

第九章 程式控制流程

本章學習目標

- 認識關係運算子與邏輯運算子
- 學習選擇性敘述的用法
- 學習各種迴圈的用法
- 探討迴圈的效率問題

關係運算子

- 關係運算子（relational operators）可用來比較二數字之間的大小。

關係運算子	說明
<	小於
<=	小於或等於
>	大於
>=	大於或等於
==	等於
~=	不等於

```
>> 4<=12
ans =
     1

>> 5>6
ans =
     0

>> test=[1 3 4]<[2 2 1]
test =
     1     0     0

>> whos test
Name      Size  Bytes  Class
test      1x3    3      logical array
```

logical陣列之建立與應用

```
>> a=[1 0 1 1]
```

```
a =  
    1    0    1    1
```

```
>> logical(a)
```

```
ans =  
    1    0    1    1
```

```
>> logical([3 2 0 12.4 0])
```

```
ans =  
    1    1    0    1    0
```

```
>> a=[-1 0 2 3 1 5]
```

```
a =  
   -1    0    2    3    1    5
```

```
>> b=a<2
```

```
b =  
    1    1    0    0    1    0
```

```
>> a(b)
```

```
ans =  
   -1    0    1
```

```
>> a(a<2)
```

```
ans =  
   -1    0    1
```

```
>> a(logical([1 0 1 1]))
```

```
ans =  
   -1    2    3
```

```
>> a(a>0)=-9
```

```
a =  
   -1    0   -9   -9   -9   -9
```

find() 函數的使用

函數	說明
ind=find(array)	找出陣列array中，元素值不0之元素的一維索引值，並把此索引值設定給變數ind存放
[r,c]=find(array)	同上，但回應陣列的二維索引值，並把"列"索引值設定給變數r存放，把"行"索引值設定給c存於。

```
>> a=[1 2 3 6;3 2 1 1;9 3 7 0]
```

```
a =  
    1    2    3    6  
    3    2    1    1  
    9    3    7    0
```

```
>> b=(a>=6)
```

```
b =  
    0    0    0    1  
    0    0    0    0  
    1    0    1    0
```

```
>> find(b)
```

```
ans =  
     3  
     9  
    10
```

```
>> [r,c]=find(a>=6)
```

```
r =  
     3  
     3  
     1  
c =  
     1  
     3  
     4
```

```
>> find([0 2 0;0 1 0;0 0 0])
```

```
ans =  
     4  
     5
```

邏輯運算子

邏輯運算子	說明
&	and運算。兩個運算元必須同為true，其結果才為true
	Or運算。兩個運算元中，只要有一個為true，其結果便為true
~	not運算。也就是把true變為false，或把false變為true

```
>> 4&0
ans =
    0

>> [1 0 3] | [2 0 0]
ans =
     1     0     1

>> ~[0 2 2 5 0]
ans =
     1     0     0     0     1
```

6

all() 與any()函數

表 9.1.4 any() 與 all()

函數	說明
all(v)	當向量 v 裡所有的元素皆為 true (非零) 時，則傳回 1，否則傳回 0
any(v)	只要向量 v 裡有任何一個元素為 true (非零) 時，則傳回 1，否則傳回 0

```
>> any([1 2 0 0 0])
ans =
     1
```

```
>> any([0 0 0])
ans =
     0
```

```
>> all([1 2 0 0 0])
ans =
     0
```

```
>> a=[1 2 3;0 3 2;3 0 1]
a =
     1     2     3
     0     3     2
     3     0     1
```

```
>> all(a)
ans =
     0     0     1
```

```
>> all(all(a))
ans =
     0
```

7

$$A = \begin{bmatrix} 2 & 0 & 1 \\ 3 & 5 & 9 \\ 0 & 4 & 6 \end{bmatrix}$$

回答下列問題：

1. `all(A)`
2. `all(all(A))`
3. `any(A)`

性質測試函數

表 9.1.5 性質測試函數

函 數	說 明
<code>ischar(a)</code>	測試引數 a 是否為一個字元陣列
<code>isempty(a)</code>	測試引數 a 是否為一個空陣列
<code>isequal(a,b)</code>	測試引數 a 與 b 裡的元素個數與數值是否均相等
<code>isfloat(a)</code>	測試引數 a 是否為一個浮點數陣列 (包含虛數)
<code>isinteger(a)</code>	測試引數 a 是否為一個 n -bit 整數陣列
<code>islogical(a)</code>	測試引數 a 是否為一個邏輯型態的陣列
<code>isnan(a)</code>	測試引數 a 是否為一個 NaN (not a number) 的陣列
<code>isnumeric(a)</code>	測試引數 a 是否為數值 (包含 n -bit 整數、實數與虛數)
<code>isprime(a)</code>	測試引數 a 是否為質數
<code>isreal(a)</code>	測試引數 a 是否為實數 (含邏輯型態, 但不包含 n -bit 整數)
<code>isscalar(a)</code>	測試引數 a 是否為純量 (scalars)
<code>issorted(a)</code>	測試引數 a 是否為已經排序好
<code>isspace(a)</code>	測試陣列 a 裡的字元是否為空白字元。若是, 則回應 1, 否則回應 0
<code>isvector(a)</code>	測試引數 a 是否為一個向量

性質測試函數的範例

```
>> isempty([])
ans =
    1
```

```
>> isequal([1 2 3 4],[1 2 3 0])
ans =
    0
```

```
>> isfloat(5)
ans =
    1
```

```
>> isfloat(uint8(5))
ans =
    0
```

```
>> isnumeric(uint8(5))
ans =
    1
```

```
>> isscalar(3)
ans =
    1
```

```
>> isscalar([3 3])
ans =
    0
```

10

選擇性敘述 if-elseif-else

指令	說明
if 判斷條件 敘述主體 end	若判斷條件為true，則執行敘述主體
If 判斷條件 敘述主體 1 else 敘述主體 2 end	若判斷條件為true，則執行敘述主體 1，否則執行敘述主體 2

```
%script9_1.m
if 5>3
    disp('5大於3')
end
```

```
>> script9_1
5大於3
```

```
function func9_1(num)
if mod(num,2)==0
    fprintf('%g是偶數\n',num)
else
    fprintf('%g是奇數\n',num)
end
```

```
>> func9_1(12)
12是偶數
```

```
>> func9_1(35)
35是奇數
```

```
>> func9_1(3.6)
3.6是奇數
```

11

修正func9_1() 的問題

```
function func9_2(num)
if num-fix(num)==0 & num>0
    if mod(num,2)==0
        fprintf('%g是偶數\n',num)
    else
        fprintf('%g是奇數\n',num)
    end
else
    fprintf('%g不是正整數\n',num)
end
```

```
>> func9_2(12)
12是偶數
```

```
>> func9_2(12.44)
12.44不是正整數
```

```
>> func9_2(13)
13是奇數
```

第二行的fix() 可用來取出數值的整數部份。若num-fix(num)的結果為0，代表num並沒有小數部分，所以可藉此來判定輸入的引數num是否為整數。

Int16()以四捨五入取出引數整數值

12

隨堂測驗

- 寫一函數(fun_sum1)輸入2個引數，輸出此2引數的差。
- 寫一函數(fun_sub1)輸入2個引數，輸出此2引數的乘積。
- 寫一主程式，設定2個數值分別為A和B，當A小於等於B時，呼叫fun_sum1函數，計算此2數值的差; 當A大於B時，呼叫fun_sub1函數計算此2數值的乘積值。

13

if-elseif-else敘述

指令	說明
if 判斷條件 1 敘述主體 1 elseif 判斷條件 2 敘述主體 2 elseif 判斷條件 3 敘述主體 3 ... else 敘述主體 n end	若判斷條件1成立，則執行敘述主體1，否則繼續執行判斷條件2。若判斷條件2成立，則執行敘述主體2，否則繼續執行判斷條件3，以此類推。若所有判斷條件都不成立，則執行敘述主體n <pre>function func9_3(num) if isinteger(num) disp('傳入的引數是n-bit整數') elseif islogical(num) disp('傳入的引數是邏輯型態') elseif isfloat(num) disp('傳入的引數是浮點數') else disp('傳入的引數是其它型態') end</pre>

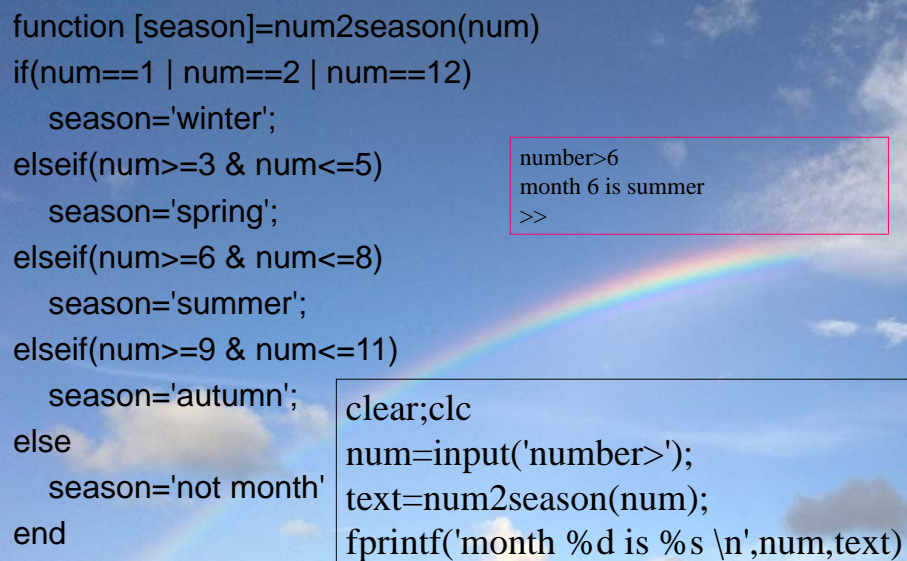
```
>> func9_3(logical(1))
傳入的引數是邏輯型態
>> func9_3(uint8(1))
傳入的引數是n-bit整數
```

```
>> func9_3(3.4)
傳入的引數是浮點數
>> func9_3('a string')
傳入的引數是其它型態
```

14

- 試寫一函數，可接收月份的數值，然後以if條件敘述判斷其所屬的季節，春季：3-5月，夏季：6-8月，秋季：9-11月，冬季：12-2月。

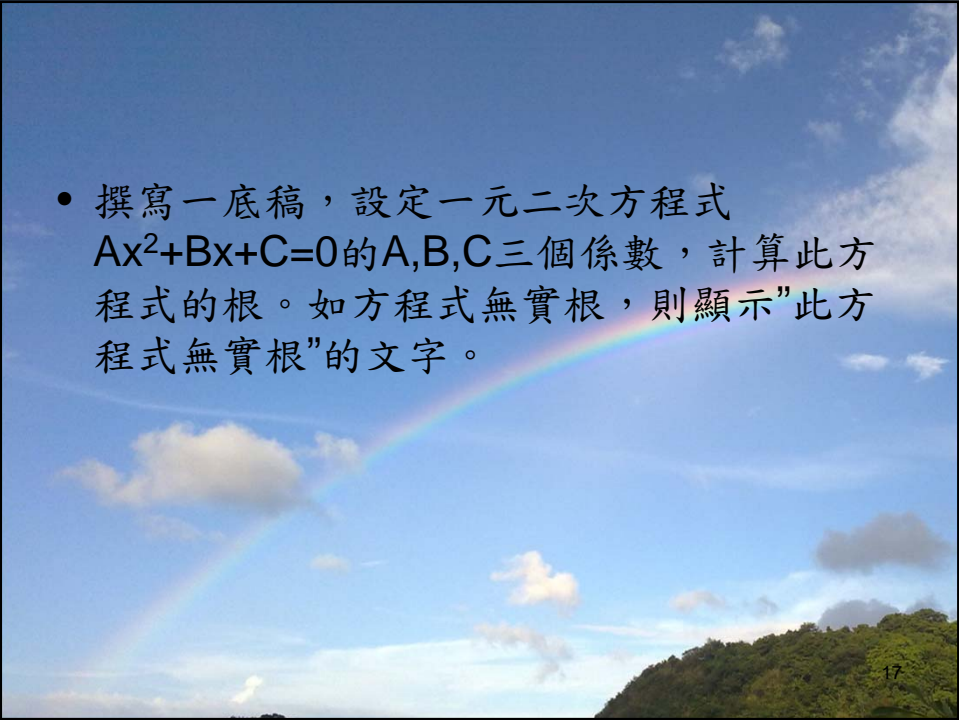
15



```
function [season]=num2season(num)
if(num==1 | num==2 | num==12)
    season='winter';
elseif(num>=3 & num<=5)
    season='spring';
elseif(num>=6 & num<=8)
    season='summer';
elseif(num>=9 & num<=11)
    season='autumn';
else
    season='not month'
end
```

```
number>6
month 6 is summer
>>
```

```
clear;clc
num=input('number>');
text=num2season(num);
fprintf('month %d is %s \n',num,text)
```

- 
- 撰寫一底稿，設定一元二次方程式 $Ax^2+Bx+C=0$ 的A,B,C三個係數，計算此方程式的根。如方程式無實根，則顯示“此方程式無實根”的文字。

使用switch-case-otherwise指令

指令	說明
<pre>switch 運算式 case 選擇值1 敘述主體 1 case 選擇值 2 敘述主體 2 otherwise 敘述主體 n end</pre>	<p>若運算式的值等於選擇值1，則執行敘述主體1，若運算式的值等於選擇值2，則執行敘述主體2，以此類推。如果運算式的值皆不等於所列的選擇值，則執行敘述主體n</p> <p>若接在case後面的選擇值不只一個時，可用大括號將它們括起來，如 {選擇值1, 選擇值2, ..., 選擇值n}</p>

18

switch指令的應用

```
function func9_4(method)
switch method
case {'linear','bilinear'}
    disp('linear/bilinear
method')
case 'cubic'
    disp('Cubic method')
otherwise
    disp('Unknown method')
end
```

```
>> func9_4('bilinear')
linear/bilinear method
```

```
>> func9_4('newton')
Unknown method
```

```
function test=func9_5(num)
n=mod(num,2);
switch n
case 1
    test=logical(0);
case 0
    test=logical(1);
otherwise
    disp('not a positive integer')
    test=logical(0);
end
```

```
>> func9_5(12)
ans =
    1
```

```
>> func9_5(11)
ans =
    0
```

```
>> func9_5(12.22)
not a positive integer
ans =
    0
```

19

- 試寫一函數，可接收月份的數值，然後以 switch 敘述判斷其所屬的季節，春季：3-5月，夏季：6-8月，秋季：9-11月，冬季：12-2月。

20

```
function [season2]=num2season2(num)
```

```
switch num
```

```
case {12,1,2}
```

```
    season2='winter';
```

```
case {3,4,5}
```

```
    season2='spring';
```

```
case {6,7,8}
```

```
    season2='summer';
```

```
case {9,10,11}
```

```
    season2='autumn';
```

```
otherwise
```

```
    season2='not month';
```

```
end
```

```
clear;clc
```

```
num=input('number>');
```

```
text=num2season2(num);
```

```
fprintf('month %d is %s \n',num,text)
```

```
number>7
```

```
month 7 is summer
```

21