

第七章

字串處理

謝明興 撰

7-1 字元與字串	2
7-2 字串處理指令	4
7-3 字串數值轉換指令	10
習題	13

7-2 MATLAB 程式設計基礎

<http://www.tau.edu.tw/shiehms/matlab.htm>
<http://www.tau.edu.tw/shiehms/m7.zip>

7-1 字元與字串

MATLAB 幾個字元 **characters** 可以構成字串 **strings**。 **MATLAB** 是以單引號 (')來括住字串，字串是以列向量 **row vector** 的方式儲存，列向量的每一元素即為字元，以 **ASCII** 碼的方式儲存。

在 **MATLAB** 變數中要呈現單引號(')，只要重複(輸入兩次)輸入單引號(')即可。
計算字串長度可用 **length()**函式。

檢視字串儲存的內碼內容可用 **double** 指令。

將字串內碼轉換為字元可用 **char** 指令。

字串是以列向量的方式呈現，因此要並排字串 **S1, S2** 只要用[**S1 S2**]即可。

字串變數佔用的記憶體空間是倍精確度浮點數的 1/4。



str01.m

```
%str01.m  
s1='存在即幸福'  
s2='我是小惡魔'  
s3=[s1 s2]  
s4='I'm busy, let me alone'  
a4= length(s4)
```



str01.m 執行結果

```
>> str01  
s1 =  
存在即幸福  
s2 =  
我是小惡魔
```

```
s3 =  
存在即幸福我是小惡魔  
s4 =  
I'm busy, let me alone  
a4 =  
22
```

截去字串變數尾部空白可用 **deblank**(字串變數)指令。



delstr.m

```
%delstr.m  
s1='存在即幸福'  
length(s1)  
s2='我是小惡魔'  
s3=[s1 s2]  
length(s3)  
s4=[deblank(s1) s2]  
length(s4)
```



delstr.m 執行結果

```
>> delstr  
s1 =  
存在即幸福  
ans =  
20  
s2 =  
我是小惡魔  
s3 =  
存在即幸福 我是小惡魔  
ans =  
25  
s4 =  
存在即幸福我是小惡魔  
ans =  
10
```

7-4 MATLAB 程式設計基礎

7-2 字串處理指令

1. strcmp(s1, s2)

比較兩字串 **s1**, **s2**，相同時傳回 **1**，不相同時傳回 **0**。

2. strcmpi(s1, s2)

比較兩字串(忽略大小寫)**s1**, **s2**，相同時傳回 **1**，不相同時傳回 **0**。



strcmp1.m

```
%strcmp1.m  
s1='存在即幸福'  
s2='我是小惡魔'  
s3='存在即幸福'  
fprintf('比較%s %s 傳回 %d \n', s1, s2, strcmp(s1, s2))  
fprintf('比較%s %s 傳回 %d \n', s1, s3, strcmp(s1, s3))
```



strcmp1.m 執行結果

```
>> strcmp1  
s1 =  
存在即幸福  
s2 =  
我是小惡魔  
s3 =  
存在即幸福  
比較存在即幸福 我是小惡魔 傳回 0  
比較存在即幸福 存在即幸福 傳回 1
```

3. strncmp(s1, s2, n)

比較兩字串 **s1**, **s2** 前 **n** 個字元，相同時傳回 **1**，不相同時傳回 **0**。

4. strncmpi(s1, s2, n)

比較兩字串(忽略大小寫)**s1**, **s2** 前 **n** 個字元，相同時傳回 **1**，不相同時傳回 **0**。



strncmp1.m

```
%strncmp1.m
clear s1, s2;
s1='存在即幸福'
s2='存在的意義'
n=2;
fprintf('比較%s %s 前 %d 個字元傳回 %d \n', s1, s2, n, strncmp(s1, s2, n))
m=3;
fprintf('比較%s %s 前 %d 個字元傳回 %d \n', s1, s2, m, strncmp(s1, s2, m))
```



strncmp1.m執行結果

```
>> strncmp1
s1 =
存在即幸福
s2 =
存在的意義
比較存在即幸福 存在的意義 前 2 個字元傳回 1
比較存在即幸福 存在的意義 前 3 個字元傳回 0
```

5. findstr(s1, s2) or findstr(s2, s1)

尋找短字串在長字串啟始的字元位置，不知 s1, s2 那個字串長時用 findstr() 取代 strfind()。

6. strfind(s1, s2)

尋找短字串 s2 在長字串 s1 啟始的字元位置，s2 比 s1 長時傳回 []。

7-6 MATLAB 程式設計基礎

findstr1.m

```
%findstr1.m
s1='愛我的人為我癡心不悔我卻為我愛的人甘心一生傷悲'
k= findstr(s1, '我')
k= findstr('我', s1)
k= strfind(s1, '我')
k= strfind('我', s1)
```

findstr1.m執行結果

```
>> findstr1
s1 =
愛我的人為我癡心不悔我卻為我愛的人甘心一生傷悲
k =
     2     6    11    14
k =
     2     6    11    14
k =
     2     6    11    14
k =
     []
```

7. **strvcat(s1, s2, s3, ...)**

將字串 **s2, s2, s3** 以垂直並排的方式排列，用來建立任意長度的大的字串矩陣。

8. **strcat(s1, s2, s3, ...)**

將字串 **s2, s2, s3** 以水平並排的方式排列。

9. **strmatch(s1, strarray)**

尋找出現在 **strarray** 列陣列中 **s1** 的列數位置。

strmatch(s1, strarray, 'exact')

尋找出現在 **strarray** 列陣列中跟 **s1** 完全相同的列數位置。

下個範例傳回第 **1, 2** 列，是因'愛我'字串起頭的只有第一列第二列。



strmatch1.m

```
%strmatch1.m
s1='愛我的人為我癡心不悔'
s2='愛我的人為我付出一切'
s3='我卻為愛我的人'
s4='放不下愛我 的人'
s5=strvcat(s1, s2, s3, s4)
k= strmatch('愛我', strvcat(s1, s2, s3, s4))
k= strmatch('愛我', strvcat(s1, s2, s3, s4), 'exact')
```



strmatch1.m執行結果

```
>> strmatch1
s1 =
愛我的人為我癡心不悔
s2 =
愛我的人為我付出一切
s3 =
我卻為愛我的人
s4 =
放不下愛我 的人
k =
     1
     2
k =
Empty matrix: 0-by-1
```

10. strrep(s1, r1, p1)

將出現在 **s1** 的字串 **r1** 以 **p1** 取代。

7-8 MATLAB 程式設計基礎



strrep1.m

```
%strrep1.m  
s1='愛我的人為我癡心不悔'  
k1= strrep( s1, '愛我的人', '不在意我的人')
```



strrep1.m執行結果

```
>> strrep1  
s1 =  
愛我的人為我癡心不悔  
k1 =  
不在意我的人為我癡心不悔
```

11. strtok(s1, d)

傳回 **s1** 字串出現 **d** 字元之前的字串。 **strtok(s1)**傳回 **s1** 字串出現空白字元之前的字串。



strtok1.m

```
%strtok1.m  
s1='愛我的人為我癡心不悔'  
k1= strtok( s1, '為')  
k2= strtok(s1, '心')  
e1='where there is a will, there is a way'  
k3= strtok(e1)
```




strtok1.m執行結果

```
>> strtok1
s1 =
愛我的人為我癡心不悔
k1 =
愛我的人
k2 =
愛我的人為我癡
e1 =
where there is a will, there is a way
k3 =
where
```

12. upper(s)

將 **s** 字串中的小寫字元改為大寫。

13. lower(s)

將 **s** 字串中的大寫字元改為小寫。



upper1.m

```
%upper1.m
e1='where there is a will, there is a way'
k1= upper(e1)
k2= lower(k1)
```



upper1.m執行結果

```
>> upper1
e1 =
where there is a will, there is a way
k1 =
WHERE THERE IS A WILL, THERE IS A WAY
k2 =
where there is a will, there is a way
```

7-10 MATLAB 程式設計基礎

7-3 字串數值轉換指令

1. num2str(n) or num2str(n, d)

將數值 **n** 轉換為字串，或將數值 **n** 轉換為 **d** 長度的字串。

2. int2str(n)

將數值 **n** 四捨五入後轉換為字串。



num2str1.m

```
%num2str1.m
n1=[3.52 6.183 pi; 9.2 5.7312 3.6664]
k1= num2str(n1)
k2= num2str(n1, 1)
k3= num2str(n1, 2)
p1= int2str(n1)
```



num2str1.m執行結果

```
>> num2str1
n1 =
    3.5200    6.1830    3.1416
    9.2000    5.7312    3.6664
k1 =
3.52    6.183    3.1416
9.2    5.7312    3.6664
k2 =
4      6      3
9      6      4
k3 =
3.5    6.2    3.1
9.2    5.7    3.7
p1 =
4 6 3
9 6 4
```

3. x= str2num(s)

將字串矩陣 **s** 轉換為數值陣列

4. str2double(s)

將字串 **s** 轉換為倍精確度浮點數。



str2num1.m

```
%str2num1.m  
s1=['1  2  3'  
     '4  5  6'  
     '7  8  9']  
x1= str2num(s1)
```



str2num1.m執行結果

```
>> str2num1  
s1 =  
1  2  3  
4  5  6  
7  8  9  
x1 =  
    1    2    3  
    4    5    6  
    7    8    9
```

5. hex2num(s)

將十六進位字串 **s** 轉換為倍精確度浮點數。

6. hex2dec(s)

將十六進位字串 **s** 轉換為十進位整數。

```
>> hex2dec('100')  
ans =  
    256  
>> hex2dec('10')  
ans =  
    16
```

7. dec2hex(d)

將十進位整數 **d** 轉換為十六進位字串，**d** 需為小於 2^{52} 的非負整數。

```
>> dec2hex(4095)  
ans =  
FFF
```

7-12 MATLAB 程式設計基礎

8. bin2dec(b)

將二進位字串 **b** 轉換為十進位整數，字串中的空白字元會忽略不計。

```
>> bin2dec('010 111')  
ans =  
    23
```

9. dec2bin(d)

將十進位整數 **d** 轉換為二進位字串，**d** 需為小於 2^{52} 的非負整數。

dec2bin(d, n)：轉換的二進位數為 **n** 位數。

```
>> dec2bin(23)  
ans =  
    10111  
>> dec2bin(23, 8)  
ans =  
    00010111
```

10. base2dec(s, b)

將基底為 **b** 的字串 **s** 轉換為十進位整數。

```
>> base2dec('210', 3)  
ans =  
    21
```



11. dec2base(d, b)

將基底為 **b** 的數字 **d** 轉換為字串，**d** 需為小於 2^{52} 的非負整數。

dec2base(d, b, n)：將基底為 **b** 的數字 **d** 轉換為 **n** 位長度的字串。

```
>> dec2base(20, 3)  
ans =  
    202  
>> dec2base(108, 2, 8)  
ans =  
    01101100
```

習題

-  1. 利用 **bitset** 指令將 **dec2base(108, 2, 8)** 右邊算來的第四個字元設定為 **0**。
-  2. 利用 **bitget** 指令將 **dec2base(108, 2, 8)** 取出右邊算來的第四個字元。