



Sử dụng biến có chỉ số trong Fortran

Bởi:

PGS. TS. NGUYỄN Phạm Văn Huân

Trong chương 2, mục 2.3 đã xét cách khai báo kiểu biến có chỉ số và khái niệm mảng trong Fortran, nêu một số đặc điểm về lưu giữ đối với các biến có chỉ số hay gọi là biến mảng.

Chương này sẽ cung cấp thêm phương pháp lưu giữ và xử lý những nhóm giá trị mà không cần cung cấp tên một cách tường minh cho từng giá trị đó. Trong thực tế, ta thường xử lý một nhóm các giá trị ít nhiều liên hệ hoặc hoàn toàn không liên hệ với nhau. Trong trường hợp này, nếu sử dụng biến mảng, cả nhóm dữ liệu sẽ có một tên chung, nhưng những giá trị riêng biệt có chỉ số riêng duy nhất. Kỹ thuật này cho phép ta phân tích dữ liệu sử dụng các vòng lặp một cách thuận tiện. Trong các mục dưới đây sẽ bổ sung thêm những cấu trúc, những lệnh của Fortran cho phép thao tác thuận lợi với các biến mảng, kỹ thuật đọc dữ liệu từ file để gán vào các biến mảng v.v...

Mảng là yếu tố quan trọng và mạnh mẽ nhất của Fortran. Nếu so sánh với một số ngôn ngữ lập trình khác, thí dụ như Pascal, ta thấy trong Fortran cho phép khai báo những mảng dữ liệu rất lớn và thao tác rất mềm dẻo. Nhiều khi khả năng khai báo mảng dữ liệu lớn làm cho thuật giải của chương trình xử lý trở nên đơn giản. Ngoài ra, sử dụng mảng đúng đắn và thành thạo sẽ giúp chúng ta viết những chương trình hoặc những đoạn chương trình rất ngắn gọn.

Mảng một chiều

Trong lập trình, mảng một chiều thường dùng để biểu diễn một dòng hoặc một cột dữ liệu.

Về phương diện ngôn ngữ, một mảng là *một nhóm địa chỉ lưu giữ* trong bộ nhớ máy tính có cùng tên. Từng thành phần của mảng được gọi là phần tử mảng và được phân biệt với phần tử khác bởi tên chung kèm theo chỉ số trong cặp dấu ngoặc. Những chỉ số được biểu diễn bằng những số nguyên liên tiếp nhau, thường là bắt đầu (chỉ số đầu) bằng số nguyên 1. Những trường hợp dùng chỉ số đầu khác 1 thường liên quan tới tính thuận tiện thao tác các công thức toán học hoặc phương diện thực tiễn. Thí dụ muốn biểu diễn các

Sử dụng biến có chỉ số trong Fortran

hệ số a của phương trình hồi quy nhiều biến liên hệ giữa đại lượng y và các đại lượng x_1, x_2, \dots, x_m

$$y = a_0 + a_1x_1 + a_2x_2 + \dots + a_mx_m$$

ta có thể dùng mảng một chiều với tên A để chỉ tất cả các hệ số, kể cả hệ số tự do, của phương trình hồi quy này và khai báo như sau:

```
REAL A (0 : 20)
```

Trong trường hợp này phần tử thứ nhất A(0) của mảng A biểu diễn hệ số a_0 . Như vậy rất thuận tiện trong khi sử dụng các công thức của đại số.

Nếu ta có tập hợp số liệu về lượng mưa năm trong thế kỷ này tại một trạm khí tượng nào đó, ta có thể dùng mảng

```
REAL RAIN (1900 : 2000)
```

Trong trường hợp này, nếu muốn truy cập lượng mưa năm 1985, ta chỉ định phần tử mảng RAIN (1985).

Để đọc dữ liệu vào một mảng một chiều từ bàn phím hoặc từ file dữ liệu, ta sử dụng lệnh READ. Nếu muốn đọc toàn bộ mảng, ta dùng tên mảng không có các chỉ số. Ta cũng có thể chỉ định những phần tử cụ thể trong lệnh READ, thí dụ

```
READ *, B
```

```
READ *, B(1), B(2), B(3)
```

Cần chú ý rằng, trong thí dụ này, nếu mảng B theo khai báo chứa 3 phần tử thì hai lệnh READ trên tương đương nhau. Nhưng nếu mảng B chứa 8 phần tử thì có sự khác nhau quan trọng giữa hai lệnh READ trên đây, là vì: lệnh thứ nhất đọc vào toàn bộ 8 phần tử của mảng B, trong khi lệnh thứ hai chỉ đọc các giá trị của ba phần tử đầu tiên.

Các giá trị của biến mảng còn có thể đọc với vòng lặp **DO** ẩn. Thí dụ, nếu muốn đọc 5 phần tử đầu tiên của mảng B ta sử dụng lệnh READ như sau

```
READ *, (B (I) , I = 1 , 5)
```

Trong lệnh này, chúng ta thấy không có mặt từ khóa DO, chỉ có chỉ số I của biến mảng B biến thiên từ 1 tới 5 với gia số bằng 1. Như vậy với một lệnh READ máy đọc được liên tục 5 phần tử của mảng B.

Thí dụ 17: Một tập hợp 50 số liệu lượng mưa năm được lưu trong file dữ liệu, mỗi số liệu một dòng. Giả sử đơn vị file là 9. Viết nhóm lệnh đọc những số liệu này vào mảng LMUA.

Cách 1: Dùng lệnh READ đọc từng số, nhưng vòng lặp thực hiện 50 lần và đọc toàn bộ mảng:

	REAL LMUA (50)
	DO 10 I = 1 , 50
	READ (9, *) LMUA (I)
10	CONTINUE

Cách 2: Dùng lệnh READ không chứa chỉ số, nó sẽ đọc toàn bộ mảng, tức đọc liền 50 phần tử:

```
REAL LMUA (50)
```

```
READ (9, *) LMUA
```

Cách 3: Lệnh READ chứa vòng lặp ẩn:

```
REAL LMUA (50)
```

```
READ (9, *) (LMUA (I), I = 1, 50)
```

Lệnh DATA

Lệnh DATA là lệnh đặc tả, thuộc loại lệnh không thực hiện. Nó dùng để khởi tạo giá trị ban đầu cho các biến đơn và các mảng. Dạng tổng quát của lệnh DATA như sau

DATA Danh sách tên biến / Danh sách hằng /

Theo lệnh này các giá trị dữ liệu trong *danh sách hằng* nằm trong hai dấu gạch chéo được gán cho các biến trong *danh sách tên biến* theo tuần tự. Kiểu của các giá trị dữ liệu cũng nên phù hợp kiểu của các biến, sao cho máy tính không phải chuyển đổi. Các lệnh DATA phải đặt trước các lệnh thực hiện, tức ở gần đầu chương trình, ngay sau những lệnh mô tả kiểu như lệnh REAL, INTEGER, LOGICAL, DIMENSION...

Thí dụ, lệnh

```
DATA A , B, C , I / 0.0 , 32.75 , ?2.5 , 10 /
```

Sử dụng biến có chỉ số trong Fortran

sẽ khởi tạo giá trị 0,0 cho biến A, 32,75 cho biến B, 2,5 cho biến C và 10 cho biến I.

Chú ý rằng lệnh DATA chỉ khởi tạo giá trị ở đầu chương trình. Lệnh DATA không thể sử dụng trong vòng lặp để tái tạo giá trị các biến. Nếu cần tái tạo các biến, ta phải sử dụng các lệnh gán. Lệnh DATA cũng không thể nằm trong chương trình con.

Nếu các giá trị lặp lại trong danh sách hằng, ta có thể dùng cách viết lệnh DATA ngắn gọn. Thí dụ, nếu muốn khởi tạo giá trị 1 cho các biến I, J, K và giá trị 0,5 cho các biến X, Y, Z, thì hai lệnh sau đây tương đương nhau:

```
DATA I, J, K, X, Y, Z / 1, 1, 1, 0.5, 0.5, 0.5 /
```

```
DATA I, J, K, X, Y, Z / 3*1, 3*0.5 /
```

Lệnh DATA có thể sử dụng để khởi tạo một hoặc một số phần tử của mảng. Thí dụ, các lệnh sau khởi tạo tất cả các phần tử của mảng J và TIME:

```
INTEGER J (5)
```

```
REAL TIME (4)
```

```
DATA J, TIME / 5*0, 1.0, 2.0, 3.0, 4.0 /
```

Nhóm lệnh

```
REAL HOUR (5)
```

```
DATA HOUR (1) / 10.0 /
```

chỉ khởi tạo một giá trị của phần tử đầu tiên của mảng HOUR, các phần tử từ thứ 2 đến 5 của nó chưa biết.

Có thể sử dụng vòng DO ẩn trong lệnh DATA. Thí dụ:

```
INTEGER Y (100)
```

```
DATA (Y (I), I = 1, 50) / 50*0 /
```

khởi tạo giá trị 0 cho 50 phần tử đầu của mảng Y, 50 phần tử còn lại chưa được khởi tạo.

Mảng hai chiều

Các lệnh mô tả mảng hai chiều giống như với mảng một chiều, khác biệt duy nhất là dùng hai tham số kích thước mảng. Mỗi phần tử mảng được truy cập bởi tên mảng với hai chỉ số nằm trong cặp dấu ngoặc.

Trong thực tế lập trình người ta thường biểu diễn các ma trận, các bảng dữ liệu gồm một số cột, mỗi cột có một số dòng giá trị thành mảng hai chiều.

Thí dụ, ma trận các hệ số đứng trước các ẩn của hệ phương trình đại số tuyến tính $a_{ij}(i = 1..10, j = 1..10)$ thường biểu diễn bằng mảng hai chiều A với lệnh mô tả như sau

```
REAL A(10, 10)
```

Các giá trị quan trắc từng giờ về mực nước biển trong vòng một tháng có thể biểu diễn thành một bảng số liệu gồm 31 dòng, 24 cột. Các dòng tuần tự ứng với các ngày trong tháng. Các cột tuần tự ứng với 24 giờ trong một ngày. Trong Fortran, bảng số liệu này có thể biểu diễn bằng mảng hai chiều

```
REAL SLEV(31, 24)
```

theo cách này, khi thao tác với mực nước tại một ngày, giờ cụ thể nào đó, người ta chỉ cần chỉ định phần tử SLEV(I, J), với chỉ số thứ nhất I chỉ ngày, chỉ số thứ hai J chỉ giờ trong ngày đó. Khi cần tính mực nước trung bình ngày, thí dụ của ngày thứ nhất trong tháng, người ta chỉ cần cộng tất cả các phần tử với chỉ số I = 1:

```
SLEV(1,1) + SLEV(1,2) + ... + SLEV(1,24)
```

Sử dụng các mảng rất tiện lợi khi lập chương trình phân tích, tính toán với những ma trận, những tập số liệu lớn.

Thí dụ 18: Lập ma trận đơn vị (ma trận vuông với các phần tử trên đường chéo chính bằng 1, còn tất cả các phần tử khác bằng 0). Thí dụ ma trận kích thước $n = 3$, tức có 3 dòng và 3 cột, sẽ là

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Đoạn chương trình Fortran thực hiện việc này sẽ như sau:

```
INTEGER IDMAT(3,3)
```

Sử dụng biến có chỉ số trong Fortran

```
DO I = 1, 3
```

```
DO J = 1, 3
```

```
IF (I .EQ. J) THEN
```

```
  IDMAT (I, J) = 1
```

```
ELSE
```

```
  IDMAT (I, J) = 0
```

```
ENDIF
```

```
END DO
```

```
END DO
```

Thí dụ 19: Đọc các giá trị mảng hai chiều từ file dữ liệu. Giả sử có các số liệu về lưu lượng nước trung bình năm của một số con sông. Những số liệu này ghi trong file SONG.LLG. Dòng trên cùng của file ghi hai số nguyên tuần tự chỉ số năm quan trắc và số con sông. Sau đó có n dòng, mỗi dòng số liệu tuần tự ứng với một năm, trong mỗi dòng có m giá trị, mỗi giá trị ứng với một con sông. Ta dùng mảng hai chiều để biểu diễn tập số liệu này, chỉ số thứ nhất của mảng chỉ thứ tự năm, chỉ số thứ hai chỉ thứ tự con sông. Đoạn chương trình sau đây cho phép đọc số liệu từ file, tính lưu lượng trung bình của tất cả các sông và in kết quả lên màn hình.

REAL SLL (100, 15), TB (15)
OPEN (1, FILE = 'SONG.LLG', STATUS = 'OLD')
READ (1, *) N, M
DO I = 1, N
READ (1, *) (SLL (I, J), J = 1, M)
ENDDO
CLOSE (1)
DO J = 1, M
TB (J) = 0.0
DO I = 1, N
TB (J) = TB (J) + SLL (I, J)

	ENDDO
	ENDDO
	PRINT 4, (TB (J) , J = 1 , M)
4	FORMAT (1X, 15 F8.0)

Hãy lưu ý cách đọc số liệu lượng mưa trong chương trình này. Như đã mô tả cách ghi số liệu trong file, lượng mưa được ghi thành n dòng, mỗi dòng ứng với một năm, trên mỗi dòng lại có m giá trị lượng mưa ứng với m con sông. Muốn đọc liên tục số liệu trong n năm ta đã dùng hai vòng DO lồng nhau:

```
DO I = 1 , N
```

```
  READ (1, *) (SLL (I , J) , J = 1 , M)
```

```
END DO
```

trong đó vòng DO bên trong là vòng DO ẩn với chỉ số J chạy từ 1 đến M . Bằng vòng lặp ẩn này ta đã đọc được m giá trị số thực ứng với m sông trên cùng một dòng.

Một cách tổng quát, đây là cách đọc thường dùng nhất để bằng một lệnh đọc có thể nhận liên tiếp tất cả các phần tử trên một hàng của ma trận.

Nếu ta dùng hai vòng lặp thông thường:

```
DO I = 1, N
```

```
  DO J = 1, M
```

```
    READ (1, *) SLL (I, J)
```

```
  END DO
```

```
END DO
```

thì sẽ phạm sai lầm, bởi vì hai vòng DO này tương đương với $n \times m$ lệnh READ, và như ta đã biết, mỗi lần lệnh READ thực hiện xong thì đầu đọc file sẽ xuống dòng mới. Như vậy máy sẽ đọc $n \times m$ dòng trong khi trong file chỉ có n dòng số liệu.

Ta phát triển cách dùng vòng DO ẩn cho trường hợp trên cùng một dòng trong file có hai đại lượng. Thí dụ, cũng là file số liệu như đã mô tả trong thí dụ 19, nhưng trên mỗi dòng ngoài m giá trị lưu lượng còn có m giá trị độ đục ứng với m con sông. Trong trường

Sử dụng biến có chỉ số trong Fortran

hợp này ta khai báo thêm một biến DD (100, 15) và lệnh đọc cả lưu lượng và độ đục sẽ là:

```
DO I = 1 , N
```

```
READ (1, *) (SLL (I , J), J = 1 , M) , (DD (I , J), J = 1, M)
```

```
END DO
```

Trường hợp ở đầu mỗi dòng có ghi năm quan trắc, ta sẽ dùng

```
DO I = 1 , N
```

```
READ (1, *) NAM (I), (SLL (I, J) , J = 1, M) , (DD (I , J) , J = 1, M)
```

```
END DO
```

Mảng nhiều chiều

Fortran cho phép sử dụng các mảng với số chiều tối đa bằng 7. Chúng ta có thể hình dung mảng ba chiều giống như hình hộp chữ nhật tạo bởi nhiều hình lập phương con. Các phần tử của mảng ba chiều giống như những hình lập phương con, xếp thành một số lớp, mỗi lớp có một số hàng và mỗi hàng có một số hình lập phương. Từ đó ta biểu diễn vị trí của một phần tử nào đó như là vị trí của hình lập phương con: thứ tự của nó trong một hàng bằng chỉ số I, thứ tự hàng bằng chỉ số J và thứ tự lớp - chỉ số K.

Thí dụ, mảng ba chiều có thể định nghĩa bằng lệnh:

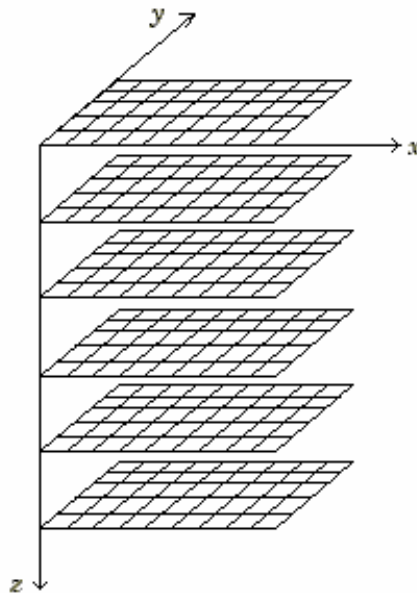
```
REAL T (3, 4, 4)
```

Nếu sử dụng tên mảng ba chiều không có chỉ số, ta xử lý mảng với chỉ số thứ nhất biến thiên nhanh nhất, chỉ số thứ hai biến thiên nhanh thứ hai và chỉ số thứ ba biến thiên chậm nhất. Thí dụ với mảng T, hai lệnh đọc sau đây là tương đương:

```
READ*, T
```

```
READ*, (((T(I, J, K), I=1, 3), J=1, 4), K=1, 4)
```

Tương tự ta hình dung mảng bốn chiều như là một chuỗi các mảng ba chiều...



Biểu diễn mảng ba chiều trong biển

Trong khí tượng thủy văn mảng ba chiều thường được dùng để biểu diễn những số liệu quan trắc trong không gian ba chiều. Thí dụ, ta có thể biểu diễn trường áp suất nước biển tại các điểm nút kinh, vĩ tuyến của một miền hình chữ nhật trên mặt biển và một số tầng sâu. Trong trường hợp này có thể quy ước chỉ số thứ nhất của mảng i biến thiên theo trục x hướng sang phía đông, chỉ số thứ hai j biến thiên theo trục y hướng lên bắc, còn chỉ số thứ ba k biến thiên theo trục z hướng thẳng đứng từ mặt xuống đáy biển để chỉ giá trị áp suất ứng với kinh độ, vĩ độ và một tầng sâu nào đó trong biển (hình 7.1). Trong khí tượng học, hai chỉ số đầu hoàn toàn tương tự, còn chỉ số thứ ba của mảng ba chiều biến thiên theo trục z hướng thẳng đứng từ mặt đất lên trên, có thể dùng để chỉ quan trắc tại một tầng cao.

Theo quy ước đó thì mảng hai chiều là một trường hợp riêng của mảng ba chiều dùng để biểu thị trường yếu tố khí tượng thủy văn nào đó trên một miền phẳng hình chữ nhật, thí dụ trường khí áp mặt đất, trường nhiệt độ nước mặt biển... Mảng bốn chiều có thể dùng để biểu diễn những trường ba chiều vừa mô tả ở trên nhưng tại nhiều thời điểm t khác nhau.

Trong thủy văn, chỉ số thứ nhất của mảng ba chiều thường dùng biểu diễn yếu tố quan trắc tại các độ sâu khác nhau của một mặt cắt, dọc theo sông ta có thể có nhiều mặt cắt được biểu diễn bằng biến thiên của chỉ số thứ hai, yếu tố quan trắc lại có thể biến đổi theo thời gian và được chỉ định bằng chỉ số thứ ba. Nếu xét nhiều sông cùng một lúc, ta cần đến mảng bốn chiều.

Chính là trong khí tượng, hải dương học chúng ta được biết tới những mô hình dự báo thời tiết hay hoàn lưu và nhiệt muối đại dương thường sử dụng các trường ba chiều ban đầu và phát sinh ra những trường bốn chiều với kích thước khổng lồ (do độ phân giải

không gian cao và bước thời gian mô phỏng, dự báo nhỏ) phải lưu trữ và quản lý trong máy tính.

Các mảng có số chiều lớn hơn bốn có thể là khó hình dung trực quan hơn. Tuy nhiên, nếu chúng ta quy ước rõ ràng, nhất quán các chỉ số thứ nhất, thứ hai... tương ứng với biến số nào trong thực tế và nắm vững quy tắc biến thiên chỉ số của mảng thì vẫn có thể truy cập, thao tác đúng với một phần tử bất kỳ của mảng trong chương trình.

Thí dụ 20: Tính tần suất mưa. Số liệu giá trị ngày của các yếu tố khí tượng thủy văn tại trạm Hòn Dấu được lưu trong file HONDAU.MAT có quy cách ghi như sau: Dòng trên cùng ghi tên trạm. Dòng thứ 2 có hai số nguyên viết cách nhau lần lượt chỉ tổng số ngày quan trắc và số yếu tố được quan trắc. Dòng thứ ba có 6 số nguyên viết cách nhau lần lượt chỉ ngày, tháng, năm đầu và ngày, tháng, năm cuối quan trắc. Dòng thứ 4 là tiêu đề cột liệt kê tên tất cả các yếu tố được quan trắc, mỗi tên được ghi với độ rộng 8 vị trí. Các dòng tiếp theo lần lượt ghi giá trị của các yếu tố, mỗi dòng một ngày. Giả sử lượng mưa ngày ghi ở cột số 6. Viết chương trình đọc và tính xem trong suốt thời gian quan trắc có bao nhiêu lần mưa kéo dài 1 ngày, bao nhiêu lần mưa kéo dài 2 ngày liền, bao nhiêu lần mưa kéo dài 3 ngày liền...

	REAL X (5000)
	INTEGER TS (5000)
	OPEN (1, FILE = 'HONDAU.MAT', STATUS = 'OLD')
	READ (1, *)
	READ (1, *) N
	READ (1, *)
	READ (1, *)
	DO I = 1, N
	READ (1, *) (X (I), J = 1, 6)
	END DO
	CLOSE (1)
	TS = 0
	I = 1
100	IF (I .GT. N) GOTO 15
	IF (X (I) .EQ. 0.0) THEN

	I = I + 1
	GOTO 100
	ELSE
	J = 1
300	IF (I .EQ. N .OR. X(I + 1) .EQ. 0.0) THEN
	TS (J) = TS (J) + 1
	I = I + 1
	GOTO 100
	ELSE
	J = J + 1
	I = I + 1
	GOTO 300
	END IF
	END IF
15	I = N
16	IF (TS (I) .EQ. 0) THEN
	I = I - 1
	GOTO 16
	ELSE
	DO N = 1, I
	PRINT '(2I5)', N, TS (N)
	END DO
	END IF
	END

Thí dụ 21: Tính ma trận tương quan của tập số liệu quan trắc các yếu tố khí tượng thủy văn. Với file số liệu trong thí dụ 20, viết chương trình đọc các thông tin cần thiết trong file và in ma trận tương quan của các yếu tố quan trắc lên màn hình.

Ta thấy, một cách tự nhiên mỗi chuỗi giá trị ngày của một yếu tố quan trắc có thể được biểu diễn thành mảng một chiều, chỉ số mảng sẽ biến thiên theo thứ tự ngày quan trắc. Tuy nhiên, ta có thể gộp tất cả các mảng một chiều thành một mảng hai chiều với chỉ số thứ hai biến thiên theo thứ tự yếu tố quan trắc: 1, 2, ... Bằng cách dùng mảng hai chiều X (5000, 15) như trong chương trình dưới đây sẽ rất thuận tiện cho việc sử dụng các vòng lặp DO với tham số đếm của vòng DO đồng thời là chỉ số của mảng.

	REAL X (5000, 15), MX (15), DX (15) , R (15, 15)
	OPEN (1, FILE = 'HONDAU.MAT', STATUS = 'OLD')
	READ (1, *)
	READ (1, *) N, M
	READ (1, *)
	READ (1, *)
	DO I = 1, N
	READ (1, *) (X (I, J), J = 1, M)
	END DO
	CLOSE (1)
C Tính trung bình và độ lệch quân phương của M yếu tố	
	DO I = 1, M
	MX (I) = X (1, I)
	DX (I) = X (1, I)*X (1, I)
	DO J = 2, N
	MX (I) = MX (I) + X (J, I)
	DX (I) = DX (I) + X (J, I) * X(J, I)
	END DO
	MX (I) = MX (I) / N
	DX (I) = SQRT (DX (I) / N - MX (I) * MX (I))
	END DO

C Tính ma trận tương quan	
	DO I = 1, M - 1
	DO J = I + 1, M
	R (I, J) = 0.0
	DO K = 1, N
	R (I, J) = R (I, J) + X (K, I) * X (K, J)
	END DO
	R (I, J) = R (I, J) / N - MX (I) * MX (J)
	R (I, J) = R (I, J) / (DX (I) * DX (J))
	END DO
	END DO
	DO I = 1, M
	R (I, I) = 1.0
	END DO
	DO I = 1, M
	PRINT 4, (R (K, I), K = 1, I - 1), (R (I, J), J = I, M)
	END DO
4	FORMAT (<M>F6.2)
	END

Những điều cần chú ý khi sử dụng các mảng

Trong các mục trước của chương này ta đã học sử dụng một mảng - một nhóm các địa chỉ lưu giữ các giá trị có một tên chung, nhưng phân biệt với nhau bởi một hoặc một số chỉ số. Mảng là một yếu tố mạnh mẽ nhất trong Fortran, vì nó cho phép lưu giữ một tập hợp dữ liệu lớn để xử lý trong chương trình của chúng ta.

Mặc dù với tiện lợi cơ bản như trên, các mảng cũng thường có thể gây ra những lỗi mới. Một khi bạn dự định sử dụng mảng để mô tả dữ liệu, hãy tự hỏi “ta có cần sử dụng dữ liệu này nhiều lần không” và “dữ liệu này có cần phải lưu trước khi ta sử dụng nó

không”. Nếu câu trả lời cho các câu hỏi trên là “không”, nên hạn chế dùng mảng, mà dùng các biến đơn.

Một khi mảng là cần thiết, nhưng chương trình làm việc sai, trước hết hãy kiểm tra những điều sau đây:

? Kích thước mảng: Mô tả mảng phải chỉ ra số phần tử tối đa dự định lưu giữ trong mảng. Mặc dù chúng ta không nhất thiết phải dùng hết tất cả các phần tử của mảng, nhưng chúng ta không được sử dụng nhiều phần tử hơn so với số phần tử đã mô tả ở phần khai báo của chương trình. Vậy với mỗi bài toán cụ thể nếu cần sử dụng mảng, hãy hình dung trước kích thước tối đa của mỗi chiều của mảng để khai báo cho đúng, có thể hơi dư ra một ít, nhưng dư nhiều quá sẽ tốn bộ nhớ, còn khai báo thiếu thì khi chạy chương trình sẽ phát sinh lỗi logic.

? Chỉ số mảng: Hãy kiểm tra từng chỉ số, đặc biệt những chỉ số là biểu thức số học, để tin chắc rằng nó là số nguyên nằm trong giới hạn đúng đắn, không vượt ra ngoài khoảng biến thiên của chỉ số. Nếu chỉ số mảng vượt ra ngoài giới hạn cho phép thì hãy xem các biến trong biểu thức số học tính chỉ số có bị nhầm không. Có thể dùng lệnh in lên màn hình để theo dõi diễn biến của chỉ số.

? Vòng lặp DO: Nếu bạn dùng chỉ số mảng làm tham số đếm của vòng lặp DO, hãy tin chắc rằng bạn đã sử dụng đúng tên biến trong chương trình của bạn. Thí dụ, nếu trong một vòng lặp DO với mảng ba chiều bạn định cho chỉ số thứ ba của mảng (K) biến thiên, hãy kiểm tra xem bạn có dùng I thay vì K không. Lỗi thường gặp là chỉ số đảo ngược: Hãy tự hỏi chỗ này cần B (K, L) hay B (L, K)? Hãy có ý thức về đặt tên cho chỉ số. Tập quán chung là sử dụng biến I cho chỉ số thứ nhất, J - thứ hai và K - thứ ba; như vậy rất tiện lợi khi sử dụng các vòng lặp lồng nhau và chỉ số đếm của vòng lặp DO đồng thời là chỉ số mảng.

Bài tập

1. File dữ liệu với đơn vị file 9 chứa 28 số liệu lượng mưa ngày trong bốn tuần lễ liên tiếp, ghi thành 4 dòng, mỗi tuần một dòng. Nhóm lệnh sau

```
REAL DMUA (28)
```

```
DO I = 1, 28
```

```
READ (9, *) DMUA (I)
```

```
END DO
```

có đọc đúng các số liệu lượng mưa ứng với từng ngày không. Nếu không đúng, chỉ ra phương án đọc đúng.

Sử dụng biến có chỉ số trong Fortran

2. Viết chương trình cho phép đọc từ bàn phím ba số nguyên, kiểm tra xem ba số nguyên đó có thể chỉ ngày, tháng, năm hợp lý không. Kết quả kiểm tra ghi thành dòng thông báo thích hợp lên màn hình.

3. Viết chương trình đọc một chuỗi Y gồm 20 giá trị thực từ file EXPER, trong đó mỗi giá trị ghi trên một dòng. Lập một chuỗi Z gồm 20 giá trị thỏa mãn các điều kiện:

$$Z_1 = Y_1; Z_{20} = Y_{20}; Z_i = \frac{Y_{i-1} + Y_i + Y_{i+1}}{3} (i = 2..19)$$

In chuỗi xuất phát và chuỗi mới cạnh nhau thành bảng hai cột.

4. Viết chương trình đọc file RAIN chứa bảng dữ liệu lượng mưa gồm 12 dòng (mỗi dòng tương ứng một tháng) và 5 cột (mỗi cột tương ứng một năm trong các năm 1978-1982). Xác định và in bảng thông tin sau đây:

LUONG MUA TRUNG BINH

1978 - XXX.XX

1979 - XXX.XX

1980 - XXX.XX

1981 - XXX.XX

1982 - XXX.XX

LUONG MUA CUC DAI

THANG XX NAM XXXX

LUONG MUA CUC TIEU

THANG XX NAM XXXX

5. File dữ liệu tên là SCS1.TEM ghi số liệu về trường nhiệt độ nước biển trung bình tháng 1 ở vùng biển Đông có quy cách ghi như sau:

- Dòng thứ nhất gồm tuần tự các tham số: kinh tuyến biên phía tây, kinh tuyến biên phía đông, vĩ tuyến biên phía nam, vĩ tuyến biên phía bắc (các số thực) của vùng, bước lưới theo phương tây đông, bước lưới theo phương bắc nam (đo bằng phút, các số nguyên) của lưới.

Sử dụng biến có chỉ số trong Fortran

- Dòng thứ hai ghi kích thước của ma trận số liệu (các số nguyên) theo dòng (phương bắc nam), theo cột (phương tây đông), theo chiều sâu từ mặt biển xuống dưới và một số nguyên ?32767 chỉ giá trị khuyết của số liệu nhiệt độ.

- Phần còn lại gồm: một số nguyên chỉ tầng sâu quan trắc (mét) ghi ở một dòng; sau đó là mảng số liệu nhiệt độ ứng với tầng đó ghi thành các dòng từ bắc xuống nam, các cột từ tây sang đông (các số thực không dính nhau). Tiếp tục như vậy cho đến tầng sâu dưới cùng.

Hãy viết chương trình đọc dữ liệu, chọn ra một profil nhiệt độ cho điểm bất kỳ thuộc miền tính. Kết quả ghi lên màn hình như sau:

KINH DO XXX.XX

VI DO XX.XX

TANG (m) NHIET DO

XXXX XX.XX

XXXX XX.XX

.....

6. Cho file dữ liệu SCS1.TEM đã mô tả trong bài tập 5. Hãy viết chương trình đọc dữ liệu và tính các giá trị nhiệt độ trung bình của từng tầng quan trắc và giá trị nhiệt độ trung bình toàn biển kể từ tầng mặt cho tới tầng quan trắc dưới cùng.

7. Cho file dữ liệu SCS1.TEM đã mô tả trong bài tập 5. Hãy viết chương trình đọc dữ liệu và in ra file SECT17.TEM một bảng số liệu nhiệt độ nước của mặt cắt dọc vĩ tuyến 17°N với quy cách như sau:

- Dòng trên cùng là tiêu đề:

"Phân bố nhiệt độ nước trên mặt cắt dọc vĩ tuyến 17".

- Dòng thứ hai liệt kê các kinh độ từ tây sang đông.

- Các dòng tiếp dưới ghi độ sâu tầng quan trắc ở mỗi đầu dòng tương ứng, sau đó là các giá trị nhiệt độ nước (lấy đến hai chữ số thập phân) ghi thẳng cột với những kinh độ tương ứng đã liệt kê ở dòng thứ hai. Những giá trị khuyết (?32767) ghi bằng số 99.99 hoặc năm dấu hoa thị (*****).