

扩 展 FORTRAN

参 考 手 册

21

石化部石油地球物理勘探局计算中心站

1977.5.

目 录

第 一 部 分

第一节	编写 FORTRAN 语句	(1)
	FORTRAN 字符集	(1)
	FORTRAN 语句写法	(1)
	行之延续	(2)
	语句之分隔	(2)
	语句标号	(2)
	注释行	(2)
	空白卡	(2)
	数据卡	(2)
第二节	语言元素	(4)
一、	常数和变量	(4)
二、	变量名字	(9)
三、	数组	(9)
第三节	表达式	(10)
一、	算术表达式	(14)
二、	关系表达式	(18)
	关系表达式之计算	(18)
三、	逻辑表达式	(19)
四、	掩码表达式	(21)
第四节	赋值语句	(23)
	算术赋值语句	(23)
	逻辑赋值语句	(25)
	掩码赋值	(26)
	多重赋值	(26)
第五节	控制语句	(27)
一、	转向语句	(27)
	无条件转向语句	(27)

计算转向语句.....	(28)
GOTO 赋值语句.....	(29)
赋值转向语句.....	(29)
二、算术 IF 语句.....	(30)
三支情形.....	(30)
二支情形.....	(31)
三、逻辑 IF 语句.....	(31)
二支情形.....	(32)
四、DO 语句.....	(32)
五、循环转移.....	(34)
六、继续语句.....	(37)
七、暂停语句.....	(37)
八、停语句.....	(38)
九、终止语句.....	(38)
十、返回语句.....	(39)
第六节 说明语句.....	(41)
一、类型语句.....	(41)
显说明类型.....	(41)
隐类型说明语句.....	(43)
下标.....	(44)
二、维数语句.....	(44)
变量数组.....	(45)
三、公共语句.....	(46)
四、等价语句.....	(49)
五、公共语句和等价语句.....	(52)
六、LEVEL 语句.....	(53)
七、外部语句.....	(55)
八、数据语句.....	(57)
九、数据块子程序.....	(60)
第七节 程序单位.....	(61)
一、主程序和子程序.....	(61)
二、主程序.....	(61)
三、程序语句.....	(62)
四、子程序.....	(64)
第八节 FORTRAN 库.....	(79)
一、内部函数.....	(82)
二、外部函数.....	(82)

三、附加的应用子程序.....	(85)
子程序.....	(85)
函数.....	(87)
第九节 输入/输出.....	(90)
一、输出语句.....	(91)
二、穿孔语句.....	(92)
三、输入语句.....	(94)
四、文件操作语句.....	(95)
五、反绕语句.....	(96)
六、缓冲语句.....	(96)
七、缓冲输入.....	(97)
八、缓冲输出.....	(98)
九、名字表.....	(98)
十、ENCODE 和 DECODE.....	(103)
第十节 输入/输出表和格式语句.....	(109)
一、输入/输出表.....	(109)
二、数组传送.....	(110)
三、格式语句.....	(113)
四、数据转换.....	(113)
比例因子.....	(128)
记录的斜撇/FORTRAN.....	(133)
重复格式规格.....	(134)
打印机控制符.....	(135)
执行时的格式语句.....	(138)
第十一节 FORTRAN 控制卡.....	(138)
I 源输入参数.....	(139)
L 列表控制.....	(140)
E 编辑参数.....	(140)
T 错误追溯.....	(141)
舍入算术运算开关.....	(141)
D 检错方式参数.....	(141)
A 出口参数.....	(141)
S 系统文本文件.....	(141)
GT 得到系统文本文件.....	(142)
SYSEDIT 系统编辑.....	(142)
V 小缓冲的选择.....	(142)
C COMPASS 汇编.....	(142)

R 符号参考选择.....	(142)
PL 打印行限制.....	(142)
Q 程序检验.....	(142)
Z 零参数.....	(142)
XT 外部内容名字.....	(143)
OPT 优化参数.....	(143)
第十二节 复盖程序段.....	(147)
复盖段.....	(147)
复盖段的连接.....	(149)
产生一个复盖段.....	(150)
调入复盖段.....	(151)
第十三节 检错手段.....	(155)
检错语句.....	(157)
延续卡片.....	(157)
ARRAYS 语句.....	(158)
CALLS 语句.....	(160)
FUNCS 语句.....	(163)
STORES 语句.....	(166)
GOTOS 语句.....	(171)
TRACE 语句.....	(172)
NOGO 语句.....	(175)
检错卡片组的结构.....	(175)
DEBUG 语句.....	(179)
AREA 语句.....	(181)
OFF 语句.....	(183)
打印检错输出.....	(184)
STRACE (跟踪).....	(184)

第二部分 典型程序

一、程序 OUT	(186)
二、程序 B	(187)
三、程序 MASK	(189)
四、程序 EQUIV	(191)
五、程序 COME	(192)
六、程序 LIBS	(194)
七、程序 PIE	(196)

八、程序 ADD.....	(197)
九、程序 PASCAL.....	(200)
十、程序 X.....	(202)
十一、程序 VARDIM.....	(203)
十二、程序 VARDIM2.....	(204)
十三、程序 CIRCLE.....	(208)
十四、程序 OCON.....	(210)

第 三 部 分

第一节 参照图表.....	(212)
一般格式.....	(216)
符号的性质.....	(216)
入口点.....	(216)
变量.....	(217)
文件名字.....	(218)
外访.....	(219)
INLINE函数.....	(220)
名字列表.....	(220)
语句标号.....	(221)
DO循环.....	(221)
公用区.....	(222)
程序总计.....	(223)
第二节 FORTRAN 之诊断.....	(224)
编译诊断.....	(224)
执行时的诊断.....	(237)
第三节 系统子程序之说明.....	(243)
访问 SYSTEM	(243)
用 SYSTEM 处理文件名字.....	(245)
第四节 运算.....	(247)
浮点运算.....	(247)
计算错误.....	(253)
追踪计算错误	(255)
第五节 执行时产生的输入/输出.....	(256)
输入/输出文件之结构.....	(256)
记录管理.....	(257)
有标号的文件.....	(263)

	缓冲区输入/输出.....	(263)
	状态检查.....	(264)
	奇偶错误之检测.....	(266)
	数据输入错误的控制.....	(266)
	编程注意事项.....	(268)
第六节	FORTRAN 与记录管理之间的联系.....	(269)
	文件信息表的访问.....	(269)
	文件命令.....	(270)
	查错.....	(275)
第七节	大存贮输入/输出.....	(276)
	随机文件的存取.....	(278)
	目录键的类型.....	(279)
	多级文件编目.....	(281)
	目录类型.....	(282)
	错误信息.....	(282)
	与以前的大存贮子程序的兼容性.....	(285)
第八节	重新命名约定.....	(285)
	寄存器的名字.....	(285)
	外部程序名字.....	(285)
第九节	程序和内存结构.....	(286)
	子程序和函数结构.....	(287)
	主程序结构.....	(287)
	内存结构.....	(287)
第十节	混有COMPASS的子程序.....	(288)
	COMPASS码子程序.....	(288)
	用名字进行调用.....	(288)
	入口点.....	(289)
	COMPASS子程序.....	(289)
	用值调用库函数.....	(290)
	对使用库函数名字的限止.....	(291)
第十一节	FORTRAN — 终端接口.....	(294)
第十二节	列表.....	(294)
	DMPX.....	(295)
第十三节	卡片迭结构举例.....	(299)
	附录A 标准的SCOPE符号组.....	(308)

第 一 部 分

第一节 编写 F O R T R A N 语句

FORTRAN 程序包括执行语句和非执行语句。执行语句指明程序要采取的动作；非执行语句则描述操作对象的特性、语句函数、数据的安排以及数据的格式。

FORTRAN 源程序写法如图 1 所示。每一行代表一个有 80 列的卡片。FORTRAN 字符集用来编写各种语句。

FORTRAN 字符集：

字母 A 到 Z

数字 0 到 9

特殊符号	=	等号	)	右括号
	+	加号	,	逗点
	-	减号	.	小数点
	*	星号	\$	美元符号
	/	斜杠		空白符
	(	左括号	≠ 或 '	行引号

除此之外，任何字符（附录 A）可用在文字常数和注释中，空白符在文字行外是没有意义的。

FORTRAN 语句写法：

第 1 列	C 或 \$ 或 * 指出这一行为注释行 在检错方式中，在 1—2 列为 C\$，说明为调试语句
第 1—5 列	语句标号
第 6 列	空白或 0 以外之字符表示为延续行；但这不能用作注释行之延续， 调试延续卡应包含 C\$ 于第 1—2 列
第 7—72 列	语句
第 73—80 列	标识字段，编译程序不对之加工

行之延续

语句是写在 7—72 列上。一个语句在这样 66 个列上写不下时，可以延续到另一行（称为延续行），最多可有 19 个延续行。在延续行之第 6 列写一个空白或 0 以外之字

符,表示其为延续行,第1列可包含不同于C、*或\$之字符。2、3、4、5列可包含任何字符。除注释语句和OVERLAY(覆盖)语句外,任何语句都可以这样延续,包括END语句。

语句之分隔

几个短的语句可以写在一行,每两个语句间用\$分开。随在\$后面之语句和分开写成一行之语句完全一样处理。例如

```
7
ACUM = 24.  $ I = 0  $ IDIFF = 1970 - 1626
```

和下列语句是一样的

```
7
ACUM = 24.
I = 0
IDIFF = 1970 - 1626
```

这种用\$分开语句的方法可用于各种语句,除去格式语句和调试语句外。\$后面之语句不能有标号,除此之外,与穿在下一行第7列开始之语句处理起来没有任何不同。

语句标号

语句之第一行(如有延续行的话)的1—5列可用来写语句标号,任何语句都能加标号,但只有格式语句和可执行语句的标号可被别的语句所引用。语句标号是1—99999之整数,它们可以按任意序列出现。左零和左面之空白及嵌在中间的空白都没有任何意义。每个语句标号在其所在之程序体内必须是唯一的,不能从另外的程序体内来引用这个标号。在图1中语句标号是4、1、2、3。

注释行

在第一列如为C、*或\$表示这是一个注释行。注释并不影响程序的功效。它们写第2列到80列上,并可放在程序中任何地方。如注释在一行中写不下时,可以多于一行,但每行之第1列必须写C、*或\$。在注释行(卡)中第6列不再看成延续行之符号。注释也能插在一个语句的两个延续行中而并不中断语句之连续性。

第73—80列。任何信息都可写在73—80列上,它们并不是语句的一部分,这些列上的信息在源程序打印时将照样印出,但不被编译程序加工,它们用来整理一迭穿孔卡片之次序。

空白卡

如用一个空白卡去分开语句,在打印源程序时产生一个空行。随在空白卡之后的行处理成一个新行。因此延续卡间不能插入空白卡。

空白卡也不能随在END语句后面,否则打印诊断信息。

数据卡

由源程序读入之数据卡的格式没做任何限制,数据可写在1—80列上,但数据卡上73—80列是不予理采的。

第二节 语言元素

一、常数和变量

常数

常数是一个固定的量。常数的七种类型为：整型、实型、双精度实型、复型、八进制数型、文字型和逻辑型。

整常数

形如

$n_1 n_2 \dots n_m$

其中 n 为数字, $1 \leq m \leq 18$ 之十进数字。

例子:

237 -74 +136772 0 -0024

一个整常数是由 1—18 位十进数字的行构成, 其中不得有小数点。它可以是正数、负数或零。如一个整数为正的, 则 “+” 号可省略; 如它为负的, “-” 号必须写出。一个整常数不得包含有逗号。整常数之范围为 $-2^{59}-1$ 到 $2^{59}-1$ ($2^{59}-1=576\ 460\ 752\ 303\ 423\ 487$)。

不合法的整常数的例子如:

46. (不允许有小数点)

23A (不允许有字母)

7,200 (逗号不允许使用)

当整常数用作数组下标, DO 语句或隐循环的指标计数时, 最大值为 $2^{17}-2$ ($2^{17}-2=131070$), 最小值为 1。

整数在乘和除时缩短到 48 位。整数进行乘、除运算之结果应小于 $2^{48}-1$ 。如结果大于 $2^{48}-1$ ($2^{48}-1=281474976710655$) 高位就要失去, 并且不提供诊断信息。在实型常数转换成整型 (或反之) 时, 结果最大值为 $2^{48}-1$ 。如超过这个值高位将失去, 也不提供诊断信息, 对于加法和减法, 是对全字长 60 位进行的。

实常数

$n.n \quad n. \quad n.nE \pm s \quad .nE \pm s \quad n.E \pm s \quad nE \pm s$

其中 n 为系数, ≤ 15 位十进数。 $E \pm S$ 为指数部份, 如 S 为正数, “+” 号可省略, S 为幂次 (以 10 为底)。

一个实数是一个十进数字的行, 可有小数点或指数部分, 或者两者全有。逗号是不允许的, 如为正数, “+” 号可省略不写。

实常数的范围是从 10^{-230} 到 10^{+222} ; 如超过这个范围, 将打印诊断信息。实常数

精度近似为 14 位十进数，它在计算机内部存贮在一个机器字中。

例子：

7.5 -3.22 +4000. 23798.14 .5 -.72 42.E1 700.E-2

不正确写法的例子：

3,50. (不允许使用逗号)

2.5A (不允许使用字母)

实常数可以在后面跟一个10的方幂，写为字母E及一个整常数。它表示要乘上以10为底、以这个整常数为幂次之指数部分。跟在E后面之幂次范围可为0，但不能省略不写。如指数为正，则“+”号可省略不写；如为负数，“-”号却一定要写，不能省略。

例子：

42.E1 ($42. \times 10^1 = 420.$)

.00028 E+5 ($.00028 \times 10^5 = 28.$)

6.205 E12 ($6.205 \times 10^{12} = 6205000000000.$)

8.0 E+6 ($8. \times 10^6 = 8000000.$)

700.E-2 ($700. \times 10^{-2} = 7.$)

7 E 20 ($7. \times 10^{20} = 700000000000000000000.$)

不正确的实常数写法之例子：

7.2 E 3.4 指数部分不是一个整数

双精度常数

$n.nD \pm s$ $.nD \pm s$ $n.D \pm s$ $nD \pm s$

其中n为系数。D±s为指数部分，若s为正数则“+”号可省略。s是以10为底之幂次。

双精度常数写法极类似实常数，除了指数部分是用字母D而不是E加以指明这一点不同。双精度值在机器内部是用两个机器字表示的，用以给出更高的精度。双精度常数大约有29位十进数精度。

例子：

5.834 D 2 ($5.834 \times 10^2 = 583.4$)

14.D-5 ($14. \times 10^{-5} = .00014$)

9.2 D 03 ($9.2 \times 10^3 = 9200.$)

-7.D 2 ($-7. \times 10^2 = -700.$)

3120.D 4 ($3120 \times 10^4 = 31200000.$)

不正确写法的双精度常数之例子：

7.2 D 缺少指数

D 5 只有指数部分是不允许的

2, 1.3 D 2 有逗号不合法

3.141592653589793238462643383279 缺少 D 和指数部分。

复型常数

(r_1, r_2)

其中 r_1 为实部, r_2 为虚部, 每一部分都和实数有相同之范围。

复常数写法为用一对括号将实数偶括起来, 两实数中间用逗号隔开。

FORTRAN 写法

复数

(1., 7.54) $1. + 7.54 i$ $i = \sqrt{-1}$

(-2.1E1, 3.24) $-21. + 3.24 i$

(4.0, 5.0) $4.0 + 5.0 i$

(0., -1.) $0.0 - 1.0 i$

第一个实数是复数之实部; 而第二个实数是表示虚部。括号是复数写法的一部分, 一定要写。每个常数均可在其前加上正号或负号。复数在机器内部是用两个相邻之机器字存贮的。

复数两部分之常数都必须为实数, 不能为整数。

不正确写法之例如下:

(275, 3.24) 275 为一个整数

(12.7D-4 16.1) 丢掉逗号且实部用了个双精度数
 $4.7E+2, 1.942$ 丢失括号

(0, 0) 0 为整数

形成复常数之两个实常数取值范围为 10^{-200} 到 10^{320} 。

八进常数

$n_1 \dots n_m B$

其中 n 为八进数字, 即 0 到 7。有 $1 \leq m \leq 20$ 个八进数字。一个八进数是由 1 到 20 个八进数字后缀一个字母 B 构成。

例子:

777777 B

52525252 B

500127345 B

不正确写法之八进数如:

892777 B 8 和 9 不是八进数字

77000000007777752525252 B 超过 20 个数字

O 7766 不允许字母 O 出现

一个八进常数即不得超过 20 个数字, 也不得包含非八进数字。如若不然, 将打印出致命的编译错误诊断信息。若少于 20 个数字, 则向右对齐, 并用 0 来填满其余位。八进数可以用到整常数可能应用的地方, 但它们不能用作语句标号及引用语句标号, 不

能用在 FORMAT (格式) 语句中, 也不能说明一个文字常数之字符个数。

它们可以用到 DO 语句、表达式中、数据语句和维数说明中。

例子:

BAT = (I * 5252 B) * OR * JAY 掩码表达式

J = MAXO (I, 1000 B, J, K + 40 B) 八进常数用作函数之参数

NAME = I * AND * 77700000 B 掩码表达式

J = (5252 B + N) / K 算术表达式

DIMENSION BUF (1000 B) 维数说明

当八进常数用在表达式中时, 则假定表达式的类型为表达式中最优势操作数的类型
(表 3-1, 节 3)

文字常数

nHf	nLf
nRf	≡f≡

其中 n 为无符号十进整数, 表示行中符号个数, 必须大于 0。当用于表达式中时又需不大于 10。

f 为字符行

≡分隔符

H 左面对齐, 其余用空白补满

L 左面对齐, 其余用 0 补满

R 右面对齐, 其余用 0 补满

PROGRAM HOLL (OUTPUT)

A = 6 HABCDEF

B = 6 LABCDEF

C = 6 RABCDEF

D = ≡ABCDEF≡

PRINT 1, A, A, B, B, C, C, D, D

1 FORMAT (O 24, A 15)

STOP

END

内部存贮

显示码

01020304050655555555

ABCDEF A

01020304050600000000

ABCDEF B

00000000010203040506

ABCDEF C

01020304050655555555

ABCDEF D

一个文字常数是由一个无符号十进整数, 字母 H 和一个符号行构成。例如:

5 HLABEL

这个整数表示行中字符个数（包括空白在内）。空白在文字常数中是有意义的。例如：

```
18 HTHIS IS A CONSTANT
7 HTHE END
19 HRESULT NUMBER THREE
```

$I = (+5 \text{ HABCDE})$ 是一个正确有效的语句。 $(+5 \text{ HABCDE})$ 是一个表达式，其中“+”号是一个运算符。

nHf 型文字行

文字常数可用在算术表达式中，数据和格式语句中。子程序调用语句中作为变元，以及输入/输出语句中作为输出列表中的列表项。若在算术操作中，文字常数作为一个运算对象，将印出诊断信息。

在表达式及数据语句中，文字常数限为 10 个字符。在格式语句及作子程序之实在参数（简称实参），文字行之字符个数限为 150 个字符。

用一对 $\equiv$ 为分界之文字行可用到 H 形式文字行所能用到之一切地方。例如：

```
IF(V.EQ.  $\equiv$ YES $\equiv$ )Y=Y+1.
(PRINT 1,  $\equiv$ SQRT $\equiv$ , SQRT(4.))
PRINT 2,  $\equiv$ TEST PASSED  $\equiv$ 
INTEGER LINE(7) , N1THRU 9
LOGICAL NEWPAGE
IF (NEWPAGE) LINE (7)=  $\equiv$ PAGE 0 $\equiv$  +N 1 THRU 9
PROGRAM FL(OUTPUT)
PRINT 1,  $\equiv$ FIELD LENGTH=  $\equiv$ , IGETFL(I)
1 FORMAT (2 A 10, I 6)
END
```

在行内若用字符 $\equiv$ ，则用两个相邻之 $\equiv$ 表示之。空行例如 OH 或 $\equiv\equiv$ 是不允许的。

当文字行中字符个数是小于 10 时，机器字内是向左对齐并用空白填满的。若超过 10 个而不是 10 之整倍数，则最后一个机器字是向左对齐，并用空白将其余填满的。

例

```
7
| READ 1,NAME
1| FORMAT (A7)
| IF(NAME .EQ. 4HJOAN) GO TO 20
```

```
7
| WRITE (6,1000)
1000| FORMAT (1X, 73H NO COUNTRY THAT HAS BEEN THOROUGHLY EXPLORED IS
| 8 INFESTED WITH DRAGONS.)
```

nRf 和 nLf 型文字常数

这两种类型文字常数限为 10 个字符，且不能用在格式语句中。

逻辑常数

逻辑常数具有形式：

• TRUE • 或 • T • 表示真值

• FALSE • 或 • F • 表示假值

小数点作为常数之一部分，必须写出。

例子

LOGICAL X1, X2

⋮

X1 = .TRUE.

X2 = .FALSE.

二、变量名字

一个变量用以表示一个量，其值可以改变。这个值在程序执行中可以反复改变。变量是用符号名字加以标识的。符号名是由 1—7 个字母或数字组成，第一个必须为字母，一个变量是和—个存贮位置相联系的。无论变量用到与否，总是假定当前之值是放在这个存贮位置中。变量的五种类型为：逻辑型、整型、实型、双精度型和复型。

变量之类型是由其第一个字母隐含指出的，假若没有明确用类型语句或隐类型语句加以定义的话。若类型未加以说明，符号名第一个字母为 I, J, K, L, M 或 N 时变量为整型。

例子

IFOM JINX 2 KODE NEXT 23 M

一个变量若没有用类型语句或隐类型 (IMPLICIT) 语句加以定义的话，假若符号名第一个字母为上述 I, J, K, L, M, N 六个字母以外之字母，则认为类型为实型。

例如 RESULT ASUM A 73 BOX

不加以说明时变量类型如下表：

A—H, O—Z	实型
I—N	整型

整变量

一个整变量名字由 1—7 个字母、数字组成，其第一个字母必须为 I、J、K、L、M 或 N，若没有明确对其类型加以说明。

变量值之范围为 $-2^{50}-1$ 到 $2^{50}-1$ 。当整变量用作下标或 DO (循环) 语句中控制变量时, 则最大值为 $2^{17}-1$ 。当从整数向实数转换时, 整数作乘、除时, 以及在 I 格式说明下输入/输出时, 结果之绝对值应小于 2^{48} 。若值超过 2^{48} , 高位将丢失。整数作加、减法时, 结果绝对值应小于 2^{50} 。

例子

```
ITEM1    NSUM    JSUM    N72    J    K 2 SO 4
```

实型变量

实型变量之符号名由 1—7 个字母、数字组成, 第一个字母为 I、J、K、L、M 或 N 以外之字母, 且这个变量类型没有明确定义。实变量值的范围是 10^{-283} 到 10^{+822} , 大约相当于有 14 位有效数字之精度。

例子

```
AUAR    SUM 3    RESULT    TOTAL 2    BETA    XXXX
```

双精度型变量

双精度型变量名字必须明确地用类型说明加以定义。双精度型变量值的范围从 10^{-283} 到 10^{+822} 。有大约 29 位有效数字之精确度。

例子

```
DOUBLE    PRECISION    OMEGA , X, IOTA
```

则 OMEGA, X, IOTA 为双精度型变量。

复型变量

复型变量必须明显地用类型说明加以定义。一个复型变量在存储中占两个字, 每个字包括一个如同实变量形式的数, 每个数的范围从 10^{-283} 到 10^{+822} 。

例子

```
COMPLEX    ZETA, MU, LAMBDA
```

逻辑变量

逻辑变量必须明显地用类型说明加以定义。一个逻辑变量可以取值“真”或“假”, 并占据一个内存字。

例子

```
LOGICAL    L 33, PRAVDA, VALUE
```

则 L 33, PRAVDA, VALUE 为逻辑型变量名字

三、数组

一个 FORTRAN 数组是用单独一个名字加以标识的一组元素。数组中任何元素均可借助其在数组中位置加以引用。数组可为一维、二维或三维。数组名和维数必须在维数说明公共语句或类型说明中加以说明。

例子

```
PROGRAM    VARDIM (OUTPUT, TAPE 6=OUTPUT)
```