

# GrADS

## .gs檔內的變數、迴圈及判斷式

- 範例檔：
- `/home/teachers/weitingc/lecture_ex/grads/24plots.gs`
- `/home/teachers/weitingc/lecture_ex/grads/24plot_loop.gs`

# 如果在gs檔內有重複產製相似圖片的需求...

- 24plots.gs:

開啟GFS預報檔之後，畫出 $t=1\sim t=24$ 每筆時間的地面降水與風場，一筆時間存一張圖

(測試時，請先將sdfopen遠端網址內的日期改為最近的預報初始時間)

```
'reinit'
```

```
...
```

```
'sdfopen https://nomads.ncep ... /gfs_1p00_00z'
```

```
...
```

```
'set t 1'
```

```
...
```

```
'draw title 2019/10/27 00Z'
```

```
'printim d01.png'
```

```
...
```

```
...
```

```
'set t 2'
```

```
...
```

```
'draw title 2019/10/27 03Z'
```

```
'printim d02.png'
```

```
...
```

```
'set t 24'
```

```
...
```

```
'draw title 2019/10/29 21Z'
```

```
'printim d24.png'
```

# 先觀察哪些指令是會隨時間設並不同而變動

```
'reinit'
```

```
...
```

```
'sdfopen https://nomads.ncep ... /gfs_1p00_00z'
```

```
...
```

```
'set t 1'
```

```
'set gxout shaded'
```

```
'd pratesfc*86400'
```

```
'cbar'
```

```
'set gxout vector'
```

```
'set ccolor 15'
```

```
'd ugrd10m;vgrd10m'
```

```
'draw title 2019/10/27 00Z'
```

```
'printim d01.png'
```

```
...
```

```
...
```

```
'set t 2'
```

```
...
```

```
'draw title 2019/10/27 03Z'
```

```
'printim d02.png'
```

```
...
```

```
'set t 24'
```

```
...
```

```
'draw title 2019/10/29 21Z'
```

```
'printim d24.png'
```

# 改寫成.gs檔內的“迴圈”

- 24plots\_loop.gs:

使用while搭配內部變數的運算構成“迴圈”，畫出t=1~t=24每筆時間的地面降水與風場並存檔，結果與24plots.gs完全相同

```
...  
'sdfopen https://nomads.ncep ... /gfs_1p00_00z'  
...  
i=1  
while i<=24  
  'c'  
  'set t 'i  
...  
  'draw title 'date  
  if i<10  
    'printim d0'i'.png'  
...  
  i=i+1  
endwhile
```

```
...  
i=1  
while i<=24  
  'c'  
  'set t 'i  
  'set gxout shaded'  
  'd pratesfc*86400'  
  'cbar'  
  'set gxout vector'  
  'set ccolor 15'  
  'd ugrd10m;vgrd10m\  
  'q time'  
  date=subwrd(result,3)  
  say date  
  'draw title 'date  
  if i<10  
    'printim d0'i'.png'  
  else  
    'printim d'i'.png'  
  endif  
  i=i+1  
endwhile
```

- 用 `'...'` 標記起來的是 `.gs` 檔的內部變數、`while`、`if...` 等內部語法，只能在 `.gs` 檔內使用，不需要用 `'...'` 夾起
- 在 GrADS 指令模式可以下的指令，對於 `.gs` 檔來說是外部指令，必須用 `'...'`
- 內部變數可以接在外部指令之後 (`'set t 'i`)，或是安插在中間 (`'printim d0'i'.png'`)，根據內部變數被賦值的情況來執行外部指令  
(注意: 外部指令如果含有空格，必須被放在 `'...'` 內)

# .gs檔的"迴圈" (while搭配變數運算)

(24plots\_loop.gs)

```
...  
i=1  
while i<=24  
  'c'  
  'set t 'i  
  'set gxout shaded'  
...  
  if i<10  
    'printim d0 'i'.png'  
  else  
    'printim d'i'.png'  
  endif  
  i=i+1  
endwhile
```

重複直到 i 超過 24

- 設定內部變數 i
- 搭配while...endwhile 讓整段指令反覆執行
- 也可利用內部變數i 設定資料時間或存檔的檔名
- 必須讓內部變數 i 在 while內運算累加，才能有迴圈counter的功能！

# .gs檔的判斷式

(24plots\_loop.gs)

```
...  
i=1  
while i<=24  
  'c'  
  'set t 'i  
  'set gxout shaded'  
  ...  
  if i<10  
    'printim d0 'i' .png'  
  else  
    'printim d'i' .png'  
  endif  
  i=i+1  
endwhile
```

根據  
i  
值決定檔名

- if ...else...endif  
(沒有提供elseif的功能!)
- 根據內部變數的值，  
選擇性執行某段指令
- 產生圖檔  
d01.png, d02.png,...,  
d10.png, d11.png,...

# .gs檔的內部函數

- 有一些可以在.gsd檔中使用的內部函數:
- **subwrd(result, n)**  
取得外部指令執行時會顯示的字串，並以空格分割，只取出第n個欄位的結果
- **say**  
將內部變數的值顯示在GrADS環境的指令視窗
- 24plots\_loop.gsd示範如何將這些指令搭配起來，把資料每筆時間點所對應的實際日期與時間，寫在圖片的標題
- 這些內部函數不能在GrADS指令視窗使用，只能寫在.gsd檔中



# 將資料時間擷取為內部變數，在圖片標題顯示

(24plots\_loop.gs)

```
...  
i=1  
while i<=24  
  'c'  
  'set t `i`  
  ...  
  'q time'  
  date=subwrd(result,3)  
  say date  
  'draw title `date`  
  ...  
i=i+1  
Endwhile
```

- 外部指令 `'q time'`，如果在指令視窗執行，在 `set t 1` 的時候會顯示 `Time = 00Z27OCT2019 to 00Z27OCT2019 Sun to Sun`
- 取出上面這個字串的第3個欄位(即資料時間)，對內部變數 `date` 賦值
- 將內部變數 `date` 的值顯示在視窗(執行 `.gs` 檔的過程中可以確認進度)
- 將內部變數 `date` 的值顯示在圖片的標題

GrADS .gs檔內部程式語言的介紹

<http://cola.gmu.edu/grads/gadoc/script.html>