



# Nhập và xuất dữ liệu đơn giản

Bởi:

PGS. TS. NGUYỄN Phạm Văn Huân

## Các lệnh xuất và nhập dữ liệu

Máy tính có thể nhập dữ liệu từ các nguồn, các thiết bị khác nhau. Tương tự, ta cũng có thể hướng sự xuất dữ liệu ra các thiết bị khác nhau. Trong chương này, ta xét cách nhập dữ liệu từ bàn phím và xuất dữ liệu ra màn hình hoặc máy in. Việc xuất và nhập dữ liệu có dùng các file dữ liệu sẽ được xét trong chương 6.

*Lệnh xuất dữ liệu định hướng ra màn hình:*

### **PRINT \* , Danh sách các mục in**

*Lệnh nhập dữ liệu từ bàn phím:*

### **READ \* , Danh sách các biến**

*Các mục in* trong lệnh in có thể là một hằng, một biến, một biểu thức. Nếu trong danh sách các mục in có từ hai mục trở lên, thì các mục phải cách nhau bởi dấu phẩy. Trong danh sách các biến của lệnh nhập (đọc) dữ liệu, nếu có hơn một biến cần đọc dữ liệu, thì những biến đó phải được liệt kê cách nhau bởi dấu phẩy. Các mục được in ra trên một dòng màn hình theo thứ tự được liệt kê trong danh sách. Nếu trong danh sách không có một mục in nào, thì máy tính chỉ đơn giản là xuống một dòng trên màn hình. Thí dụ, xét đoạn chương trình sau đây:

```
GOC = 30.0
```

```
PRINT* , ' Khi X = ', GOC , ' 1/2 SinX = ', 0.5 * SIN (GOC *
```

```
* 3.141593 / 180.)
```

**Ghi chú:** Trong lệnh PRINT vừa rồi có một dấu nói dòng. Ở đây đã dùng dấu hoa thị bên trong hình nhũ nhật nhỏ \* để phân biệt với dấu hoa thị bình thường là ký hiệu của

*phép tính nhân. Từ nay về sau trong sách này ở những dòng lệnh nào có dấu nối dòng sẽ quy ước dùng ký hiệu này. Còn khi soạn chương trình trên màn hình máy tính, thì như đã nói trong mục 1.5, chương 1, chỉ cần viết dấu hoa thị vào vị trí thứ 6 của dòng lệnh.*

Ta thấy trong danh sách các mục in của lệnh PRINT có 4 mục liệt kê theo thứ tự là:

- 1) Cụm chữ ' Khi X ='
  - 2) Biến có tên là GOC lưu giá trị  $30^\circ$
  - 3) Cụm chữ '  $1/2 \sin X =$ '
  - 4) Biểu thức
- $0.5 * \sin (GOC * 3.141593 / 180.0)$

biểu thị nửa sin của góc  $30^\circ$  đã đổi thành radian.

Như vậy, mục in thứ nhất và thứ ba là những hằng văn bản, mục in thứ 2 là giá trị của biến số thực GOC và mục in thứ tư là một biểu thức số thực. Trước khi in mục thứ tư, máy tính phải tính giá trị của biểu thức này (bằng 0,25), rồi sau đó mới in giá trị đó lên màn hình. Kết quả trên màn hình sẽ như sau:

Khi X = 30.00000  $1/2 \sin X = 0.2500000$

Hãy chú ý rằng với lệnh PRINT \* trên đây các mục in là những cụm dữ liệu văn bản được in ra đúng như ta nhìn thấy trong dòng lệnh, từng ký tự một, kể cả dấu trống. Các giá trị của biến và biểu thức thực được in ra sau một khoảng trống và số những chữ số có nghĩa sau dấu chấm thập phân khác nhau. Nếu giá trị của các biến là những số khá nhỏ hoặc khá lớn, thì máy sẽ in ra những giá trị đó dưới dạng biểu diễn E hoặc D (xem mục 2.1). Kiểu in dữ liệu như trên gọi là in không được định dạng hay in không có format.

Chú ý rằng, trong READ \*, sau dấu phẩy là danh sách các biến, khi thực hiện lệnh này, máy tính chờ ta gõ từ bàn phím những giá trị (các ký tự văn bản, số nguyên, số thực...) tương xứng về kiểu với danh sách biến, mỗi giá trị cách nhau một dấu phẩy hay ít nhất một dấu trống, riêng những ký tự văn bản phải nằm trong cặp dấu nháy trên ( ' '). Kết thúc danh sách các giá trị phải gõ lệnh phím Enter (?). Máy tính sẽ tuân tự gán những giá trị nhận từ bàn phím vào những biến tương ứng trong danh sách biến của lệnh READ. Nếu ta gõ chưa đủ số giá trị theo danh sách biến, thì máy chờ ta gõ cho đến khi đủ các giá trị mới kết thúc thực hiện lệnh READ. Nếu kiểu dữ liệu gõ vào sai so với kiểu dữ liệu của biến, thì lập tức chương trình ngừng thực hiện và báo lỗi chạy chương trình. Thí dụ lệnh

READ \* , I , NAM , TEMP , GHICHU

Nhập và xuất dữ liệu đơn giản

đòi hỏi ta gõ vào từ bàn phím một số nguyên cho biến I, một số nguyên nữa cho biến NAM và một số thực cho biến TEMP, một xâu ký tự cho biến văn bản GHICHU, muốn nhập đúng yêu cầu ta có thể gõ vào bàn phím như sau:

1 1982 25.36 'SL quan trac' ? hay 1,1982,25.36,'SL quan trac' ?

*Lệnh in có quy cách (có định dạng):*

### **PRINT k , Danh sách các mục in**

Cũng giống như lệnh in không định dạng, danh sách các mục in chỉ ra những hằng, biến hay các biểu thức cần in theo thứ tự liệt kê. Tham số *k* nguyên dương chỉ tới nhãn của lệnh FORMAT mô tả quy cách in thông tin ra màn hình như vị trí in, khoảng cách giữa các mục in, số chữ số thập phân cần in đối với giá trị số thực... Dạng tổng quát của lệnh FORMAT như sau:

k FORMAT (Danh sách các đặc tả)

trong đó *k* là *nhãn* của dòng lệnh FORMAT. Danh sách các *đặc tả* nằm trong cặp dấu ngoặc đơn báo cho máy tính biết về cách dẫn dòng theo chiều thẳng đứng và bố trí các ký tự trong dòng thông tin in ra. Nếu in ra trên giấy máy in, thì *tùy chọn dẫn dòng* báo cho máy in điều khiển kéo giấy để in sang đầu trang mới, xuống dòng mới, xuống hai dòng mới hay in ngay trên dòng hiện thời... Máy tính sẽ thiết lập mỗi dòng in bên trong bộ nhớ trước khi thực sự in dòng đó lên giấy. Vùng bộ nhớ bên trong đó gọi là *vùng đệmbuffer*. Những ký tự đầu tiên trong vùng buffer gọi là ký tự điều khiển kéo giấy của máy in có những ý nghĩa như sau:

1 Sang trang mới

Dấu trống Xuống một dòng

0 Xuống hai dòng

+ Không xuống dòng

Bây giờ ta làm quen với những đặc tả đơn giản sau đây. Trong chương 4 sẽ còn trở lại vấn đề định dạng phức tạp hơn khi làm việc với các file dữ liệu.

### **Các đặc tả trong lệnh FORMAT**

1) *Các đặc tả văn bản* thường dùng để xuất dữ liệu là những ký tự, các đoạn văn bản, hay dùng in tiêu đề các báo cáo. Đặc tả văn bản cho phép đưa các ký tự trực tiếp vào buffer. Các ký tự phải nằm trong cặp dấu nháy trên hay dấu ngoặc kép. Thí dụ:

Nhập và xuất dữ liệu đơn giản

PRINT 4

4 FORMAT ('1' , 'KET QUA THI NGHIEM')

Ta cũng có thể dùng *đặc tả* **H** trong đó  $w$  – số vị trí để xuất dữ liệu văn bản. Thí dụ

PRINT 5 , Y

5 FORMAT (16H TICH PHAN BANG , F9.3)

2) *Đặc tả* **X** sẽ chèn  $n$  dấu trống vào bản ghi, thường dùng để căn giữa các tiêu đề báo cáo, thí dụ:

PRINT 35

35 FORMAT ('1' , 25X , 'THI NGHIEM SO 1')

3) *Đặc tả* **A** $w$  dùng cho các hằng và biến xâu ký tự, các thông tin văn bản, tùy chọn  $w$  báo cho máy tính số vị trí giành cho một biến xâu ký tự (văn bản) cần in. Mục văn bản in ra căn lề bên phải.

4) *Đặc tả* **I** $w$  dùng cho số nguyên, trong đó  $w$  số vị trí dùng để in giá trị số nguyên.

5) *Đặc tả* **F** $w.d$  dùng biểu diễn dạng thập phân của số thực,  $w$  – tổng số vị trí dành cho số thực kể cả dấu chấm thập phân,  $d$  – số chữ số thập phân sau dấu chấm. Trong hai đặc tả **I**  $w$  và **F**  $w.d$  các số in ra được căn lề bên phải. Nếu đặc tả thiếu vị trí để biểu diễn giá trị, thì giá trị số sẽ không được in ra, mà tại các vị trí in sẽ xuất hiện các dấu sao (\*) để báo hiệu cho ta biết rằng đặc tả của lệnh FORMAT không phù hợp, cấp thiếu vị trí so với giá trị của đại lượng cần in.

6) *Đặc tả* **E** $w.d$  dùng ghi ra dưới dạng lũy thừa những giá trị rất lớn hoặc rất nhỏ và khi ta chưa hình dung rõ về độ lớn của đại lượng.

Thông thường hai lệnh PRINT và FORMAT đi kèm gần nhau. Thí dụ:

PRINT 5 , I, NAM, TEMP, GHICHU

5 FORMAT (1X, I3, I8, F10.2, 1X, A20)

Sau lệnh READ và các dữ liệu được gõ vào từ bàn phím đã nói trong mục 3.2.1, thì kết quả cập lệnh in này trên màn hình sẽ như sau:

**1 1982 25.36 SL quan trac**

Nhập và xuất dữ liệu đơn giản

7) Các đặc tả  $A_w$ ,  $I_w$ ,  $F_w.d$  và  $E_w.d$  cũng dùng với lệnh đọc số liệu.

8) Đối với các giá trị logic trong Fortran dùng đặc tả  $L_w$ , trong đó  $w$  – số vị trí giành cho dữ liệu. Thí dụ theo lệnh

16 FORMAT (L5)

nếu tại một trong 5 vị trí giành cho biến có chữ **T**, thì giá trị **.TRUE.** sẽ được gán vào biến logic trong lệnh đọc. Khi xuất, chữ cái **T** hoặc **F** (tương ứng với **.TRUE.** hoặc **.FALSE.** sẽ in ra tại vị trí thứ 5, tức vị trí cuối cùng bên phải trong 5 vị trí.

9) Chúng ta có thể sử dụng một số đặc điểm bổ sung trong cách viết các đặc tả của lệnh FORMAT nhằm nâng cao chất lượng bản ghi, theo đúng ý định biểu diễn của mình hay làm cho lệnh FORMAT trông ngắn gọn. Có thể dùng những cách dưới đây:

*Cách viết lặp lại các đặc tả:* Thí dụ những cặp lệnh sau đây hoàn toàn tương đương:

10 FORMAT (3X, I2, 3X, I2)

10 FORMAT (2 (3X, I2))

20 FORMAT (1X, F4.1, F4.1, 1X, I3, 1X, I3, 1X, I3)

20 FORMAT (1X, 2F4.1, 3 (1X, I3))

*Dùng dấu gạch chéo (/)* trong lệnh FORMAT chỉ kết thúc dòng in trước khi bắt đầu các đặc tả sau nó. Thí dụ, khi cần in dòng tiêu đề của một bảng số cùng với những tiêu đề cột, chúng ta có thể dùng:

PRINT 5

5 FORMAT (1X, 'KET QUA QUAN TRAC' // 2X, 'Gio', 3X,

\* 'Toc do', 3X, 'Huong')

Sau khi in xong đoạn văn bản KET QUA QUAN TRAC, dấu gạch chéo thứ nhất chỉ dẫn cho máy kết thúc dòng, xuống dòng mới, dấu gạch chéo thứ hai chỉ dẫn bỏ qua ngay dòng này không in, phát sinh ra một dòng trống trước khi in các tiêu đề cột ở dòng thứ ba như ta thấy dưới đây:

\*\*\*SORRY, THIS MEDIA TYPE IS NOT SUPPORTED.\*\*\*

*Dùng đặc tả bảng T, TR, TL* để căn lề trái các tiêu đề cột một bảng số. Thí dụ các cặp lệnh cùng nhãn sau đây sẽ là tương đương với nhau:

Nhập và xuất dữ liệu đơn giản

600 FORMAT (F6.1, 15X, I7)

600 FORMAT (F6.1, T22, I7)

ở lệnh thứ hai: sau khi ghi ra số thực với 6 vị trí, nhảy ngay tới vị trí 22 để bắt đầu ghi số nguyên.

85 FORMAT (1X, 25X, 'Do cao', 5X, 'Huong')

85 FORMAT (T27, 'Do cao', TR5, 'Huong')

ở lệnh thứ hai: nhảy ngay tới vị trí thứ 27 để ghi tiêu đề "Do cao", sau đó do có đặc tả TR5 xuất phát từ vị trí hiện thời sẽ nhảy sang phải 5 vị trí để ghi tiêu đề "Huong".

*Đặc tả \* có tác dụng ngăn không xuống dòng trong một lệnh in hoặc đọc. Có thể dùng đặc tả này trong trường hợp muốn viết một lời nhắc yêu cầu người dùng nhập thông tin từ bàn phím nhưng sau khi viết lời nhắc thì không xuống dòng, con nháy đứng trên cùng dòng ngay sau lời nhắc chờ người dùng nhập thông tin từ bàn phím theo yêu cầu của lệnh đọc. Thí dụ nhóm lệnh sau đây sẽ làm chức năng đó:

PRINT 7

7 FORMAT (1X, 'Ten file so lieu: ', \)

READ (\*, '(A50)') NAME

*Về số lượng các đặc tả:* Khi số các đặc tả nhiều hơn số mục trong danh sách các mục in, thí dụ:

PRINT 1, TOCDO, KHOANG

1 FORMAT (4 F5.2)

máy sẽ chọn lấy số tối đa các đặc tả cần dùng, số đặc tả còn lại bị bỏ qua. Trong trường hợp này lệnh in có 2 mục in - 2 giá trị số thực, nhưng lệnh FORMAT có 4 đặc tả số thực, như vậy số đặc tả là thừa. Máy sẽ chọn lấy hai đặc tả và in bình thường như chúng ta mong muốn.

Khi số đặc tả ít hơn số mục in, thí dụ trong lệnh in sau:

PRINT 20, TEM, VOL

20 FORMAT (1X, f6.2)

## Nhập và xuất dữ liệu đơn giản

Trong trường hợp này máy căn các mục in và đặc tả cho đến hết danh sách đặc tả, sau đó có thể xảy ra hai khả năng:

- 1) In luôn buffer hiện tại và bắt đầu một buffer mới.
- 2) Quay trở lại đầu danh sách đặc tả cho đến khi gặp dấu ngoặc đơn trái và lại căn từng cặp mục in, đặc tả cho các mục in còn lại.

Trong lệnh in trên giá trị của TEM được căn theo đặc tả F6.2. Vì không có đặc tả cho VOL nên ta làm như sau:

- 1) In giá trị của TEM sau một vị trí trống.
- 2) Khi quay trở lại về phía đầu của danh sách các đặc tả (dấu ngoặc trái) và căn F6.2 cho giá trị VOL. Sau đó ta đặt tới đầu của danh sách và dấu trống để in VOL. Do đó TEM và VOL được in trên hai dòng riêng biệt.

Trong Fortran 90 cho phép các tham số độ rộng đặc tả, số lần lặp của đặc tả có thể là biến. Thí dụ FORMAT (<m>F<n>.<k>).

### Bài tập

1. Viết đoạn chương trình đọc giá trị vào hai biến thực A và B, đổi giá trị của hai biến đó cho nhau.
2. Điều gì sẽ xảy ra khi thực hiện chương trình sau và ta nhập vào bàn phím lần lượt số 1, dấu phẩy, số 10 và dấu chấm rồi gõ phím Enter.

```
PRINT *, ' Cho cac gia tri cua hai so nguyen I1, I2 ! '
```

```
READ * , IDAU, ICUOI
```

```
PRINT 4, IDAU, ICUOI
```

```
4 FORMAT (1X, 'I1 = ', I5, 'I2 = ', I5)
```

```
END
```

3. Mô tả những gì sẽ in lên màn hình khi thực hiện các lệnh dưới đây:

```
REAL X
```

```
X = ?27.632
```

Nhập và xuất dữ liệu đơn giản

```
PRINT 5 , X
```

```
5 FORMAT (1X, 'X = ', F7.1, ' DEGREES')
```

4. Mô tả những gì sẽ in ra máy in khi thực hiện những lệnh dưới đây:

```
A = 3.184
```

```
PRINT 1
```

```
1 FORMAT (1X, '0')
```

```
PRINT 2
```

```
2 FORMAT ('+' , '- = ' , F5.2)
```

5. Mô tả những gì sẽ in lên màn hình khi thực hiện các lệnh dưới đây:

```
REAL DIST , VEL
```

```
DIST = 28732.5
```

```
VEL = 72.6
```

```
PRINT 10, DIST, VEL
```

```
10 FORMAT (1X, 'DISTANCE = ' , E10.3,
```

```
* 5X , 'VELOCITY = ' , F5.2)
```

6. Viết đoạn chương trình nhập vào từ bàn phím tên 5 môn thi của học kỳ cùng với điểm thi từng môn của mình. Tính điểm trung bình và in lên màn hình thành một bảng có hình thức như sau:

```
BANG DIEM THI HOC KY
```

```
-----
```

```
TT TEN MON HOC DIEM
```

```
-----
```

```
1 Tên môn thứ 1 8
```



Nhập và xuất dữ liệu đơn giản

2 Tên môn thứ 2 7

3 Tên môn thứ 3 8

4 Tên môn thứ 4 7

5 Tên môn thứ 5 9

-----

Diem trung binh hoc ky 7,8