

陣列(Array)

陣列依其維度可分為一維、二維以及多維。若陣列只有一維，稱之為向量(vector)；陣列為二維，則稱之為矩陣(matrix)。

壹、向量

一、建立向量的基本函數

指令	說明
<code>[]</code>	建立陣列
<code>-3:3</code>	從-3 到 3，間距 1，建立一個列向量
<code>-3:0.1:3</code>	從-3 到 3，間距 0.1，建立一個列向量
<code>linspace(-3,3)</code>	從-3 到 3，建立 100 個元素的列向量
<code>linspace(-3,3,100)</code>	從-3 到 3，建立 100 個元素的列向量
<code>length(x)</code>	查詢向量 x 的元素個數
<code>x'</code>	將向量 x 轉置

```
>> []  
  
ans =  
  
    []  
  
>> -3:3  
  
ans =  
  
    -3    -2    -1     0     1     2     3  
  
>> -3:0.5:3  
  
ans =  
  
-3.0000  -2.5000  -2.0000  -1.5000  -1.0000  -0.5000   0    0.5000   1.0000  
1.5000   2.0000   2.5000   3.0000
```

```
>> x=linspace(-3,3) ;
```

```
>> length(x)
```

```
ans =
```

```
100
```

二、向量的基本處理函數

指令	說明
min(x)	計算向量最小值
max(x)	計算向量最大值
sum(x)	計算向量總和
cumsum(x)	計算向量累積和
prod(x)	計算向量乘積
cumprod(x)	計算向量累積乘積和
sort(x)	將向量中的元素由小而大排序
sort(x, 'descend')	將向量中的元素由大而小排序

```
>> x=[2 1 4 3 5]
```

```
x =
```

```
2    1    4    3    5
```

```
>> min(x)
```

```
ans =
```

```
1
```

```
>> max(x)
```

```
ans =
```

```
5
```

```
>> sum(x)

ans =
    15

>> cumsum(x)

ans =
     2     3     7    10    15

>> prod(x)

ans =
    120

>> cumprod(x)

ans =
     2     2     8    24   120

>> sort(x)

ans =
     1     2     3     4     5

>> sort(x, 'descend')

ans =
     5     4     3     2     1
```

三、 向量元素的擷取與刪除

A=[1,3,5,7,9]	
B=A*2	
A(2)	% 取出向量第二個元素
A(2:4)	% 取出向量第二至四個元素
A([2,5])	% 取出向量第二、五個元素
A(3)=100	% 將向量第三個元素以 100 取代
A(3:5)=0	% 將向量第三至五個元素以 0 取代
A(end)	% 取出向量最後一個元素
A(5)=[]	% 刪除第五個元素

貳、矩陣

一、常用陣列建立函數

指令	說明
<code>zeros(n)</code>	建立 $n \times n$ 的全零矩陣
<code>zeros(m,n,p..)</code>	建立 $m \times n \times p..$ 全零矩陣
<code>ones(n)</code>	建立 $n \times n$ 的全 1 矩陣
<code>eye(n)</code>	建立 $n \times n$ 的單位矩陣
<code>eye(m,n)</code>	建立 $m \times n$ 的單位矩陣
<code>diag(v)</code>	建立以向量 v 為對角元素的矩陣
<code>magic(n)</code>	建立 $n \times n$ 的魔術方陣

```
>> zeros(3)

ans =

    0    0    0
    0    0    0
    0    0    0

>> zeros(3,2)

ans =

    0    0
    0    0
    0    0

>> ones(3)

ans =

    1    1    1
    1    1    1
    1    1    1
```

```
>> eye(3)
```

```
ans =
```

1	0	0
0	1	0
0	0	1

```
>> eye(3,2)
```

```
ans =
```

1	0
0	1
0	0

```
>> diag( [1 2 3] )
```

```
ans =
```

1	0	0
0	2	0
0	0	3

```
>> magic(3)
```

```
ans =
```

8	1	6
3	5	7
4	9	2

二、以亂數建立陣列之函數

指令	說明
rand	產生 0~1 間的齊一分配亂數
rand(n)	建立 0~1 間的 $n \times n$ 齊一分配
rand(m,n,p...)	建立 0~1 間的 $m \times n \dots$ 齊一分配
randn	產生平均數為 0，標準差為 1 的常態分配亂數
randn(n)	建立 $n \times n$ 的常態分配
randn(m,n,p...)	建立 $m \times n$ 的常態分配

```
>> rand

ans =
    0.7482

>> rand(3)

ans =
    0.1622    0.5285    0.2630
    0.7943    0.1656    0.6541
    0.3112    0.6020    0.6892

>> rand(3,2)

ans =
    0.4505    0.9133
    0.0838    0.1524
    0.2290    0.8258

>> randn

ans =

   -0.4326
```

```
>> randn(3)
```

```
ans =
```

-1.6656	-1.1465	-0.0376
0.1253	1.1909	0.3273
0.2877	1.1892	0.1746

```
>> randn(3,2)
```

```
ans =
```

-0.1867	2.1832
0.7258	-0.1364
-0.5883	0.1139

三、矩陣元素的操作

>> A=[1,2,3 ; 4,5,6 ; 7,8,9]	
A = 1 2 3 4 5 6 7 8 9	
>> A(2 , 2)	
ans = 5	
>> A(2, 2:3)	
ans = 5 6	
>> A'	% 將矩陣 A 轉置
ans = 1 4 7 2 5 8 3 6 9	
>> B=A >> B(: , 2:3)=[]	% (:)代表一整列或一整行 % 刪除 2、3 直行
B = 1 4 7	
>> C=A >> C([1,3] , :)=[]	%刪除 1、3 橫列
C = 4 5 6	

四、取出最大值與最小值所在位置

指令	說明
[val, ind]=max(x)	取出陣列 x 中的元素最大值 val 及其所在位置 ind
[val, ind]=min(x)	取出陣列 x 中的元素最小值 val 及其所在位置 ind
[mx col]=max(val)	找出 val 最大值及其位置

>> A=[1,3,2;8,1,9;10,3,2]	
	A = 1 3 2 8 1 9 10 3 2
>> [val, ind]=max(A)	% 以直行為單位取得數值
	val = 10 3 9 ind = 3 1 2
>> [val, ind]=min(A)	
	val = 1 1 2 ind = 1 2 1
>> [mx col]=max(val)	
	mx = 2 col = 3

五、其他陣列元素相關函數

(一) 提取矩陣特定元素

指令	說明
diag(A)	取出矩陣 A 主對角線元素
diag(A, k)	取出矩陣 A 第 k 個主對角線元素(k=-2,-1,0,1,2..)
triu(A)	取出矩陣 A 主對角線以上元素，其他元素設為 0
triu(A, k)	取出矩陣 A 第 k 個主對角線以上(upper)元素，其他元素設為 0
tril(A)	取出矩陣 A 主對角線以下(lower)元素，其他元素設為 0
tril(A, k)	取出矩陣 A 第 k 個主對角線以下元素，其他元素設為 0

>> A=[1,2,3;4,5,6;7,8,9]	A = 1 2 3 4 5 6 7 8 9
>> diag(A)	ans = 1 5 9
>> diag(A,0)	ans = 1 5 9
>> triu(A)	ans = 1 3 2 0 1 9 0 0 2
>> tril(A)	ans = 1 0 0 8 1 0 10 3 2

(二) 重排矩陣元素

指令	說明
<code>fliplr(A)</code>	將陣列 A 元素左右翻轉(left/right)
<code>flipud(A)</code>	將陣列 A 元素上下翻轉(up/down)
<code>flipdim(A, n)</code>	將陣列 A 元素依第 n 個維度翻轉
<code>reshape(A,m,n,...p)</code>	將陣列 A 元素由上而下，由左而右重新排列成 $m \times n \times p$ 矩陣
<code>repmat(A,m,n,...p)</code>	將陣列 A 類似排磁磚方式排成 $m \times n \times p$ 矩陣
<code>rot90(A)</code>	將矩陣逆時鐘旋轉 90 度
<code>rot90(A, k)</code>	將矩陣逆時鐘旋轉 $k \times 90$ 度，k 為整數

```
>> A=magic(3)
```

```
A =
```

```

8     1     6
3     5     7
4     9     2
```

```
>> fliplr(A)
```

```
ans =
```

```

6     1     8
7     5     3
2     9     4
```

```
>> flipud(A)
```

```
ans =
```

```

4     9     2
3     5     7
8     1     6
```

```
>> reshape(A, 1, 9)
```

```
ans =
```

```

8     3     4     1     5     9     6     7     2
```

```
>> repmat(A, 2, 2)
```

```
ans =
```

8	1	6	8	1	6
3	5	7	3	5	7
4	9	2	4	9	2
8	1	6	8	1	6
3	5	7	3	5	7
4	9	2	4	9	2

```
>> rot90(A)
```

```
ans =
```

6	7	2
1	5	9
8	3	4

```
>> rot90(A, 3)
```

```
ans =
```

4	3	8
9	5	1
2	7	6

(三) 合併陣列

指令	說明
[A,B]	橫向合併
[A;B]	垂直合併
cat(1,A,B,...)	以第一個維度方向排列合併(相當垂直合併)
cat(2,A,B,...)	以第二個維度方向排列合併(相當水平合併)
cat(3,A,B,...)	以第三個維度(頁)方向排列合併
cat(4,A,B,...)	以第四個維度方向排列合併

```
>> A=[1 2 3 ; 4 5 6]

A =
     1     2     3
     4     5     6

>> B=2*A

B =
     2     4     6
     8    10    12

>> [A, B]

ans =
     1     2     3     2     4     6
     4     5     6     8    10    12

>> [A ; B]

ans =
     1     2     3
     4     5     6
     2     4     6
     8    10    12
```

六、矩陣基本運算函數

指令	說明
A+B	
A-B	
A*B	
A/B	
A'	計算 A 矩陣的共軛矩陣(但當元素為實數時，將等於轉置矩陣)
inv(A)	計算矩陣反矩陣(inverse)
det(A)	計算矩陣行列式(determinate)
expm(A)	計算矩陣的指數
logm(A)	計算矩陣的對數
sqrtn(A)	計算矩陣的開平方根

七、矩陣左除與右除

指令	說明
$A \setminus B$	A 左除 B，相當於 $A^{-1} \times B$ 或 $AX=B$ 的解。
B / A	B 右除 A，相當於 $B \times A^{-1}$ 或 $XA=B$ 的解。

八、矩陣內的元素對元素之運算

指令	說明
$A.*B$	將 A 矩陣每一個元素乘以 B 矩陣每一個相對應的元素
$A.^n$	計算方陣 A 每一個元素的 n 次方
$A.'$	計算矩陣 A 的轉置矩陣
$A./B$	將 A 矩陣每一個元素除以 B 矩陣每一個相對應的元素
$A.\backslash B$	將 B 矩陣每一個元素除以 A 矩陣每一個相對應的元素

九、查詢陣列資訊相關函數

指令	說明
size(x)	查詢陣列維度(2X3)
length(x)	查詢陣列行數(3 行數)
ndims(x)	查詢陣列維數(2 維)
numel(x)	查詢陣列元素總數(number of element)(6 個)

```
>> A=[1 2 3 ; 4 5 6]
```

```
A =
```

```
    1    2    3
    4    5    6
```

```
>> size(A)
```

```
ans =
```

```
    2    3
```

```
>> length(A)
```

```
ans =
```

```
    3
```

```
>> ndims(A)
```

```
ans =
```

```
    2
```

```
>> numel(A)
```

```
ans =
```

```
    6
```