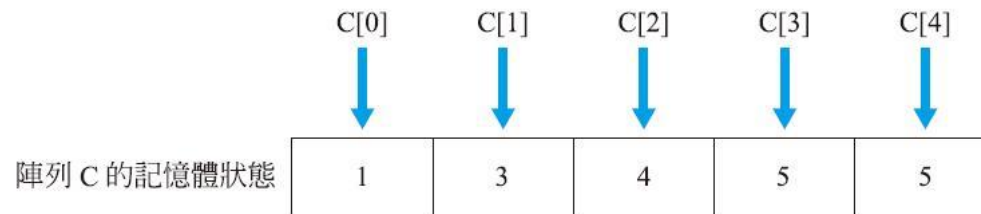
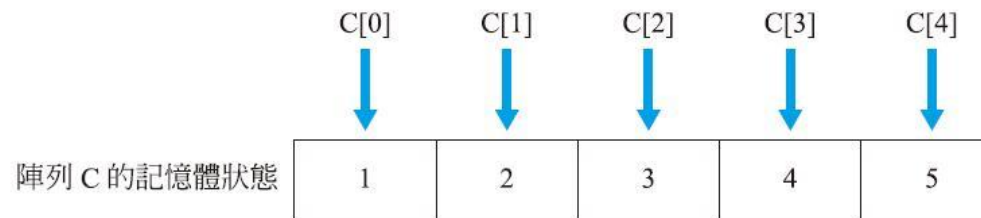
```
C=[1,2,3,4,5]  
C.insert(2, 6)
```

```
C = [1, 2, 3, 4, 5]  
C.append(0)  
for i in range(len(C)-1, 2, -1):  
    C[i] = C[i-1]  
C[2] = 6  
print(C)
```

從一維陣列刪除元素 (1/3)

■ 方法一 (1/2)

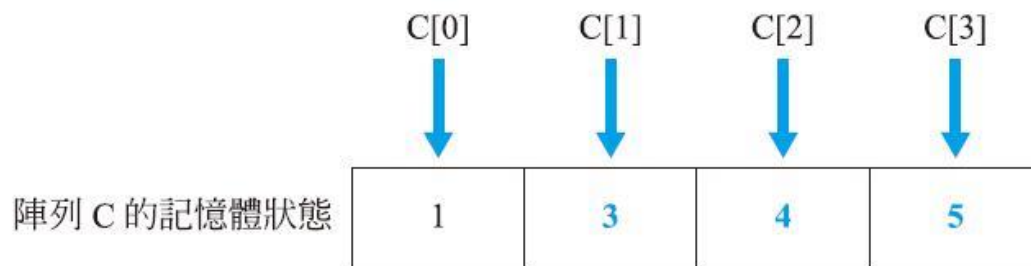
- 當要刪除陣列C的第2個元素，則將第3個元素移動到第2個元素，第4個元素移動到第3個元素。



從一維陣列刪除元素 (2/3)

■ 方法一 (2/2)

- 使用「`C.pop(-1)`」刪除陣列C最後一個元素。
- 演算法最花時間在每個陣列元素左移一格時，平均需要花 $O(n)$ ， n 為陣列的元素個數。



```
C = [1, 2, 3, 4, 5]
for i in range(1, 4):
    C[i] = C[i+1]
C.pop(-1)
print(C)
```



從一維陣列刪除元素 (3/3)

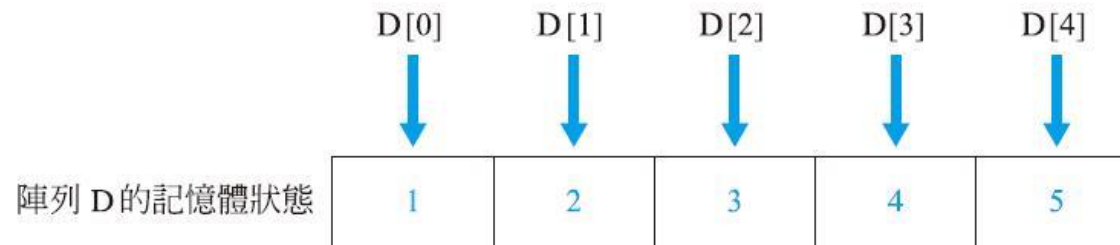
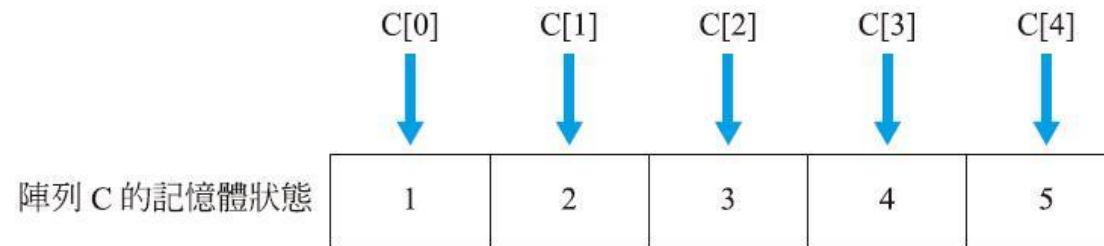
■ 方法二

- del 串列名稱(索引值)
- del 串列名稱(起始值:終止值[:間隔值])

```
C=[1,2,3,4,5]  
del C(1)
```

複製一維陣列 (1/3)

- 方法一：當要複製陣列C的每一個元素到陣列D，使用for迴圈與 $D[i] = C[i]$ ，迴圈變數i由0到4，每次遞增1，取出陣列C的每一個元素複製到陣列D。





複製一維陣列 (2/3)

- 演算法最花時間是在陣列內每一個元素複製到另一個陣列時，平均需要花 $O(n)$ ， n 為陣列的元素個數。

陣列各種操作的演算法效率， n 為陣列中的元素個數。

	讀取所有元素	插入元素	刪除元素	複製陣列
演算法效率	$O(n)$	$O(n)$	$O(n)$	$O(n)$

```
C = [1, 2, 3, 4, 5]
D = [0]*5
for i in range(len(C)):
    D[i] = C[i]
print(D)
```




複製一維陣列 (3/3)

■ 方法二

```
C=[1,2,3,4,5]  
D=C
```

```
C=[1,2,3,4,5]  
D=C[:]
```



Practice

- 題目：取出成績陣列的每一個元素進行加總，計算出成績陣列的總分。
- 想一想：如何取出成績陣列的每一個元素？
- 解題想法：將成績資料置於陣列中，再利用迴圈存取陣列中每一個元素進行加總，當每個元素都存取到時，就可以得到成績的加總。



Practice

- 題目：費氏數列有一特性是第3項為第1項與第2項相加；第4項為第2項與第3項相加，依此類推。初始化費氏數列的第1項為1，且第2項為1，求費氏數列前16項。
- 想一想：陣列使用什麼技術，可以存取陣列內任一個元素？請寫出「將陣列F的第 $n-1$ 項加上陣列F的第 $n-2$ 項的結果存入陣列F的第 n 項」的公式為何？

二維陣列的概念 (1/2)

	第 1 行	第 2 行	第 3 行	第 38 行	第 39 行	第 40 行
	↓	↓	↓	↓	↓	↓	↓
	座號 1 號 學生	座號 2 號 學生	座號 3 號 學生	座號 38 號 學生	座號 39 號 學生	座號 40 號 學生
第 1 列 → 國文	89	78	99	45	44	98
第 2 列 → 英文	88	95	73	44	77	67
第 3 列 → 數學	67	37	77	67	88	82
第 4 列 → 社會	77	67	66	99	99	92
第 5 列 → 自然	98	73	82	33	76	62



二維陣列的概念 (2/2)

- 使用五個一維陣列需宣告五個不同陣列名稱，每個陣列分開計算加總而獲得各科總分。
- 使用二維陣列可以使用五列，每一列元素代表一個科目的成績，利用巢狀迴圈存取二維陣列的每一個元素，計算每一科的總分。
- 使用二維陣列儲存各科成績所撰寫程式碼較簡潔，且新增科目時只需要增加陣列列數與修改迴圈變數範圍。
- 若計算3個年級五科成績需要15個科目時，使用二維陣列的好處就更為明顯，不需要宣告15個一維陣列，只要宣告一個15列的二維陣列即可。

二維陣列的操作 (1/2)

- 二維陣列的宣告是用於定義二維陣列的名稱與陣列中元素的個數，而初始化是指定陣列中元素的值。
- 例如：
 - `score = [[0]*40 for i in range(5)]`，就是宣告一個整數的二維陣列，名稱為 `score`，其列索引值由0到4，共5列，其行索引值0到39，共40行。
 - 程式中使用 `score[1][2]` 可以存取陣列 `score` 的第2列第3行元素。

	第 1 行	第 2 行	第 3 行	第 38 行	第 39 行	第 40 行
第 1 列 →	score[0][0]	score[0][1]	score[0][2]	score[0][37]	score[0][38]	score[0][39]
第 2 列 →	score[1][0]	score[1][1]	score[1][2]	score[1][37]	score[1][38]	score[1][39]
第 3 列 →	score[2][0]	score[2][1]	score[2][2]	score[2][37]	score[2][38]	score[2][39]
第 4 列 →	score[3][0]	score[3][1]	score[3][2]	score[3][37]	score[3][38]	score[3][39]
第 5 列 →	score[4][0]	score[4][1]	score[4][2]	score[4][37]	score[4][38]	score[4][39]



陣列初始化的方法

行數	初始化方式一
1	<code>score = [[0]*40 for i in range(5)]</code>
2	<code>score[0][0] = 90</code>
3	<code>score[0][1] = 56</code>
4	<code>score[0][2] = 98</code>
⋮	⋮
199	<code>score[4][37] = 93</code>
200	<code>score[4][38] = 47</code>
201	<code>score[4][39] = 88</code>

說明

第 1 行：宣告二維陣列名稱爲 score 有 5 列 40 行。

第 2 行：初始化陣列 score 第 1 列第 1 行值爲 90。

第 3 行：初始化陣列 score 第 1 列第 2 行值爲 56。

第 4 行：初始化陣列 score 第 1 列第 3 行值爲 98。

⋮

第 199 行：初始化陣列 score 第 5 列第 38 行值爲 93。

第 200 行：初始化陣列 score 第 5 列第 39 行值爲 47。

第 201 行：初始化陣列 score 第 5 列第 40 行值爲 88。



二維陣列的操作 (2/2)

■ 存取二維陣列

■ 使用陣列索引值存取陣列元素。

- `score[1][2]`的括號內1與2分別表示為列索引值為1，行索引值為2，表示為陣列`score`的第2列第3行元素。
- 將索引值改成變數`i`與變數`j`，當`i`等於1且`j`等於2，則`score[i][j]`相當於`score[1][2]`，`score[i][j]`也是表示陣列`score`的第2列第3行元素。
- 使用迴圈控制變數`i`與變數`j`，當變數`i`與變數`j`變化時，`score[i][j]`所對應的元素也會跟著改變。
- 使用索引值存取陣列的概念可以存取陣列中所有元素。

	第 1 行	第 2 行	第 3 行	第 38 行	第 39 行	第 40 行
第 1 列 →	score[0][0]	score[0][1]	score[0][2]	score[0][37]	score[0][38]	score[0][39]
第 2 列 →	score[1][0]	score[1][1]	score[1][2]	score[1][37]	score[1][38]	score[1][39]
第 3 列 →	score[2][0]	score[2][1]	score[2][2]	score[2][37]	score[2][38]	score[2][39]
第 4 列 →	score[3][0]	score[3][1]	score[3][2]	score[3][37]	score[3][38]	score[3][39]
第 5 列 →	score[4][0]	score[4][1]	score[4][2]	score[4][37]	score[4][38]	score[4][39]



Practice

- 題目：隨機產生二維(5×40)成績陣列(score)每一個陣列元素的數值。假設一列表示一個科目的成績，則此成績陣列可以儲存5科成績。使用程式**計算出各科的總分**。
- 解題想法：使用巢狀迴圈存取二維陣列的每一個元素。外層迴圈控制列，假設變數i為外層迴圈變數，內層迴圈控制行，假設變數j為內層迴圈變數。初始化變數total為0，用於暫存每一個陣列元素的累加值。當i值等於0，j值變化由0到39，可以存取第一列每一個元素，利用 $total = total + score[i][j]$ 計算總分，最後變數total即為第1科成績的總分。外層迴圈變數i加1，則i值等於1，初始化變數total為0，j值變化一樣由0到39，可以存取第二列每一個元素，利用 $total = total + score[i][j]$ 計算總分，最後變數total即為第2科成績的總分。以此方式即可獲得各科總分。

Practice

- 題目：設計一個程式計算矩陣相加的結果。矩陣相加概念舉例如下：假設有兩個2x3矩陣A與B相加，得另一個2x3矩陣C。矩陣C的第1列第1行元素等於矩陣A第1列第1行元素的值，加上矩陣B第1列第1行元素的值；矩陣C的第1列第2行元素等於矩陣A第1列第2行元素的值，加上矩陣B第1列第2行元素的值，依此類推，完成矩陣A與B相加獲得矩陣C。

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$$

$$B = \begin{bmatrix} 1 & 1 & 1 \\ 2 & 2 & 2 \end{bmatrix}$$

$$C = A + B = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} + \begin{bmatrix} 1 & 1 & 1 \\ 2 & 2 & 2 \end{bmatrix} = \begin{bmatrix} 2 & 3 & 4 \\ 6 & 7 & 8 \end{bmatrix}$$



陣列的表示法 (1/11)

- 在C程式語言中常利用陣列設置線性串列，以線性的對應方式將元素 a_i 置於陣列的第 i 個位置上。
- 若要讀取 a_i 時，可利用 a_i 的相對位址等於陣列的起始位址加 $i*d$ 來求得，其中 d 是每一元素所佔空間的大小，C的陣列從0開始。



陣列的表示法 (2/11)

■ 一維陣列(one dimension array)

- 若陣列是 $A[0 : u-1]$ ，並假設每一個元素佔 d 個空間，則 $A[i] = \alpha + i * d$ ，其中 α 是陣列的起始位置。
- 陣列元素： $A[0], A[1], A[2], \dots, A[i], \dots A[u-1]$
- 位址： $\alpha, \alpha + 1, \alpha + 2, \dots, \alpha + (i), \dots \alpha + (u-1)$

■ EX.

- $A[0:5-1]$ ，其中 $u=5$ ，代表有5個元素，其索引為：0,1,2,3,4
- 若起始位置是100，每個資料佔1個單位的空間，每個元素的位址則為：100,101,102,103,104

陣列的表示法 (3/11)

■ 二維陣列

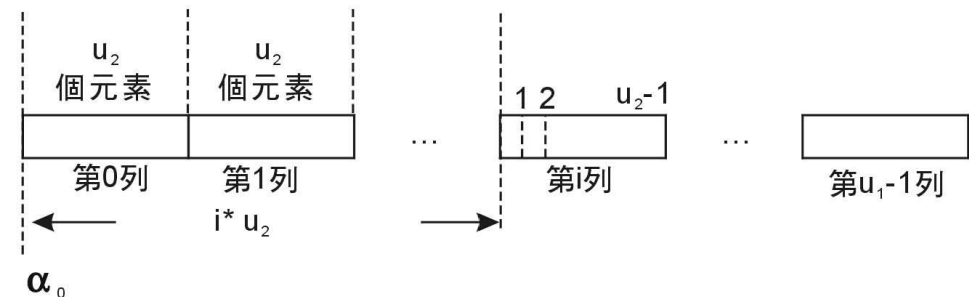
- 假若有一陣列是 $A[0 : u_1 - 1, 0 : u_2 - 1]$ ，表示此陣列有 u_1 列及 u_2 行；每一列是由 u_2 個元素組成。
- 二維陣列化成一維陣列時，對映方式有二種：一種以列為主 (row-major)，二為以行為主 (column-major)。

		行				
		0	1	2	...	$u_2 - 1$
列	0	X	X	X	...	X
	1	X	X	X	...	X
	2	X	X	X	...	X
	⋮					
	$u_1 - 1$	X	X	X	...	X

⇒

■ 以列為主

- 視此陣列有 u_1 (列) 個元素 $0, 1, 2, \dots, u_1 - 1$ ，每一元素有 u_2 (行) 個單位，每個單位佔 d 個空間。
- $A[i, j] = \alpha + i * u_2 * d + j * d$ ，其中 α 為此陣列第一個元素的位址。



陣列的表示法 (4/11)

■ 以行為主

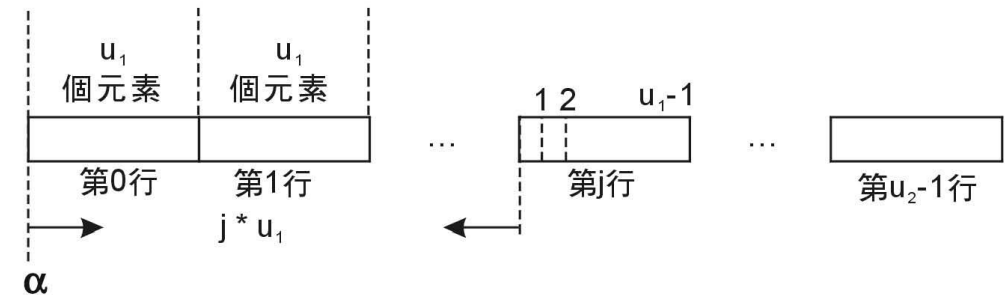
- 視此陣列有 u_2 (行) 個元素 0, 1, 2, ..., u_2 ，其中每一元素含有 u_1 (列) 個單位，每單位佔 d 個空間。

- $A[i,j] = \alpha + j * u_1 d + i * d$

		行				
		0	1	2	...	u_2-1
列	0	X	X	X		X
	1	X	X	X		X
	2	X	X	X		X

	u_1-1	X	X	X		X

⇒





Practice

- 若有一個1維陣列，包含10個單位。請以程式列出每個單位的索引值(index)。
 - EX. [0][1][2][3][4][5][6][7][8][9]
- 若有一個5*5 的2維陣列，請以程式列出每個單位的索引值(index)。
 - [0,0][0,1][0,2][0,3][0,4]
 - [1,0][1,1][1,2][1,3][1,4]
 - [2,0][2,1][2,2][2,3][2,4]
 - [3,0][3,1][3,2][3,3][3,4]
 - [4,0][4,1][4,2][4,3][4,4]



陣列的表示法 (5/11)

■ 假若陣列是 $A[s_1 : u_1, s_2 : u_2]$,
則此陣列共有 $m = u_1 - s_1 + 1$ 列 ,
 $n = u_2 - s_2 + 1$ 行 。

■ 計算 $A(i, j)$ 的位址如下：

■ 以列為主：

$$\square A[i, j] = \alpha + (i - s_1)nd + (j - s_2)d$$

■ 以行為主：

$$\square A[i, j] = \alpha + (j - s_2)md + (i - s_1)d$$

s_1, s_2			s_1, u_2
u_1, s_2			u_1, u_2

範例

假設 $A[-3:5, -4 : 2]$ 之起始位址 $A[-3, -4]=100$ ，以列為主排列，
請問 $A(1,1)$ 所在的位址？（ $d=1$ ）

解： $m=5-(-3)+1=9$, $n=2-(-4)+1=7$, $s_1=-3$, $s_2=-4$, $i=1$, $j=1$

$$A[i, j] = \alpha + (i-s_1)nd + (j-s_2)d$$

$$a[1, 1] = 100 + (1-(-3))*7 + (1-(-4))$$

$$= 100 + (4)*7 + (5)$$

$$= 100 + 33$$

$$= 133$$



-3, -4	-3, -3	-3, -2	-3, -1	-3, 0	-3, 1	-3, 2
-2, -4	-2, 2
-1, -4			...			-1, 2
0, -4			...			0, 2
1, -4			...		1, 1	1, 2
2, -4			...			2, 2
3, -4			...			3, 2
4, -4	4, 2
5, -4	5, -3	5, -2	5, -1	5, 0	5, 1	5, 2



練習題

■ 假設 $A[-3:5, -4:2]$ 且其起始位置 $A[-3, -4]=100$ ，試問以行為主排列時， $A[1, 1]$ 所在位址？（ $d=1$ ）

■ ANS:

■ $100 + 1 - (-4) * 9 + 1 - (-3) = 100 + 45 + 4 = 149$

■ $A(i, j) = \alpha + (j - s_2) * md + (i - s_1) * d$

$$m = u_1 - s_1 + 1 = 5 - (-3) + 1 = 9$$

$$\begin{aligned} \therefore A[1, 1] &= 100 + (1 - (-4)) * 9 + (1 - (-3)) \\ &= 100 + 45 + 4 = 149 \end{aligned}$$

三維陣列



▲立體停車場

- 三維陣列可視為多個二維陣列組合而成。
- 左圖就是 6 個平面停車場組成的立體停車場。

左圖中，📍表示為：

樓，列，行
汽車格 [6 , 1 , 2]
→ ① ② ③

要用到 3 個索引值才能表示某個陣列元素。📦

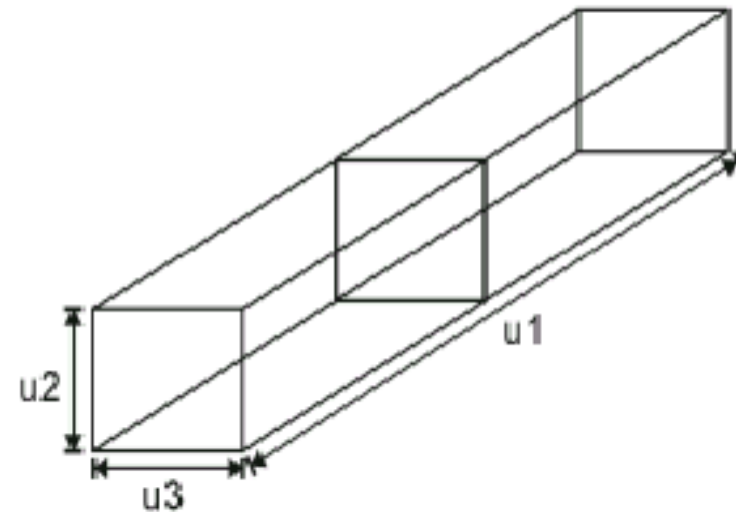
陣列的表示法 (7/11)

■ 三維陣列

- 一般三維陣列皆先化為二維陣列後再對映到一維陣列，對映方式也有二種：

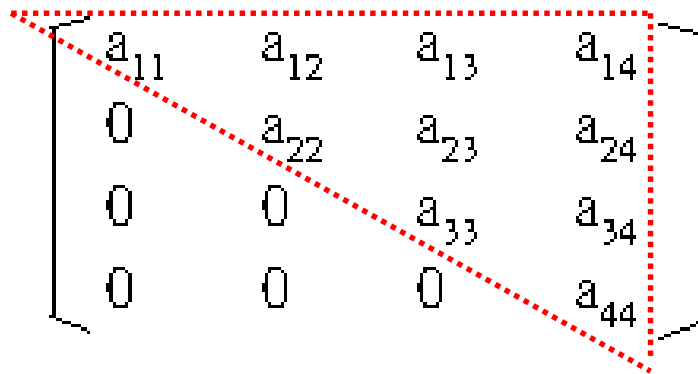
- 以列為主
- 以行為主

三維陣列 $A[0:u_1-1, 0:u_2-1, 0:u_3-1]$ ，如圖 2-3 所示：

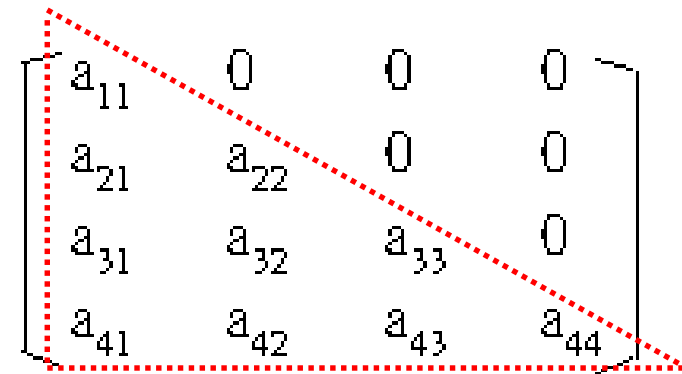


上三角形和下三角形表示法 (1/7)

- 若一矩陣的對角線以下的元素均為零時，亦即 $a_{ij}=0, i>j$ ，則稱此矩陣為上三角形矩陣（upper triangular matrix）。
- 反之若一矩陣的對角線以上的元素均為零，亦即 $a_{ij}=0, i<j$ ，此矩陣稱為下三角形矩陣（lower triangular matrix）。



(a)上三角形矩陣



(b)下三角形矩陣

一個 $n \times n$ 個的上、下三角形矩陣共有 $[n(n+1)]/2$ 個元素，依序對映至 $D(1:[n(n+1)]/2)$ 。



多項式表示法 (1/3)

- 有一多項式 $p = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0$,
 - 我們稱A為n次多項式。
 - $a_i x^j$ 是多項式的項 ($0 \leq i \leq n, 1 \leq j \leq n$) 。
 - 其中 a_i 為係數， x 為變數， j 為指數。



多項式表示法 (2/3)

- 多項式使用線性串列來表示有兩種方法：
 - 使用一個 $n+2$ 長度的陣列，依據指數由大至小依序儲存係數，陣列的第一個元素是此多項式最大的指數，如 $p=(n, a_n, a_{n-1}, \dots, a_0)$ 。
 - 另一種方法只考慮多項式中非零項的係數，若有 m 項，則使用一個 $2m+1$ 長度的陣列來儲存，分別存每一個非零項的指數與係數，而陣列中的第一個元素是此多項式非零項的個數。
- 若有一多項式 $p=8x^5+6x^4+3x^2+12$ ，分別利用第1種和第2種方式來儲存，其情形如下：
 - $p=(5, 8, 6, 0, 3, 0, 12)$
 - $p=(4, 5, 8, 4, 6, 2, 3, 0, 12)$

多項式表示法 (3/3)

- 若是一個兩變數的多項式，需利用二維陣列。
- 若 m, n 分別是兩變數最大的指數，需要一個 $(m+1) \times (n+1)$ 的二維陣列。
- 多項式
 $p_{xy} = 8x^5 + 6x^4y^3 + 4x^2y + 3xy^2 + 7$ ，需要一個 $(5+1) \times (3+1) = 24$ 的二維陣列。

	y^0	y^1	y^2	y^3
x^0	7	0	0	0
x^1	0	0	3	0
x^2	0	4	0	0
x^3	0	0	0	0
x^4	0	0	0	6
x^5	8	0	0	0



魔術方陣 (1/9)

- 有一 $n \times n$ 的方陣，其中 n 為奇數，請你 $n \times n$ 的魔術方陣，將1到 n^2 的整數填入其中，使其各列、各行及對角線之和皆相等。



魔術方陣 (2/9)

- 首先將1填入最上列的中間格，然後往左上方走
 - (1)以1的級數增加其值，並將此值填入空格；
 - (2)假使方格已填滿，則在原地的下一方格填上數字，並繼續做；
 - (3)若超出方陣，則往下到最底層或往右到最右方，視兩者那一個有方格，則將數目填上此方格；
 - (4)若兩者皆無方格，則在原地的下一方格填上數字。



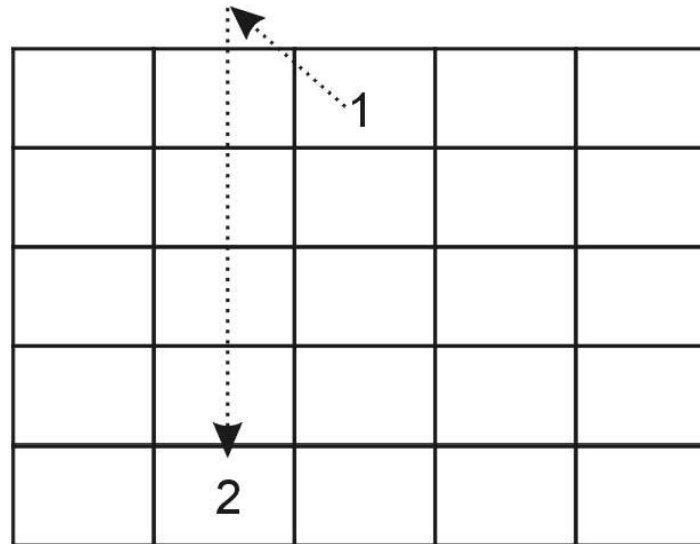
魔術方陣 (3/9)

1. 將1填入此方陣最上列的中間方格，如下所示：

		j				
		0	1	2	3	4
i	0			1		
	1					
	2					
	3					
	4					

魔術方陣 (4/9)

2. 往左上方走，由於超出方陣，依據規格(3)發現往下的最底層有空格，因此將2填上。如下所示：



魔術方陣 (5/9)

3. 往左上方，依據規格(1)將3填上，然後再往左上方，此時，超出方陣，依據規則(3)將4填在最右方的方格，如下所示：

		1		
				4
3				
	2			

魔術方陣 (6/9)

4. 往左上方，依據規則(1)將5填上，再往左上方時，此時方格已有數字，依據規則(2)往5的下方填，如下所示：

		1		
			5	
			6	4
3				
	2			



魔術方陣 (7/9)

5. 依此類推，依據上述四個規格繼續填，填到15的結果如下：

15	8	1		
	14	7	5	
		13	6	4
3			12	10
9	2			11



魔術方陣 (8/9)

6. 此時往左上方，發現往下的最底層和往右的最右方皆無空格，依據規則(4)在原地的下方，將此數字填上，如下所示：

15	8	1		
16	14	7	5	
		13	6	4
3			12	10
9	2			11



魔術方陣 (9/9)

7. 繼續往下填，並依據規則(1)、(2)、(3)、(4)最後的結果如下：

15	8	1	24	17
16	14	7	5	23
22	20	13	6	4
3	21	19	12	10
9	2	25	18	11

■ 各行、各列及對角線之和皆為65。