

# Fortran95 程序设计

彭国伦 编著

第3章 Fortran程序设计基础

## 3-1 字符集

►“字符集”—编写Fortran程序时所能使用的所有字符及符号。  
基本的**FORTRAN 95**字符集：

|         |   |
|---------|---|
| 英文26个字母 | A~Z 及 a~z （英文字母大小写不分）   |
| 数字      | 0~9   |
| 22个特殊符号 | : 冒号 =等号 !叹号 +加号 ” 引号 -减号 % 百分号<br>*星号 & 英语的and / 斜线 ; 分号 (左括号 <小于<br>) 右括号 > 大于 , 逗号 ?问号 .小数点或句号<br>\$ 货币符号 ' 撇号 _下划线 空格 |



## 3-2 书面格式

Fortran程序代码的编写格式有两种：

- Free Format（自由格式） [ex0301.for]
  - ◆ Fortran90后的新写法，取消很多的旧限制。
  - ◆ 往往以.f90 .f95为扩展名
  - ◆ 建议使用
- Fixed Format（固定格式） [ex0302.f90]
  - ◆ 旧式写法，编写版面有很多限制。
  - ◆ 往往以.for 为扩展名
  - ◆ 不推荐使用



## 3-2-1 固定格式

|          |                                    |
|----------|------------------------------------|
| 第1个字符    | 如果是字母c、C或*，这一行文本会当成说明或批注，不会被编译。    |
| 第1~5个字符  | 如果是数字，就是用来给这一行程序代码取个代号，不然只能是空格。    |
| 第6个字符    | 如果是非0非空的任何字符，表示这一行程序会接续上一行。        |
| 第7~72个字符 | Fortran程序代码的编写区域。                  |
| 第73个字符以后 | 不使用这些列，语句超过该列的部分会被忽略，有的编译器会发出错误信息。 |



## ex0301.for

|   |   |   |  |  |  |   |                     |  |
|---|---|---|--|--|--|---|---------------------|--|
| c |   |   |  |  |  |   | Fixed Format Demo   |  |
|   |   |   |  |  |  |   | Program main        |  |
|   |   |   |  |  |  |   | Write (*,*) 'Hello' |  |
|   |   |   |  |  |  |   | Write (*,*)         |  |
|   |   |   |  |  |  | 1 | 'Hello'             |  |
| 1 | 0 | 0 |  |  |  |   | Write (*,*) 'Hello' |  |
| 1 | 0 |   |  |  |  |   | Stop                |  |
|   |   |   |  |  |  |   | end                 |  |

程序代码命令之间的空格，不会有任何意义。FIXED FORMAT是为了配合早期使用穿孔卡片输入程序所发明的格式。



## 3-2-2 自由格式

基本上允许非常自由的编写格式，它没有规定每一行的前几个字符有什么作用，但要注意：

1. ! 感叹号后面的文本都是注释。
2. 每行可以编写132个字符。
3. 行号放在每行程序的最前面。
4. 一行程序代码的最后如果是符号&，代表下一行程序会和这一行连接。如果一行程序代码的开头是符号&，代表它会和上一行程序连接。
5. 程序命令之间的空格，无意义。



## ex0302.f90

```
! Free Format
```

```
Program main
```

```
Write& (*,*) "Hello" ! 这个也是注释
```

```
Write (*,*) &
```

```
"Hello"
```

```
Wri&
```

```
&te(*,*) "Hello"
```

```
end
```



## 3-3 Fortran的数据类型

数据类型是指使用Fortran在计算机内存中记录文本、数值等数据的最小单位及方法。

### 1. 整数 (INTEGER)

长整形: 32bits (4bytes)  $-2^{31}+1 \sim 2^{31}$

短整型: 16bits (2bytes)  $-2^{15}+1 \sim 2^{15}$  -32768~ +32767

### 2. 浮点数 (REAL)

单精度: 32bits (4bytes) 6~7位  $\pm 3.4 * 10^{38}$   $\pm 1.18 * 10^{-38}$

双精度: 64bits (8bytes) 15~16位  $\pm 1.79 * 10^{308}$   $\pm 2.23 * 10^{-308}$

### 3. 复数 (COMPLEX)

复数由两个浮点数来做记录

单精度复数和双精度复数两种





## 3-3 Fortran的数据类型

### 4. 字符型 (CHARACTER)

可记录任何从键盘输入的东西

记录一连串字符时称“字符串”

### 5. 逻辑判断 (LOGICAL)

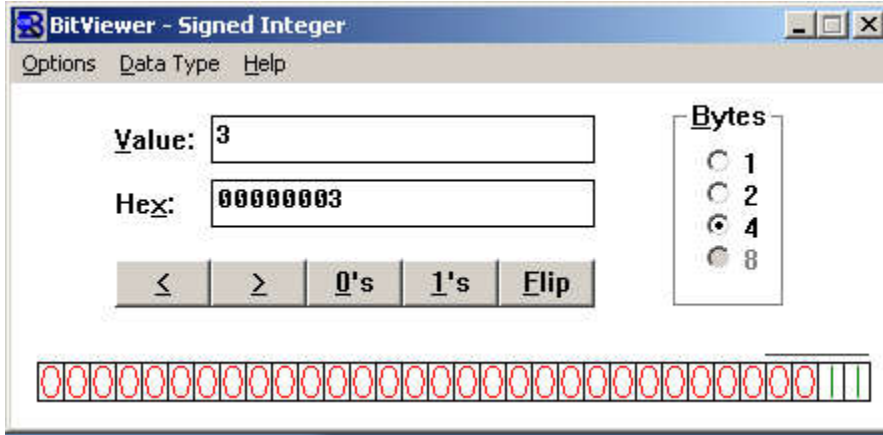
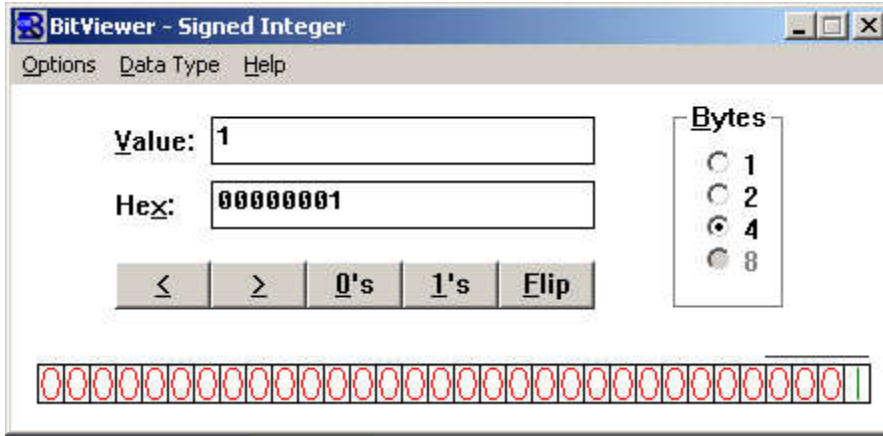
只有两种逻辑结果：“是” TRUE 和 “否” FALSE

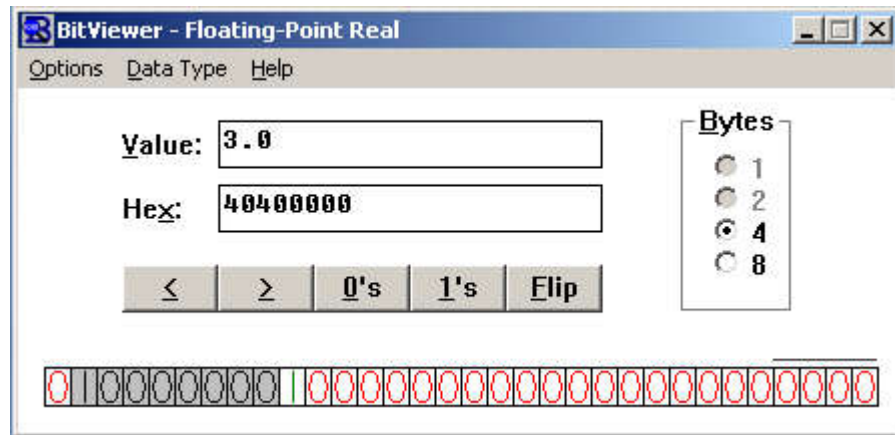
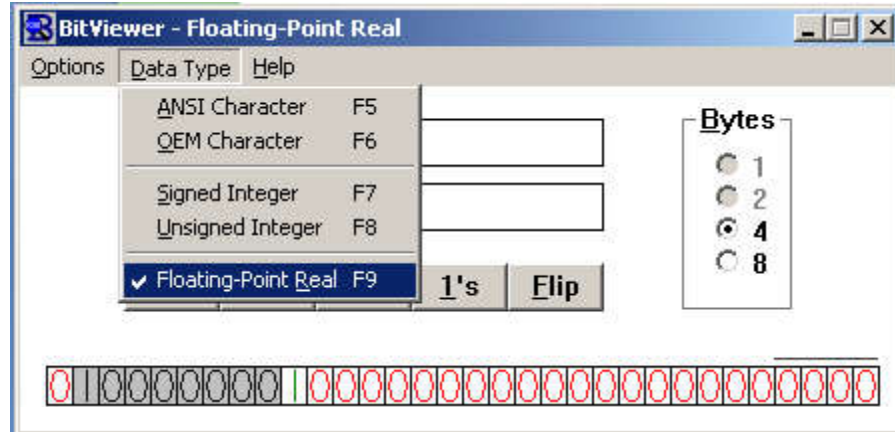
对-错 真-假

二进制 1 - TRUE 0 - FALSE

系统自带的BITVIEWER 可观看各种数据格式实际在内存中的二进制数据







## 3-4 Fortran的数学表达式

FORTRAN使用的数学运算符优先级（上低下高排序）

+加法 - 减法

\*乘法 / 除法

\*\* 乘幂（两星号要连续）

( ) 括号（表示括号起来的部分优先计算）

注意：

乘幂要连用两个星号；

乘号不能省略；

除法用计算机编写时没有下面的表示方法：

$$\frac{(A + B) * (C + D)}{2 * (E + F)}$$



## 3-5 Fortran语句结构

- Fortran程序由一系列语句组成，这些语句完成程序员希望完成的任务。
- 语句有两种基本类型：

注释语句(非语句行)：！后的所有字符都被编译器忽略  
可独占一行，可在其它语句之后，  
空行为注释行（固定格式用C和\*）

语句行：可执行语句：输入、赋值、输出  
说明语句：用于说明变量的类型、属性等。

- Fortran语句可以放在书写行的任意位置，每行可以长达132个字符。如果语句太长不适合放在一行，可以用&符合标记，在下一行继续这一行的书写，直到结束。如果需要，一条Fortran95语句可以写在40行上。



## 3-5 Fortran语句结构

- 以数字开头的独立语句，这些数字称为语句标号(statement label)。语句标号可以是1~99999之间的任何一个数字，它是Fortran语句的“名字”，使用它可以在程序的其他地方引用这条语句。
- 语句标号除了是语句的“名字”外，没有其他特殊的地方。它既不是行号，也不能说明语句的执行顺序。
- 现代Fortran程序中很少使用语句标号，大多数Fortran95/2003都没有语句标号。
- 在程序单元中使用语句标号，必须是唯一的。
- ! 号后面跟随的字符都是对程序的注释(comment)，Fortran编译器不编译注释语句。



## 3-6 Fortran程序结构

- 每一个Fortran程序由可执行和不可执行语句组成，它们按特定的顺序排列。

```
Program My_first_program
```

```
! Purpose:
```

```
! To illustrate some of the basic features of a Fortran program
```

```
! Declare the variables used in this program
```

```
Integer :: I, j, k
```

```
! All variables are integers
```

```
! Get two values to store in variables I and j
```

```
Write (*,*) 'Enter the numbers to multiply:'
```

```
Read (*,*) I,j
```

```
! Multiply the numbers together
```

```
K=i*j
```

```
! Write out the result
```

```
Write (*,*) 'Result=',k
```

```
! Finish up
```

```
Stop
```

```
End program my_first_program
```

**声明部分：**由不可执行语句组成，位于程序的开头，定义程序名和程序引用的数据以及变量的类型。

**执行部分：**由一条或多条可执行语句组成，描述程序将完成的操作。

**终止部分：**由Stop和End program语句组成，stop语句告诉计算机停止运行，End program语句告诉编译器程序中不再有语句需要编译。



## 3-7 Fortran简史

- Fortran的起源，1954年IBM的一项计划，John Backus 在IBM704上开发程序。1957年交出第一套Fortran编译器，从而诞生了Fortran语言。
- Fortran ← Formula Translator 数学公式翻译
- 1966年美国国家标准局ANSI 统一了Fortran语言标准，即Fortran66
- 1977年制订了新的F77，特点是引入结构化设计。
- 1992年公布F90，是F77的增集，加入面向对象、指针数组等功能，并改良了语法编写格式。
- 1997年公布F95，是F90的修正版，并开始支持并行运算。
- 最新的200X仍未正式推出。

