

Phần 1 - Lập trình

Làm quen với Fortran

Phan Ngọc Hưng

Khoa Vật lý - Đại học Sư phạm Tp. Hồ Chí Minh
280 - An Dương Vương - P.4 - Q.5 - Tp.HCM

Bài giảng môn Phương pháp số & lập trình - 2017

Mục lục

- 1 Mục tiêu**
- 2 Các bước xây dựng một chương trình**
- 3 Cấu trúc điển hình của một chương trình**
- 4 Vận hành một chương trình Fortran 95**
- 5 Các phép toán & các kiểu dữ liệu**
- 6 Câu lệnh điều kiện**
- 7 Vòng lệnh lặp**
- 8 Chuỗi - mảng dữ liệu**
- 9 Xuất - nhập dữ liệu với tập tin ngoài**
- 10 Hàm & Chương trình con**

Mục lục

- 1** Mục tiêu
- 2** Các bước xây dựng một chương trình
- 3** Cấu trúc điển hình của một chương trình
- 4** Vận hành một chương trình Fortran 95
- 5** Các phép toán & các kiểu dữ liệu
- 6** Câu lệnh điều kiện
- 7** Vòng lệnh lặp
- 8** Chuỗi - mảng dữ liệu
- 9** Xuất - nhập dữ liệu với tập tin ngoài
- 10** Hàm & Chương trình con

Mục tiêu

- 1 Biết cách chạy một chương trình Fortran sẵn có.
- 2 Hiểu cấu trúc điển hình của một chương trình Fortran.
- 3 Hiểu sơ lược về qui định cú pháp của Fortran.
- 4 Viết được một chương trình Fortran đơn giản.

Tài liệu tham khảo

Introduction to Programming using Fortran 95.

Địa chỉ download: <http://www.fortrantutorial.com>

Mục lục

- 1 Mục tiêu**
- 2 Các bước xây dựng một chương trình**
- 3 Cấu trúc điển hình của một chương trình**
- 4 Vận hành một chương trình Fortran 95**
- 5 Các phép toán & các kiểu dữ liệu**
- 6 Câu lệnh điều kiện**
- 7 Vòng lệnh lặp**
- 8 Chuỗi - mảng dữ liệu**
- 9 Xuất - nhập dữ liệu với tập tin ngoài**
- 10 Hàm & Chương trình con**

Các bước xây dựng một chương trình

- 1** Xác định bài toán - đúng, đủ, rõ ràng
- 2** Xây dựng giải thuật
- 3** Viết chương trình trên ngôn ngữ lập trình
- 4** Biên dịch và chạy chương trình
- 5** Kiểm tra, điều chỉnh

Mục lục

- 1 Mục tiêu**
- 2 Các bước xây dựng một chương trình**
- 3 Cấu trúc điển hình của một chương trình**
- 4 Vận hành một chương trình Fortran 95**
- 5 Các phép toán & các kiểu dữ liệu**
- 6 Câu lệnh điều kiện**
- 7 Vòng lệnh lặp**
- 8 Chuỗi - mảng dữ liệu**
- 9 Xuất - nhập dữ liệu với tập tin ngoài**
- 10 Hàm & Chương trình con**

Cấu trúc điển hình của một chương trình

- 1 PROGRAM program_name - khai báo tên chương trình chính
- 2 Declarative statements - câu lệnh khai báo
- 3 Executive statements - câu lệnh thực hiện
- 4 END PROGRAM program_name - khai báo kết thúc chương trình chính
- 5 Subroutines - Functions (các chương trình và hàm con nếu có)

Làm quen với Fortran

└ Cấu trúc điển hình của một chương trình

 └ Ví dụ về cấu trúc chương trình FORTRAN

Ví dụ về cấu trúc chương trình

PROGRAM Cinderella

Cinderella is beautiful.

Sisters are ugly, and coarse, and there are two of them.

The Prince is rich.

Dress Cinderella.

Take her to the dance.

Nullify the Sisters.

Make the Prince marry Cinderella.

END PROGRAM Cinderella

Làm quen với Fortran

└ Cấu trúc điển hình của một chương trình

 └ Ví dụ về cấu trúc chương trình FORTRAN

Ví dụ về cấu trúc chương trình

FUNCTION Dress Cinderella

Cinderella's dress is old.

The fairy has a magic wand.

Turn the old dress into new one.

Give Cinderella the glass slippers.

Cinderella is now dressed beatifully.

END FUNCTION Dress Cinderella

Làm quen với Fortran

└ Cấu trúc điển hình của một chương trình

 └ Ví dụ về cấu trúc chương trình FORTRAN

Ví dụ về cấu trúc chương trình

PROGRAM Cinderella

Cinderella is beautiful

Sisters are ugly, and coarse, and there are two of them

The Prince is rich

Dress Cinderella

Take her to the dance

Nullify the Sisters

Make the Prince marry her

END PROGRAM Cinderella

Làm quen với Fortran

└ Cấu trúc điển hình của một chương trình

 └ Ví dụ về cấu trúc chương trình FORTRAN

Ví dụ về cấu trúc chương trình

FUNCTION Dress Cinderella

Cinderella's dress is old.

The fairy has a magic wand.

Turn the old dress into new one.

Give Cinderella the crystal shoes.

Cinderella is now dressed beautifully.

END FUNCTION Dress Cinderella

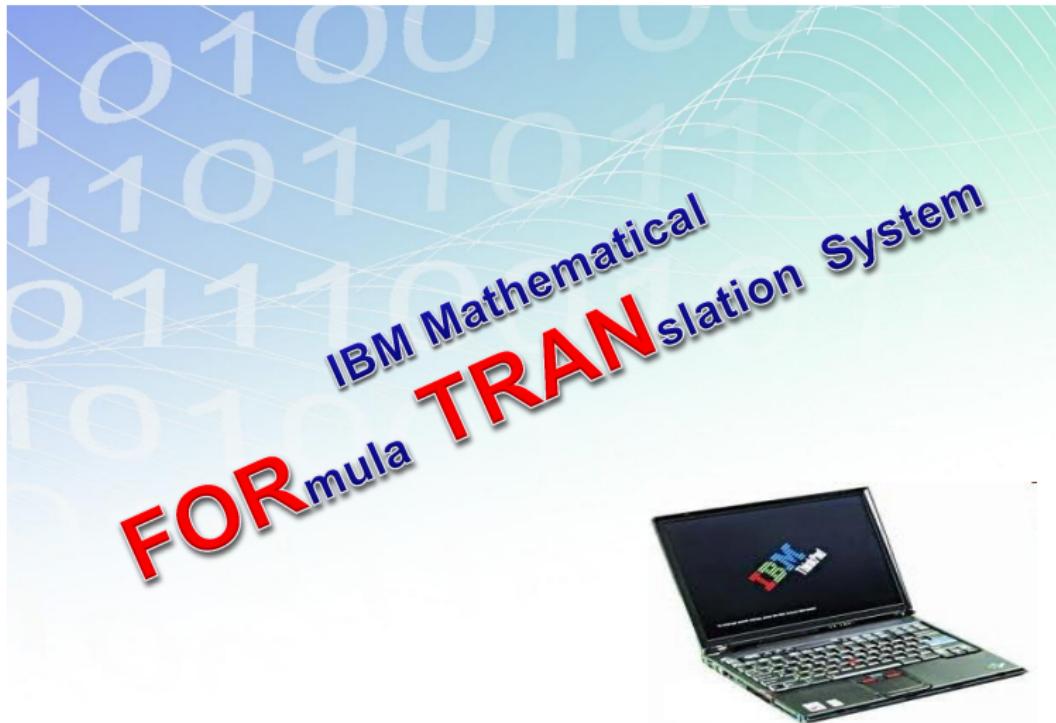
Mục lục

- 1 Mục tiêu**
- 2 Các bước xây dựng một chương trình**
- 3 Cấu trúc điển hình của một chương trình**
- 4 Vận hành một chương trình Fortran 95**
- 5 Các phép toán & các kiểu dữ liệu**
- 6 Câu lệnh điều kiện**
- 7 Vòng lệnh lặp**
- 8 Chuỗi - mảng dữ liệu**
- 9 Xuất - nhập dữ liệu với tập tin ngoài**
- 10 Hàm & Chương trình con**

Làm quen với Fortran

- └ Vận hành một chương trình Fortran 95
- └ Vận hành chương trình trên FORTRAN

Giới thiệu về Fortran

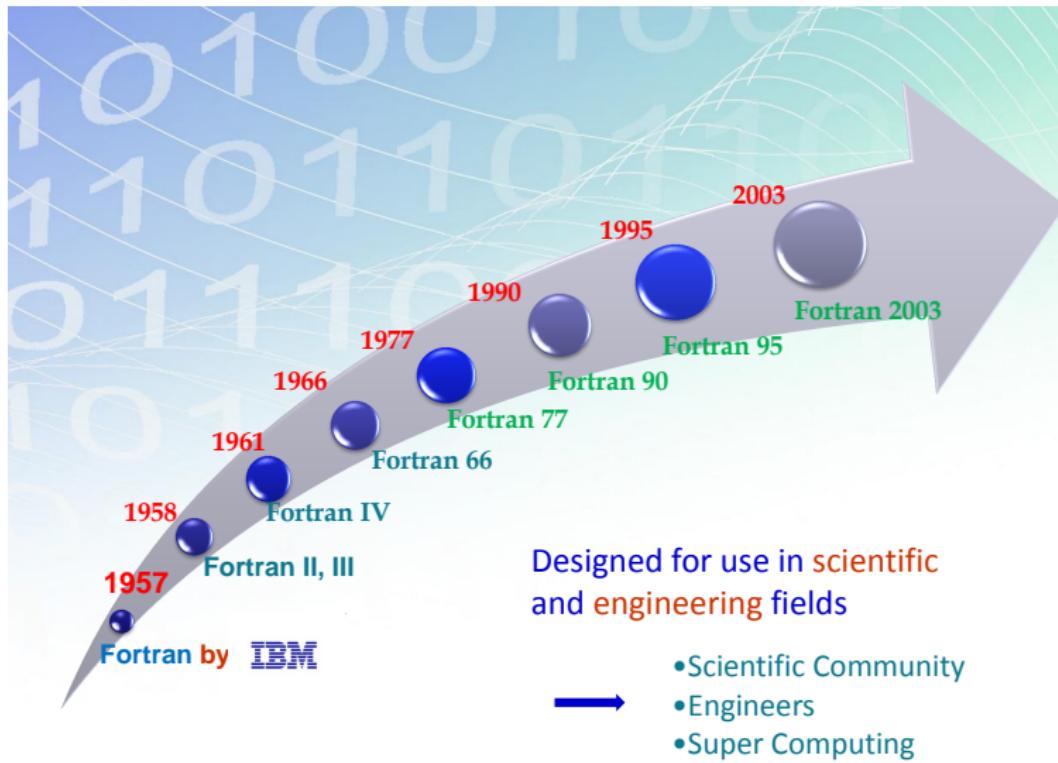


Làm quen với Fortran

└ Vận hành một chương trình Fortran 95

└ Vận hành chương trình trên FORTRAN

Giới thiệu về Fortran



Các bước vận hành một chương trình Fortran

- 1 Mở chương trình Fortran
- 2 Soạn thảo / mở chương trình sẵn có
- 3 Biên dịch (compile)
- 4 Tạo file thực thi (Build)
- 5 Chạy chương trình (Start / Run / Execute)
- 6 Đóng chương trình

Qui ước cú pháp

Có 2 cách định dạng file Fortran: định dạng tự do (free format) và định dạng cố định (fixed format).

Vị trí cột	Định dạng cố định	Định dạng tự do
1	Đặt dấu chú thích (C hoặc !)	Vùng gõ gì cũng được
1–5	Đặt nhãn nếu có	Vùng gõ gì cũng được
6	Đặt dấu tiếp tục dòng lệnh (&)	Vùng gõ gì cũng được
7–72	Vùng viết câu lệnh	Vùng gõ gì cũng được
73–80	Đánh số thứ tự dòng lệnh nếu cần	Vùng gõ gì cũng được
81–132		Vùng gõ gì cũng được

Làm quen với Fortran

└ Vận hành một chương trình Fortran 95

└ Ví dụ

Ví dụ 1

Chương trình in ra màn hình dòng chữ “FORTRAN first run”

```
PROGRAM First
  WRITE(*,*)"FORTRAN first run"
END PROGRAM First
```

Làm quen với Fortran

└ Vận hành một chương trình Fortran 95

└ Ví dụ

Ví dụ 2

Chương trình nhập hai dữ liệu số

PROGRAM Second

WRITE(,*)"Moi ban nhap vao 2 so"*

READ(,*)a,b*

END PROGRAM Second

Làm quen với Fortran

└ Vận hành một chương trình Fortran 95

└ Ví dụ

Ví dụ 3

Chương trình nhập hai dữ liệu số

PROGRAM Third

REAL::a,b

WRITE(,*)"Moi ban nhap vao 2 so"*

READ(,*)a,b*

WRITE(,*)"Ban vua nhap vao 2 so ",a," va ",b*

END PROGRAM Third

Làm quen với Fortran

└ Vận hành một chương trình Fortran 95

└ Ví dụ

Ví dụ 4

Chương trình tính tổng của hai số tự nhiên

```
PROGRAM Forth
INTEGER::a,b
REAL::answer
WRITE(*,*)"Moi ban nhap vao 2 so"
READ(*,*)a,b
WRITE(*,*)"Ban vua nhap vao 2 so ",a," va ",b
answer=a+b
WRITE(*,*)"Tong cua chung la ",answer
END PROGRAM Forth
```

Ví dụ 5

Chương trình tính thương của hai số tự nhiên

```
PROGRAM Fifth
```

```
INTEGER::a,b
```

```
REAL::answer
```

```
WRITE(*,*)"Moi ban nhap vao 2 so"
```

```
READ(*,*)a,b
```

```
WRITE(*,*)"Ban vua nhap vao 2 so ",a," va ",b
```

```
answer=a/b
```

```
WRITE(*,*)"Thuong cua chung la ",answer
```

```
END PROGRAM Fifth
```

Làm quen với Fortran

└ Vận hành một chương trình Fortran 95

└ Ví dụ

Ví dụ 6

Chương trình tính thương của hai số tự nhiên

```
PROGRAM Sixth
INTEGER::a,b
REAL::answer
WRITE(*,*)"Moi ban nhap vao 2 so"
READ(*,*)a,b
WRITE(*,*)"Ban vua nhap vao 2 so ",a," va ",b
answer=a/1.0/b
WRITE(*,*)"Thuong cua chung la ",answer
END PROGRAM Sixth
```

Ví dụ 7

Chương trình tính thương của hai số

PROGRAM Seventh

WRITE(,*)"Moi ban nhap vao 2 so"*

READ(,*)a,b*

WRITE(,*)"Ban vua nhap vao 2 so ",a," va ",b*

answer=a/b

WRITE(,*)"Thuong cua chung la ",answer*

END PROGRAM Seventh

Làm quen với Fortran

└ Vận hành một chương trình Fortran 95

└ Ví dụ

Ví dụ 8

Chương trình tính thương của hai số

PROGRAM Eighth

WRITE(,*)"Moi ban nhap vao 2 so"*

READ(,*)m,n*

WRITE(,*)"Ban vua nhap vao 2 so ",m," va ",n*

answer=m/n

WRITE(,*)"Thuong cua chung la ",answer*

END PROGRAM Eighth

Ví dụ 7 và Ví dụ 8 có cấu trúc giống nhau nhưng kết quả lại khác nhau, vì sao như vậy?

Ví dụ 9

Chương trình tính thương của hai số

```
PROGRAM Ninth
IMPLICIT NONE
WRITE(*,*)"Moi ban nhap vao 2 so"
READ(*,*)m,n
WRITE(*,*)"Ban vua nhap vao 2 so ",m," va ",n
answer=m/n
WRITE(*,*)"Thuong cua chung la ",answer
END PROGRAM Ninth
```

Ví dụ 9 (t.t.)

Chương trình báo lỗi

```
...Error: 'ANSWER' has no IMPLICIT type  
...Error: 'M' has no IMPLICIT type  
...Error: 'N' has no IMPLICIT type
```

Lý do: IMPLICIT NONE xóa bỏ tất cả các mặc định về kiểu dữ liệu ngầm định của Fortran. Mọi biến số được sử dụng cần phải được khai báo kiểu dữ liệu.

Lời khuyên

Hãy sử dụng **IMPLICIT NONE** trong tất cả các chương trình.

Mục lục

- 1 Mục tiêu**
- 2 Các bước xây dựng một chương trình**
- 3 Cấu trúc điển hình của một chương trình**
- 4 Vận hành một chương trình Fortran 95**
- 5 Các phép toán & các kiểu dữ liệu**
- 6 Câu lệnh điều kiện**
- 7 Vòng lệnh lặp**
- 8 Chuỗi - mảng dữ liệu**
- 9 Xuất - nhập dữ liệu với tập tin ngoài**
- 10 Hàm & Chương trình con**

Các kiểu dữ liệu

FORTRAN dùng 5 kiểu dữ liệu:

- 1 INTEGER - Số nguyên
- 2 REAL - số thực
- 3 COMPLEX - số phức
- 4 LOGICAL - logic
- 5 CHARACTER - ký tự

Cách khai báo

<kiểu dữ liệu> :: <tên biến số>

Ví dụ:

INTEGER::a

REAL::b,c

Làm quen với Fortran

- └ Các phép toán & các kiểu dữ liệu
 - └ Kiểu số nguyên - INTEGER

Kiểu số nguyên - INTEGER

1 Cú pháp khai báo:

INTEGER :: i, j, k, ...

2 Ví dụ:

1

+365

-9

299791345

Làm quen với Fortran

└ Các phép toán & các kiểu dữ liệu

└ Kiểu số nguyên - INTEGER

Phép toán & hàm trên kiểu số nguyên

- Xét 2 số nguyên $i = 15$ và $k = 6$

- Phép cộng $i + k = 21$
- Phép trừ $i - k = 9$
- Phép nhân $i \times k = 90$
- Phép chia $i \div k = 2$
- Phép lũy thừa $i^k = 11390625$

- Hàm dựng sẵn (intrinsic functions):

$$\text{MOD}(i,k) = i - \frac{i}{k} \times k$$

ABS(i)

HUGE(i)

...

- └ Các phép toán & các kiểu dữ liệu
 - └ Kiểu số thực - REAL

Kiểu số thực - REAL

1 Cú pháp khai báo:

REAL :: a, b, c,...

2 Ví dụ:

-78.915443

1877724

+3.000

4.65E-9

-.1E+6

Phép toán & hàm trên kiểu số thực

- Hàm dựng sẵn (intrinsic functions):

INT(2.3) = 2

REAL(3) = 3.00000

ABS, SQRT, SIN, COS, EXP, LOG, ...

HUGE, RANGE, ...

Kiểu số phức - COMPLEX

1 Cú pháp khai báo:

COMPLEX :: a, b, c,...

2 Biểu diễn số phức

(real-part, imaginary-part)

trong đó, *real-part* và *imaginary-part* là kiểu số thực

3 Ví dụ:

$z1 = (3.756, 0.051)$

$z2 = (3E-2, -.4)$

Kiểu logic - LOGICAL

1 Cú pháp khai báo:

LOGICAL :: a, b, c,...

2 Biến số logic chỉ nhận 1 trong 2 giá trị: .TRUE. hoặc .FALSE.

3 Ví dụ:

a = .TRUE.

b = .FALSE.

c = (4 < 5)

d = (2 == 4 / 2)

e = (a == .FALSE.) ...

Làm quen với Fortran

└ Các phép toán & các kiểu dữ liệu

 └ Kiểu logic - LOGICAL

Hàm dựng sẵn

Ví dụ: a = .TRUE., b = .FALSE., c = .TRUE., d = .FALSE.

.NOT.

a.NOT.a = .FALSE.

.AND.

a.AND.b = .FALSE.

a.AND.c = .TRUE.

b.AND.d = .FALSE.

.OR.

a.OR.b = .TRUE.

a.OR.c = .TRUE.

b.OR.d = .FALSE.

.EQV.

a.EQV.b = .FALSE.

a.EQV.c = .TRUE.

b.EQV.d = .TRUE.

.NEQV.

a.NEQV.b = .TRUE.

a.NEQV.c = .FALSE.

b.NEQV.d = .FALSE.

Kiểu ký tự - CHARACTER

1 Cú pháp khai báo:

CHARACTER(LENGTH) :: a, b, c,...

trong đó LENGTH là độ dài của chuỗi ký tự

2 Ví dụ

CHARACTER(10)::name

CHARACTER(11)::code

CHARACTER::vowel

Làm quen với Fortran

└ Các phép toán & các kiểu dữ liệu

└ Kiểu ký tự - CHARACTER

Hàm dựng sẵn

Ví dụ:

CHARACTER(20)::name

TRIM(name)→"Barack Obama"

"home"// "work" → "homework"

- └ Các phép toán & các kiểu dữ liệu
 - └ Các phép toán

Các phép toán

1 Các phép toán đại số

Cộng	Trừ	Nhân	Chia	Lũy thừa
+	-	*	/	**

2 Các phép toán quan hệ

Nhỏ hơn	nhỏ hơn hoặc bằng	bằng	lớn hơn hoặc bằng	lớn hơn	không bằng
< .LT.	<= .LE.	== .EQ.	>= .GE.	> .GT.	\= .NE.

3 Các phép toán logic

Phủ định	Và	Hoặc	Tương đương	Không tương đương
.NOT.	.AND.	.OR.	.EQV.	.NEQV.

Các phép toán

Thứ tự ưu tiên các phép toán

- 1** dấu ngoặc ()
- 2** lũy thừa **
- 3** Nhân * & chia /
- 4** Cộng + & trừ -
- 5** Các phép toán quan hệ
- 6** .NOT.
- 7** .AND.
- 8** .OR.
- 9** .EQV. & .NEQV.

Ví dụ

- a.AND..NOT.b.NEQV.c.OR.d
$$(a.\text{AND}.(\text{.NOT}.b)) .\text{NEQV}.(c.\text{OR}.d)$$
- a.EQV.b.NEQV.c.EQV.d
$$((a.\text{EQV}.b) .\text{NEQV}.c) .\text{EQV}.d$$
- j.NE.k.AND.m.GT.k.EQV.j+m.EQ.k
$$j \neq k.\text{AND}.m > k.\text{EQV}. j + m == k$$
$$((j \neq k) .\text{AND}. (m > k)) .\text{EQV}. ((j + m) == k)$$

Ví dụ

- $3*5+7-9 = ((3*5)+7)-9 = 13$
- $21/3*5/7 = ((21/3)*5)/7 = 5$
- $2^{**}2^{**}2^{**}2 = (2^{**}(2^{**}(2^{**}2))) = 65536$
- $4^{**}4/2*2^{**}2 = ((4^{**}4)/2)*(2^{**}2) = 512$
- ...

Mục lục

- 1 Mục tiêu**
- 2 Các bước xây dựng một chương trình**
- 3 Cấu trúc điển hình của một chương trình**
- 4 Vận hành một chương trình Fortran 95**
- 5 Các phép toán & các kiểu dữ liệu**
- 6 Câu lệnh điều kiện**
- 7 Vòng lệnh lặp**
- 8 Chuỗi - mảng dữ liệu**
- 9 Xuất - nhập dữ liệu với tập tin ngoài**
- 10 Hàm & Chương trình con**

Làm quen với Fortran

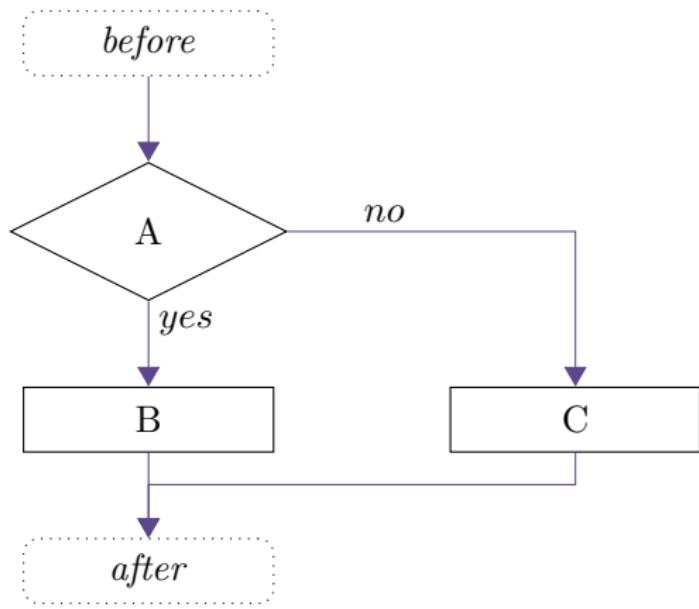
└ Câu lệnh điều kiện

└ Cú pháp - sơ đồ khối

Cú pháp - Sơ đồ khối

Cú pháp

```
IF A THEN  
    B  
ELSE  
    C  
END IF
```



Làm quen với Fortran

└ Câu lệnh điều kiện

└ Ví dụ

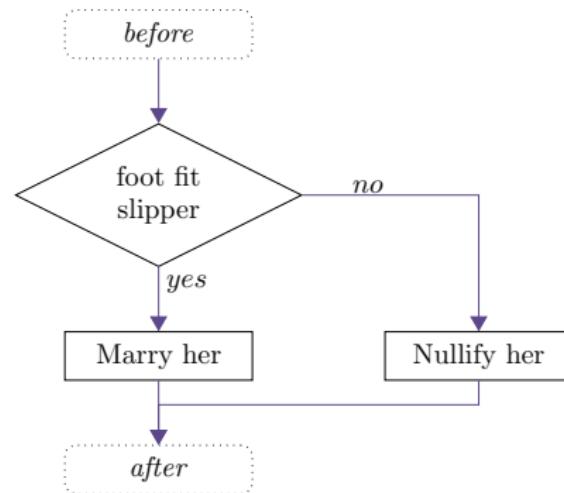
Ví dụ 1

In the story of Cinderella...

“ The prince announced, with the sound of trumpets, that he would marry the lady whose foot would exactly fit the slipper...”

Cú pháp

```
IF foot fits slipper THEN  
    marry her!  
ELSE  
    nullify her!  
END IF
```



Ví dụ 2

Chương trình xác định một số là chẵn hay lẻ

```
PROGRAM first
IMPLICIT NONE
INTEGER::i
WRITE(*,*)"Moi ban nhap 1 so tu nhien"
READ(*,*)i
IF (MOD(i,2)==0) THEN
    WRITE(*,*)"Day la mot so chan"
ELSE
    WRITE(*,*)"Day la mot so le"
END IF
END PROGRAM first
```

Làm quen với Fortran

└ Câu lệnh điều kiện

└ Ví dụ

Ví dụ 3

Chương trình tính diện tích tam giác khi biết độ dài ba cạnh

PROGRAM Triangle

! Declarative statements:

IMPLICIT NONE

REAL :: a, b, c ! The lengths of a triangle's sides

REAL :: s ! The semiperimeter of the triangle

REAL :: areasq ! The square of the triangle's area

REAL :: area ! The area of the triangle

Ví dụ 3 (cont. 1)

! Data input:

WRITE (,*) "This program calculates the area of a triangle."*

WRITE (,*) "Type in the lengths of the three sides:"*

READ (,*) a, b, c*

WRITE (,*) "Check: you have input the following lengths"*

WRITE (,*) a, b, c*

Ví dụ 3 (cont. 2)

! Calculations:

*s = 0.5 * (a+b+c)*

areasq = s(s-a)*(s-b)*(s-c)*

IF (areasq<0.0) THEN

WRITE (,*) "Error: that is not a real triangle"*

ELSE

area=SQRT(areasq)

Làm quen với Fortran

└ Câu lệnh điều kiện

└ Ví dụ

Ví dụ 3 (cont. 3)

! Data output:

WRITE (,*) “The area of the triangle is”, area
END IF*

END PROGRAM Triangle

Làm quen với Fortran

└ Câu lệnh điều kiện

└ Ví dụ

Luyện tập

Bài tập 1

Viết lưu đồ tìm nghiệm thực của phương trình bậc 1 $ax + b = 0$ và viết chương trình tương ứng với lưu đồ đó.

Bài tập 2

Viết lưu đồ tìm nghiệm thực của phương trình bậc 2 $ax^2 + bx + c = 0$ và viết chương trình tương ứng với lưu đồ đó.

Làm quen với Fortran

└ Câu lệnh điều kiện

└ Ví dụ

Luyện tập

Bài tập 3

Viết chương trình tính tiền taxi từ số km đã đi theo qui tắc:

- Km đầu tiên giá 14000 đồng.
- Từ km thứ 2 đến km thứ 20 giá 11000 đồng.
- Từ km thứ 20 trở đi giá 10000 đồng.
- Đi trên 100 km được giảm 10% tổng giá tiền.

Mục lục

- 1** Mục tiêu
- 2** Các bước xây dựng một chương trình
- 3** Cấu trúc điển hình của một chương trình
- 4** Vận hành một chương trình Fortran 95
- 5** Các phép toán & các kiểu dữ liệu
- 6** Câu lệnh điều kiện
- 7** Vòng lệnh lặp
- 8** Chuỗi - mảng dữ liệu
- 9** Xuất - nhập dữ liệu với tập tin ngoài
- 10** Hàm & Chương trình con

Làm quen với Fortran

└ Vòng lệnh lặp

 └ Vòng lặp xác định

Câu hỏi

Vòng lặp là gì

Làm quen với Fortran

└ Vòng lệnh lặp

 └ Vòng lặp xác định

Vì sao phải cần vòng lặp



Làm quen với Fortran

└ Vòng lệnh lặp

 └ Vòng lặp xác định

Vì sao phải cần vòng lặp

PROGRAM Description

Snow White is beautiful

There are seven dwarfs, who are small

Snow White is looking at the first dwarf

Snow White is looking at the second dwarf

Snow White is looking at the third dwarf

...

Snow White is looking at the seventh dwarf

Làm quen với Fortran

└ Vòng lệnh lặp

 └ Vòng lặp xác định

Vì sao phải cần vòng lặp

The first dwarf is looking at Snow White

The second dwarf is looking at Snow White

The third dwarf is looking at Snow White

...

The seventh dwarf is looking at Snow White

END PROGRAM Description

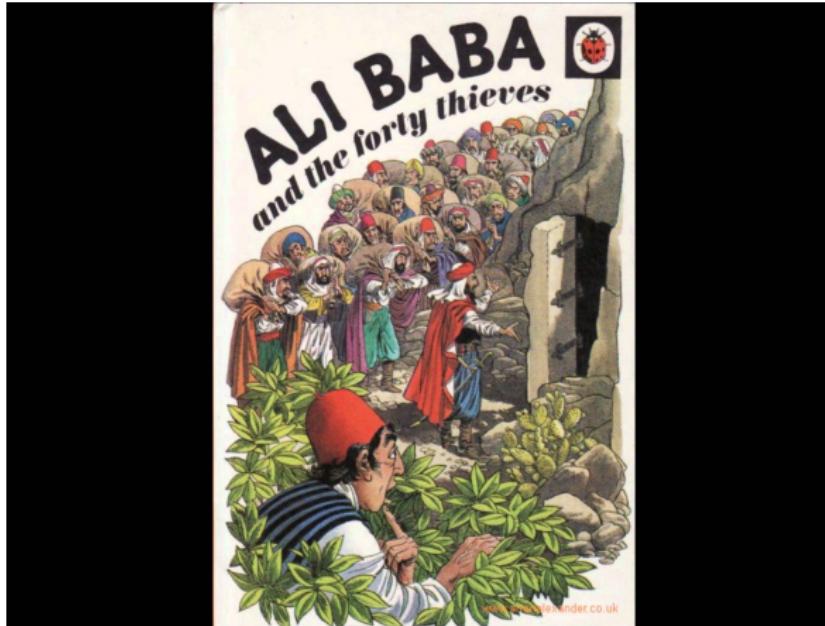
Làm quen với Fortran

└ Vòng lệnh lặp

└ Vòng lặp xác định

Vì sao phải cần vòng lặp

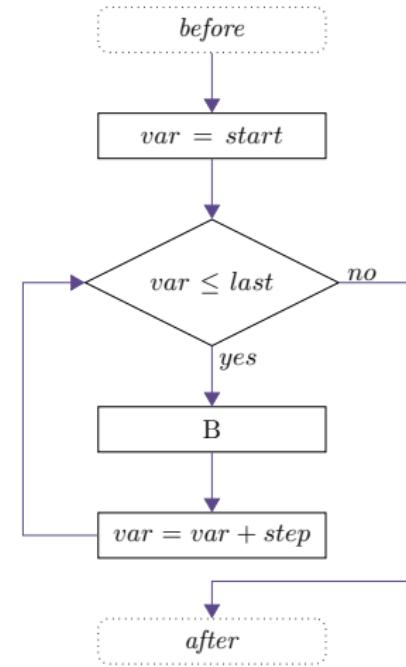
Sẽ thật mệt mỏi...



Vòng lặp xác định

Cú pháp

```
...
DO var = start, last, step
    B
END DO
...
```



Làm quen với Fortran

└ Vòng lệnh lặp

 └ Vòng lặp xác định

Ví dụ

PROGRAM Description

Snow White is beautiful

There are seven dwarfs, who are small

DO i = 1,7

Snow White is looking at the i-th dwarf

END DO

DO i=1,7

The i-th dwarf is looking at Snow White

END DO

END PROGRAM Description

Ví dụ 1

Viết chương trình tính tổng của $S = 1 + 2 + 3 + \dots + 100$

```
PROGRAM Tong
implicit none
INTEGER :: i
INTEGER :: s
s=0
DO i=1,100
    s=s+i
END DO
WRITE(*,*)"tong la ",s
END PROGRAM Tong
```

Làm quen với Fortran

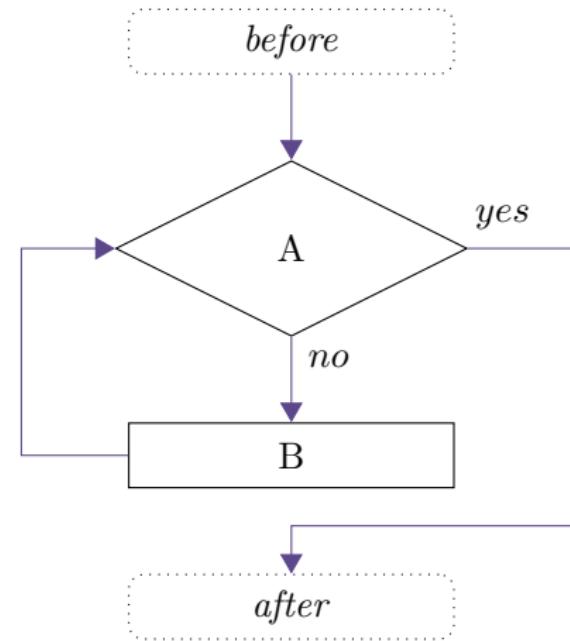
└ Vòng lệnh lặp

 └ Vòng lặp không xác định

Câu lệnh lặp không xác định 1

Cú pháp

```
...
DO
    IF A EXIT
    B
END DO
...
```



Làm quen với Fortran

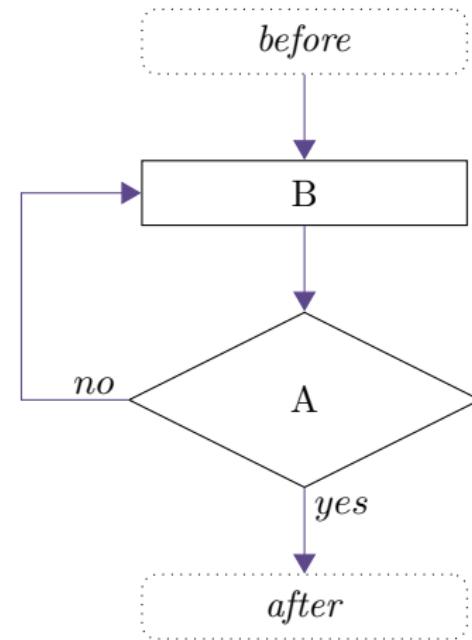
└ Vòng lệnh lặp

 └ Vòng lặp không xác định

Câu lệnh lặp không xác định 2

Cú pháp

```
...
DO
    B
    IF A EXIT
END DO
...
```



Làm quen với Fortran

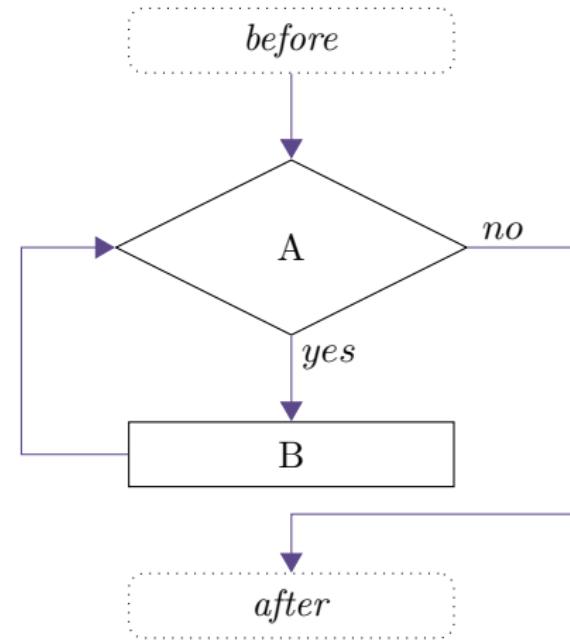
└ Vòng lệnh lặp

 └ Vòng lặp không xác định

Câu lệnh lặp không xác định 3

Cú pháp

```
...
DO WHILE A
    B
END DO
...
```



Bài tập

- 1 Viết chương trình tính tổng $S = 1 + 2 + 3 + \dots + n$ với n nhập từ bàn phím.
- 2 Viết chương trình kiểm tra số nhập vào có phải số nguyên dương hay không, nếu không thì yêu cầu nhập lại.
- 3 Viết chương trình tính giai thừa của một số nguyên dương nhập vào từ bàn phím.
- 4 Viết chương trình kiểm tra một số nhập vào có phải là số nguyên tố.
- 5 Viết chương trình tìm số n nhỏ nhất thỏa $1 + 2 + \dots + n > 1000$.
- 6 Viết chương trình tìm n số nguyên tố đầu tiên.
- 7 Viết chương trình phân tích một số ra thừa số nguyên tố.

Mục lục

- 1 Mục tiêu**
- 2 Các bước xây dựng một chương trình**
- 3 Cấu trúc điển hình của một chương trình**
- 4 Vận hành một chương trình Fortran 95**
- 5 Các phép toán & các kiểu dữ liệu**
- 6 Câu lệnh điều kiện**
- 7 Vòng lệnh lặp**
- 8 Chuỗi - mảng dữ liệu**
- 9 Xuất - nhập dữ liệu với tập tin ngoài**
- 10 Hàm & Chương trình con**

Làm quen với Fortran

└ Chuỗi - mảng dữ liệu

└ Khái niệm

Ví dụ 1

Viết chương trình tính giá trị trung bình của mươi số.

PROGRAM Ave1

IMPLICIT NONE

REAL::x1,x2,x3,x4,x5,x6,x7,x8,x9,x10

REAL::s,average

Làm quen với Fortran

└ Chuỗi - mảng dữ liệu

└ Khái niệm

Ví dụ 1 (t.t)

$s=0$

WRITE(,*)"Xin nhap so thu nhat: "; READ(*,*)x1*

$s=s+x1$

WRITE(,*)"Xin nhap so thu hai: "; READ(*,*)x2*

$s=s+x2$

...

WRITE(,*)"Xin nhap so thu muoi: "; READ(*,*)x10*

$s=s+x10$

average=s/10

WRITE(,*)"Trung binh la ",average*

END PROGRAM Ave1

Làm quen với Fortran

└ Chuỗi - mảng dữ liệu

└ Khái niệm

Ví dụ 1 (t.t)



...

DO i=1,7

Snow White is looking at i-th dwarf

END DO

...

Mảng dữ liệu là gì

Mảng dữ liệu (Data array / Array) là một cách tổ chức các dữ liệu cùng cấp, cùng kiểu dữ liệu, cùng chức năng thành một nhóm.

Cách khai báo

Cú pháp khai báo

data_type, **DIMENSION**(array_length) :: array_name

hoặc

data_type :: array_name(array_length)

Ví dụ:

CHARACTER, DIMENSION(7) :: dwarf

CHARACTER ::dwarf(7)

Làm quen với Fortran

└ Chuỗi - mảng dữ liệu

└ Cách khai báo & xử lý

Ví dụ 2

Viết chương trình tính giá trị trung bình của 10 số

PROGRAM Ave2

implicit none

REAL, DIMENSION(10) :: x

REAL :: s, average

INTEGER :: i

Làm quen với Fortran

└ Chuỗi - mảng dữ liệu

└ Cách khai báo & xử lý

Ví dụ 2 (t.t)

DO i=1,10

WRITE(,*)"Nhập vào số thu ",i," : "*

READ(,*) x(i)*

s = s + x(i)

END DO

average = s / (i-1)

WRITE(,*)"Giá trị trung bình là ",average*

END PROGRAM Ave2

Luyện tập

- 1 Viết chương trình tìm số lớn nhất, nhỏ nhất trong một dãy số nguyên nhập từ bàn phím.
- 2 Viết chương trình tìm số lớn nhất, nhỏ nhất trong một dãy số nguyên và cho biết các số này ở vị trí thứ bao nhiêu trong dãy số.
- 3 Viết chương trình nhập vào một dãy số nguyên và một số nguyên n , cho biết số này xuất hiện bao nhiêu lần trong dãy số trên và ở những vị trí nào.
- 4 Viết chương trình nhập vào 3 số nguyên, in ra màn hình ba số đó theo thứ tự tăng dần (hoặc giảm dần).
- 5 Viết chương trình nhập vào một dãy số nguyên, in ra màn hình dãy số đó theo thứ tự tăng dần (hoặc giảm dần).

Làm quen với Fortran

└ Chuỗi - mảng dữ liệu

└ Mảng nhiều chiều

Mảng nhiều chiều

Ta có thể mở rộng khái niệm của mảng nhiều chiều với cách khai báo như sau:

Cú pháp khai báo

data_type, **DIMENSION**(length1,length2,...) :: array_name

hoặc

data_type :: array_name(length1,length2,...)

Làm quen với Fortran

└ Chuỗi - mảng dữ liệu

└ Mảng nhiều chiều

Ví dụ mảng hai chiều

Viết chương trình nhập vào một ma trận 2×2 và tính định thức của ma trận đó

PROGRAM Matrix

implicit none

REAL, DIMENSION(2,2) :: A

REAL :: dinhthuc

INTEGER :: i,j

Làm quen với Fortran

└ Chuỗi - mảng dữ liệu

└ Mảng nhiều chiều

Ví dụ mảng hai chiều (t.t)

DO i=1,2

DO j=1,2

READ(,*) A(i,j)*

END DO

END DO

*dinhthuc=A(1,1)*A(2,2)-A(1,2)*A(2,1)*

WRITE(,*)"Gia tri dinh thuc la",dinhthuc*

END PROGRAM Matrix

Mục lục

- 1** Mục tiêu
- 2** Các bước xây dựng một chương trình
- 3** Cấu trúc điển hình của một chương trình
- 4** Vận hành một chương trình Fortran 95
- 5** Các phép toán & các kiểu dữ liệu
- 6** Câu lệnh điều kiện
- 7** Vòng lệnh lặp
- 8** Chuỗi - mảng dữ liệu
- 9** Xuất - nhập dữ liệu với tập tin ngoài
- 10** Hàm & Chương trình con

Làm quen với Fortran

- └ Xuất - nhập dữ liệu với tập tin ngoài
 - └ Xuất dữ liệu ra tập tin ngoài

Cú pháp xuất dữ liệu ra tập tin ngoài

Cú pháp khai báo

```
OPEN(label,file='out.txt')  
WRITE(label,*)var
```

Làm quen với Fortran

└ Xuất - nhập dữ liệu với tập tin ngoài

 └ Ví dụ xuất dữ liệu ra tập tin ngoài

Ví dụ 1

Viết chương trình tính x^n của một số, với $n = 1, \dots, 10$ và x nhập từ bàn phím.

```
PROGRAM Power
IMPLICIT NONE
REAL :: x
INTEGER :: n
WRITE (*,*) "Nhap so "
READ(*,*)x
DO n=1,10
    WRITE(*,*)n , x**n
END DO
END PROGRAM Power
```

Làm quen với Fortran

└ Xuất - nhập dữ liệu với tập tin ngoài

 └ Ví dụ xuất dữ liệu ra tập tin ngoài

Ví dụ 2

Viết chương trình tính x^n của một số, với $n = 1, \dots, 10$ và x nhập từ bàn phím.

```
PROGRAM Power
IMPLICIT NONE
REAL :: x
INTEGER :: n
WRITE (*,*) "Nhập số ";READ(*,*)x
OPEN(10,file='output.dat')
DO n=1,10
    WRITE(10,*)x**n
END DO
END PROGRAM Power
```

Làm quen với Fortran

└ Xuất - nhập dữ liệu với tập tin ngoài

 └ Nhập dữ liệu từ tập tin ngoài

Cú pháp nhập dữ liệu từ tập tin ngoài

Để nhập dữ liệu từ một tập tin ngoài, ta cần chuẩn bị một tập tin chứa dữ liệu cần nhập, tạm gọi là tập tin input.

Thông thường, các tập tin input thường được đặt dưới dạng .txt, hoặc .dat.

Cú pháp khai báo

OPEN(label,file='input.txt')

READ(label,*)var

Làm quen với Fortran

└ Xuất - nhập dữ liệu với tập tin ngoài

 └ Ví dụ nhập dữ liệu từ tập tin ngoài

Ví dụ 1

Viết chương trình tính diện tích tam giác khi biết độ dài 3 cạnh

```
PROGRAM Triangle
```

```
IMPLICIT NONE
```

```
REAL :: x,y,z
```

```
REAL::s,areasq,area
```

```
WRITE (*,*) "This program calculates the area of a triangle."
```

```
OPEN(10,file='input.dat')
```

```
READ(10,*)x,y,z
```

```
WRITE (*,*) "Check: you have input the following lengths"
```

```
WRITE (*,*) x,y,z
```

Làm quen với Fortran

└ Xuất - nhập dữ liệu với tập tin ngoài

└ Ví dụ nhập dữ liệu từ tập tin ngoài

Ví dụ 1 (t.t)

```
s = 0.5 * (x+y+z) ! Semiperimeter
areasq = s*(s-x)*(s-y)*(s-z) ! Square of the area
IF (areasq<0.0) THEN
    WRITE (*,*) "Error: that is not a real triangle"
ELSE
    area = SQRT(areasq)
    WRITE (*,*) "The area of the triangle is ", area
END IF
END PROGRAM Triangle
```

Trong đó, file ‘input.dat’ chứa dữ liệu là 3 số đo cạnh tam giác, ngăn cách bởi khoảng trắng.

Luyện tập

- 1 Viết chương trình để ghi kết quả của phép tính \sqrt{n} ra tập tin ngoài dạng 2 cột, cột thứ nhất là giá trị của n từ 1 đến 20, cột thứ hai là giá trị tương ứng của \sqrt{n} (n là số nguyên).
- 2 Viết chương trình để từ tập tin đã tạo ở bài trên, hãy tính giá trị phương sai và độ lệch chuẩn của \sqrt{n} .
- 3 Viết chương trình nhập một ma trận 3×3 từ một tập tin ngoài và ghi ma trận đó lên màn hình.
- 4 Viết chương trình tính tổng của hai ma trận 3×3 từ 2 tập tin ngoài, và ghi ma trận kết quả lên một tập tin khác.
- 5 Viết chương trình tính tích của hai ma trận 4×4 từ 2 tập tin ngoài, và ghi ma trận kết quả lên một tập tin khác.

Mục lục

- 1 Mục tiêu**
- 2 Các bước xây dựng một chương trình**
- 3 Cấu trúc điển hình của một chương trình**
- 4 Vận hành một chương trình Fortran 95**
- 5 Các phép toán & các kiểu dữ liệu**
- 6 Câu lệnh điều kiện**
- 7 Vòng lệnh lặp**
- 8 Chuỗi - mảng dữ liệu**
- 9 Xuất - nhập dữ liệu với tập tin ngoài**
- 10 Hàm & Chương trình con**

Làm quen với Fortran

└ Hàm & Chương trình con

 └ Hàm (Function)

Ví dụ về cấu trúc hàm

FUNCTION Dress Cinderella

Cinderella's dress is old.
The fairy has a magic wand.

Turn the old dress into new one.
Give Cinderella the glass slippers.
Cinderella is now dressed beautifully.

END FUNCTION Dress Cinderella



Làm quen với Fortran

└ Hàm & Chương trình con

 └ Hàm (Function)

Cách khai báo hàm

FUNCTION *Func-name* (*var1, var2, ...*)

data-type :: *Func-name*

data-type :: *var1, var2, ...*

...

Func-name = ...

END FUNCTION *Func-name*

Làm quen với Fortran

└ Hàm & Chương trình con

 └ Hàm (Function)

Cách khai báo hàm

Ta có thể có cách khai báo hàm như sau:

data-type **FUNCTION** *Func-name* (*var1, var2, ...*)

data-type :: *var1, var2, ...*

...

Func-name = ...

END FUNCTION *Func-name*

Làm quen với Fortran

└ Hàm & Chương trình con

 └ Hàm (Function)

Sử dụng hàm trong chương trình

PROGRAM Prog-name

Declarative statements

Executive statements

END PROGRAM prog-name

Function 1 declarative statements

Function 2 declarative statements

Làm quen với Fortran

└ Hàm & Chương trình con

 └ Hàm (Function)

Ví dụ 1

FUNCTION Cot(x)

REAL::Cot

REAL::x

IF (x==0.0) THEN

WRITE(,*)"Error: cotangent function called &*

& with zero argument"

Cot = HUGE(x)

ELSE

Cot = COS(x) / SIN(x)

END IF

END FUNCTION Cot

Làm quen với Fortran

└ Hàm & Chương trình con

 └ Hàm (Function)

Ví dụ 2

Viết chương trình tính phần thể tích nằm giữa hai mặt cầu đồng tâm, bán kính là R_1 và R_2 .

PROGRAM Volume1

implicit none

REAL:::r1,r2,Vol,VolSq

WRITE(,*)"Nhap hai ban kinh" READ(*,*)r1,r2*

Vol=ABS(VolSq(r1)-VolSq(r2))

WRITE(,*)"The tich la",Vol*

END PROGRAM Volume1

Làm quen với Fortran

└ Hàm & Chương trình con

 └ Hàm (Function)

Ví dụ 2 (t.t)

```
!_____  
FUNCTION VolSq(x)  
REAL :: VolSq  
REAL :: x  
VolSq=4.0/3.0*3.14*x**3  
END FUNCTION VolSq  
!_____
```

Làm quen với Fortran

└ Hàm & Chương trình con

 └ Chương trình con (Subroutine)

Ví dụ

Xét lại ví dụ tính trên, ta có thể thực hiện tính như sau

```
PROGRAM Volume1
```

```
implicit none
```

```
REAL::r1,r2,Vol1,Vol2
```

```
READ(*,*)r1,r2
```

```
CALL Vol(r1,Vol1)
```

```
CALL Vol(r2,Vol2)
```

```
WRITE(*,*)"The tich la",ABS(Vol1-Vol2)
```

```
END PROGRAM Volume1
```

Làm quen với Fortran

└ Hàm & Chương trình con

 └ Chương trình con (Subroutine)

Ví dụ (t.t)

```
!_____
SUBROUTINE Vol(radius,v)
REAL :: v
REAL :: radius
vol=4.0/3.0*3.14*radius**3
END SUBROUTINE Vol
!_____
```

Làm quen với Fortran

└ Hàm & Chương trình con

 └ Chương trình con (Subroutine)

Cách khai báo và sử dụng

Khai báo

SUBROUTINE subr-name(var1, var2,...)

 data-type1::var1,...

 data-type1::var2,...

 ...

END SUBROUTINE subr-name

Sử dụng

CALL subr-name(var1, var2,...)

Subroutine và Function

- Đều sử dụng để làm ngắn gọn và rõ ràng chương trình
- Điểm khác biệt:
 - FUNCTION cho ta 1 giá trị của hàm ứng với điều kiện nhập vào tương ứng (**trả lại 1 giá trị**).
 - SUBROUTINE có thể xem là một tập hợp các câu lệnh thực hiện trên các điều kiện nhập vào (**trả lại nhiều giá trị cho nhiều biến số**).
 - Sử dụng tốt SUBROUTINE và FUNCTION sẽ cho ta chương trình rất ngắn gọn và rõ ràng.

Làm quen với Fortran

└ Hàm & Chương trình con

 └ Chương trình con (Subroutine)

Cấu trúc chương trình chính

PROGRAM ...

Declarative statements

CALL Input(var1,var2,...)

CALL Calculate-1

CALL Calculate-2

...

CALL Output(var1,var2,...)

END PROGRAM ...

Làm quen với Fortran

└ Hàm & Chương trình con

 └ Chương trình con (Subroutine)

Cấu trúc chương trình chính - Ví dụ

Khi sử dụng tốt các FUNCTION & SUBROUTINE, chương trình chính sẽ ngắn gọn và dễ hiểu:

```
PROGRAM GiaiPTBac2
REAL::a,b,c,x,x1,x2,delta
CALL Input(a,b,c)
IF (a==0) THEN
    CALL Bac1(b,c,x)
ELSE
    CALL Bac2(a,b,c,x,x1,x2,delta)
END IF
END PROGRAM GiaiPTBac2
```

Hai SUBROUTINE Bac1 và Bac2 các bạn thử tự viết nhé.

Bài tập

- 1** Viết lưu đồ và chương trình giải phương trình bậc nhất.
Viết lại dưới dạng chương trình con.
- 2** Viết lưu đồ và chương trình giải phương trình bậc hai. Viết lại dưới dạng sử dụng chương trình con.
- 3** Viết chương trình đọc một dãy số nguyên dương từ bàn phím, chương trình sẽ yêu cầu nhập lại nếu số đó là âm hoặc bằng không (có sử dụng hàm hoặc chương trình con).
- 4** Viết chương trình con để xác định một số có phải số nguyên tố hay không. Ứng dụng viết chương trình xác định có bao nhiêu số nguyên tố trong một dãy số.
- 5** Viết chương trình tìm ước số chung lớn nhất, bội số chung nhỏ nhất của hai số tự nhiên có sử dụng hàm hoặc chương trình con.