

计算机软件手册丛书

FORTRAN 5.0 使用手册

FORTRAN 5.0 使用手册

刘捐献 编

天津大学出版社

天津大学

1991.12



123.2
L573
计算机软件手册丛书

FORTRAN 5.0 使用手册

刘捐献 编

天津大学出版社

内 容 提 要

本书是为提高 FORTRAN 5.0 程序设计效率而编写的简明手册。该书系统、简洁地介绍了 FORTRAN 5.0 的命令、语句、函数(其中包括绘图和字处理函数)的使用格式与规则。为了方便使用,书中列出了有关命令和函数分类表,供读者查阅。此外,书后还附有 FORTRAN 5.0 编译、连接、运行时常见错误信息。

本书适用于初学者及软件人员的学习和使用,也可作为大、中专各类院校和科技工作者学习 FORTRAN 5.0 的参考书。

计算机软件手册丛书

FORTRAN 5.0 使用手册

刘捐献 编

天津大学出版社出版

(天津大学内)

邮编:300072

河北省永清县印刷厂印刷

新华书店天津发行所发行

*

开本:787×1092 毫米¹/₁₆ 印张: 0 字数:202千

1997年3月第一版 1997年3月第一次印刷

印数:1—5000

ISBN 7-5618-0939-5

TP·97 定价:10.00元

序 言

一个以计算机、多媒体和通信技术为主要特征的信息革命,正在我国兴起。随着信息技术的高速发展,信息科学、技术及产业已成为推动国民经济增长和社会进步的巨大推动力。邹家华副总理在国务院信息化工作领导小组第一次全体会议上,就我国信息化建设的有关问题做了重要讲话。他指出,为了推动我国经济实现两个根本转变,必须把加快信息化进程放到重要的战略地位上来,并强调,我国的信息化建设,必须立足中国国情。无论是信息基础设施、重大的信息工程项目建设,还是信息资源的开发利用,都要充分考虑我国的经济、技术基础,要立足于我国的信息技术和信息产业。

国家信息化建设,涉及各行各业、各个部门。增强全民信息化意识,培养大批信息化人才,普及信息化技术和产品,促进公众信息化能力的提高,是信息化建设的重要措施。

为满足迫切需要掌握微型机常用软件和语言的使用和操作的读者的要求,我们组织了部分在信息化领域第一线从事教学和科研的教师、学者和专家,编写了这套微型机常用软件和语言系列丛书。根据作者使用的经验,有针对性地、较全面地介绍各个领域软件的功能、特点及使用方法。本丛书涉及面广、实用性强、概括简练,可满足不同读者的要求。

本系列丛书由边莫英负责组织有关人员编写、定稿。天津大学出版社给予了大力的支持,在此表示感谢。

边莫英

1996年7月

75 16/06
前

言

自 50 年代问世以来,世界上广泛流行的 FORTRAN 语言版本不断更新,功能也不断增强。美国 Microsoft 公司推出的最新版本 FORTRAN5.0 吸取了其它高级语言的一些优点,进一步扩充完善了 FORTRAN 语言功能,使 FORTRAN 语言更加成熟。

《FORTRAN5.0 使用手册》是根据此版本在微机上使用需要而编写的。对 FORTRAN5.0 的命令、基本概念、各种类别语句等归纳于不同章节,分别做了简单扼要介绍,以便使用者学习、上机操作或编程时查阅。此手册附录 2 列出了在程序单位中语句、注释行和元命令的顺序表,更便于快速查阅。

在编写本手册过程中,得到边莫英和匙彦斌两位教授的热情帮助和指导,在此表示衷心的感谢。

由于时间仓促,水平所限,本手册中若有错误或不妥之外,敬请广大读者批评指正。

刘捐献

1997 年 1 月

目 录

第 1 章 FORTRAN 5.0 概述	(1)
1.1 FORTRAN 5.0 版本简介	(1)
1.2 FORTRAN 5.0 运行环境、安装和源程序建立	(1)
1.3 FORTRAN 5.0 的编译和连接	(2)
1.4 FORTRAN 5.0 源程序的运行	(6)
1.5 FORTRAN 5.0 程序结构	(6)
第 2 章 FORTRAN 5.0 基本元素和成分	(12)
2.1 字符集	(12)
2.2 常量和变量	(12)
2.3 数组和记录	(19)
2.4 内部函数	(26)
2.5 表达式	(32)
第 3 章 赋值语句和基本输入输出语句	(36)
3.1 赋值语句	(36)
3.2 基本输入输出语句	(36)
第 4 章 控制语句	(48)
4.1 GOTO 语句	(48)
4.2 IF 语句	(49)
4.3 分块语句	(49)
4.4 SELECT CASE、CASE 和 END SELECT 语句	(51)
4.5 循环语句和循环	(53)
4.6 STOP、PAUSE 和 END 语句	(58)
第 5 章 过程	(60)
5.1 语句函数	(60)
5.2 函数子程序	(61)
5.3 子例行程序	(62)
5.4 子程序调用时的虚实结合	(63)
5.5 等价语句、公用语句和数据块子程序	(64)
5.6 外部过程中使用的其它语句	(67)
第 6 章 文件	(69)
6.1 文件类型及访问方式	(69)
6.2 文件的输入与输出	(70)
6.3 文件的辅助输入和输出语句	(72)

第7章 元命令	(78)
7.1 元命令的分类.....	(78)
7.2 元命令的使用说明.....	(78)
第8章 FORTRAN 5.0 绘图	(84)
8.1 概述.....	(84)
8.2 图形库的使用方法.....	(86)
8.3 绘图过程.....	(87)
8.4 彩色图形适配器的彩色属性.....	(87)
8.5 图形坐标系.....	(88)
8.6 图形库子程序的使用.....	(90)
第9章 FORTRAN 5.0 字形处理	(109)
9.1 字形处理的技术术语	(109)
9.2 字形处理与使用步骤	(109)
9.3 字形子程序	(110)
附录1 FORTRAN 5.0 出错信息与提示	(113)
附录2 FORTRAN 5.0 语句、注释行和元命令的顺序	(133)

第1章 FORTRAN 5.0 概述

1.1 FORTRAN 5.0 版本简介

Microsoft FORTRAN 5.0 简称 FORTRAN 5.0, 是使用在微机上新版的 FORTRAN 版本。它是在 Microsoft FORTRAN 4.0 版本 (Microsoft FORTRAN 4.0 版本符合美国 ANSI X3.9—1978 标准 FORTRAN 77 版本) 基础上发展而来的, 完全包含了标准 FORTRAN 77 版本的所有功能。FORTRAN 5.0 汇集了其它高级语言的一些特点, 补充了许多新的内容, 使只适应于科学计算的 FORTRAN 不仅具有各种计算功能, 还具有绘图和西文文字 (字体、字形) 处理等功能, 从而使新版本的 FORTRAN 功能更加丰富和完善。

FORTRAN 5.0 与标准 FORTRAN 77 相比较, 扩充的主要内容如下:

- ①增加了双精度复型、结构 (记录) 型的数据类型和动态数组 (程序执行时, 方可确定数组的大小);
- ②数组可作为操作数 (运算量) 直接参加运算 (例如: 两个数组相加, 不用引用 DO 循环进行运算);
- ③增加 SELECT CASE、DO WHILE、EXIT 等控制语句, 使 FORTRAN 程序补充了新的控制结构;
- ④增加了一些新的内部函数, 如: 测试文件尾和位操作函数等;
- ⑤提供几个用户直接可调用的过程, 如随机数过程 (调用此过程时, 用 CALL 语句实现) 等。
- ⑥采用条件编译, 可以实现对 FORTRAN 源程序部分内容的编译;
- ⑦丰富的 FORTRAN 图形库不仅包含绘图函数, 而且还包含了西文字体输出函数。

1.2 FORTRAN 5.0 运行环境、安装和源程序建立

1.2.1 硬件配置

硬件需要 IBM PC 286 以上及其兼容机, 并有以下配置:

- ①一个硬盘和一个软盘驱动器;
- ②一个图形显示器 (GGA、EGA、VGA 等);
- ③内存容量 1MB 以上。

1.2.2 软件配置

1. 操作系统

DOS (MS DOS 或 IBM PC DOS) 版本 3.0 以上或 OS/2 (Microsoft OS/2x, IBM OS/2) 版本, 或 UC DOS 版本 3.0 以上。

2. FORTRAN 5.0 系统程序

FORTRAN 5.0 全集(SAA)或子集(VAX)。

1.2.3 FORTRAN 5.0 的安装

使用 DOS 命令或工具软件(如:PC TOOLS 软件),直接将 FORTRAN 5.0 系统程序拷贝到指定的硬盘上。

1.2.4 FORTRAN 5.0 源程序的建立

选用任意一种编辑软件(如 Worstar、EDIT、PE 等)均可编写所需要的 FORTRAN 5.0 源程序,并可对其程序进行修改。

1.3 FORTRAN 5.0 的编译和连接

对 FORTRAN 5.0 源程序的编译和连接主要是由 FL 命令同时完成的。FL 命令既可实现对源程序文件的编译,又可实现目标文件之间的连接。

FL 命令功能很强,使用格式如下:

FL [选项表] [文件名表] [/LINK] [连接库] [/MA 选项]

在具体使用本命令时,有以下几种形式。

1.3.1 最基本的编译与连接

主要是对简单的 FORTRAN 源程序进行编译和连接。使用的命令形式为:

FL [文件名表]

其中文件名表是由一个或几个(用空格分开的)FORTRAN 源程序文件名和源程序目标文件名组成(文件名中的英文字母可以大写或小写)。当文件名表省略时,对当前内存中的 FORTRAN 源程序进行编译,并完成与 FORTRAN 库的连接,自动形成一个与内存源程序文件同名的可执行文件。

例如:

FL ABC.FOR

该命令编译 ABC.FOR 源程序文件,形成 ABC.OBJ 目标文件,并与 FORTRAN 库自动连接,最后形成一个 ABC.EXE 可执行文件。

使用这种基本 FL 命令时应注意以下两点:

①被连接的源程序中若有数学浮点运算,所使用的计算机上必须配有浮点运算器,否则将出现“浮点运算器未装入”错误信息;

②被编译的 FORTRAN 源程序文件中只有常规函数,即只需与 FORTRAN 库连接,没有绘图函数和过程及字形处理函数的调用。

基本编译与连接命令举例如下。

1. FL *.FOR

将当前目录下的全部 FORTRAN 源程序文件进行编译,并分别形成与其同名的目标文件(*.OBJ),最后与 FORTRAN 库连接,形成一个以当前目录中第一个 FORTRAN 源程序文件同名的可执行文件。

2. FL A.FOR B.FOR C.FOR

作用同上,编译后,形成 A.OBJ、B.OBJ 和 C.OBJ 的目标文件,最后形成 A.EXE 的可执行文件。

3. FL A. FOR C. FOR D. OBJ

编译 A. FOR 和 C. FOR 源程序文件,形成 A. OBJ 和 C. OBJ 目标文件,并与 D. OBJ 目标文件连接,最后与 FORTRAN 库再连接形成可执行文件 A. EXE。

4. FL A. OBJ B. OBJ

因 A. OBJ 和 B. OBJ 两个文件均为目标文件,故此 FL 命令仅起目标文件和 FORTRAN 库连接的作用,最后形成 A. EXE 可执行文件。

1.3.2 带有选项的 FL 命令

带有选项的命令形式如下:

FL 选项表 文件名表

其中,文件名表同基本编译与连接形式中的规定。选项表可为如下内容。

1. 浮点运算处理方式选项

FORTRAN 5.0 中的浮点运算有三种处理方式:一是使用 80287、80387 等协处理器方式;二是软件仿真(仿真数学库)方式;三是备用数学库方式。在编译具有浮点运算的程序时,必须指明采用哪些浮点处理方式。使用此选项形式有以下几种选择(注意选项中要区分英文字母大小写,不能混淆)。

(1)/FPi

选择仿真数学库。例如:

FL/FPi A. FOR

(2)/FPi87

选择协处理器。例如:

FL/FPi87 A. FOR

以上两种选项形式均是产生内部指令方式进行。

(3)/FPc

选择仿真数学库,以浮点调用形式进行。例如:

FL/FPc X. FOR

(4)/FPc87

选择协处理器,以浮点调用形式进行。例如:

FL/FPc X. FOR

(5)/FPa

选择备用数学库处理,以浮点调用形式进行。例如:

FL/FPa X. FOR

若该选项省缺,则使用协处理器进行处理,但必须认定微机上有 80287、80387、80487 等相应的芯片配置,否则应采用其它方式。常用/FPi 选项。

2. HELP 选项

使用该选项可以从屏幕上列出 FL 命令的选项清单。例如:

FL/HELP

本选项不区分大小写英文字母。

3. C 选项

C 选项是只编译不连接的控制选项,用于对多个子程序的编译。例如:

FL/C *.FOR

FL/C A.FOR

4. Zs 选项

该选项也称为语法检查选项,用于编译时只检查语法错误,不产生编译代码,从而加快对 FORTRAN 源程序文件查错的速度。例如:

FL/Zs *.FOR

5. W 选项

该选项是对警告性错误信息的控制开关,引用形式为:

/W0 禁止警告错误信息的显示。

/W1 或省缺数字 1,显示所有警告性错误信息。

/W2 禁止发送以下两个警告信息:

F4998 Variable used but not declared

F4999 Variable declared but not used

例如:

FL/W0 *.FOR

对当前目录中所有 FORTRAN 源文件编译时禁止发送警告信息。

6. /Fo 选项

此选项可以指定目标文件名字。例如:

FL/Fo xy.OBJ A.FOR

对 A.FOR 源程序文件进行编译,产生 xy.OBJ 目标文件。

如果缺省/Fo 项时,目标文件名与源文件名同名,而扩展名为.OBJ。

7. /Fe 选项

此选项用来指定可执行文件的名字。例如:

FL/Fe xy C;R.FOR

经编译、连接之后,源文件 R.FOR 形成一个 xy.EXE 可执行文件。

8. 内存模式的选项

/AL 选择大内存模式

/AM 选择中内存模式

/AH 选择巨内存模式

内存模式的选项与浮点运算处理方式选项(简称浮点选项)配合使用,决定了目标文件所连接的标准库名,见表 1-1。

表 1-1

浮点选项	内存模式选项	隐含指定的标准库名
/FPI87	/AL 或/AH	LLIBFOR7.LIB
	/AM	MLIBFOR7.LIB
/FPI 或/FPc	/AL 或/AH	LLIBFORE.LIB
	/AM	MLIBFORE.LIB
/FPa	/AL 或/AH	LLIBFORA.LIB
	/AM	MLIBFORA.LIB

1.3.3 带有/LINK 选项的 FL 命令

若不指定 FL 命令/LINK 选项,则自动与 FORTRAN 库连接。当需与其它库连接时,应用/LINK 后面的选项。命令格式为:

FL 选项 文件名表 /LINK 连接库

命令中的连接库项应指明非标准 FORTRAN 库的名字。例如:

FL/FPi xy.FOR/LINK GRAPHICS.LIB

编译 xy.FOR 源程序文件形成的 xy.OBJ 文件与 FORTRAN 5.0 图形库 GRAPHICS.LIB 连接,产生 xy.EXE 可执行文件。

1.3.4 控制编译的选项

1. /4Ys

/4Ys 选项使所有 FORTRAN 5.0 的扩充无效。而/4Ns 取消该选项功能,与 \$STRICT 元命令功能相同。

2. /4Yf

/4Yf 选项允许使用源程序自由格式,同元命令 \$FREEFORM。/4Nf 取消这种功能。

3. /4Yt

/4Yt 选项截断变量名为六个字符。/4Nt 取消这种功能,同 \$TRUNCATE 元命令。

4. /4Y6

/4Y6 选项使用 FORTRAN 66 的规则处理 DO 循环,与 \$DO 66 元命令功能相同。/4N6 取消这种功能。

5. /4Yb

/4Yb 选项表示使用调试功能,与 \$DEBUG 元命令功能同。/4Nb 表示不使用调试功能,与 \$NODEBUG 元命令功能同。

6. /4Yd

/4Yd 选项对未声明的变量不产生警告信息,同元命令 \$DECLARE 功能。/4Nd 将产生警告信息。

7. /4cc 字串

/4cc 字串选项用于对某源程序文件的条件编译,仅编译第一列为指定字串中字母的行。与元命令 \$DEBUG:字串中的作用相同。例如:

FL/c/4ccAB X1.FOR

编译 X1.FOR 源程序文件中所有以 A、B 字母打头的行。

控制编译选项组合使用例子如下:

FL/4Ybd *.FOR>LIS

此命令的作用是,编译、连接当前目录下的所有 FORTRAN 源程序文件,/4Yb 和/4Yd(组合为 4Ybd)选项均有效,全部信息送 LIS(重定向)文件中。

1.3.5 /MA 汇编选项

/MA 的使用形式为:

/MA 选项

功能为其选项之后的汇编语言文件指定 Microsoft 汇编程序选项。例如:

FL P1.FOR/MA/B40 T1.ASM

此命令作用是将选项/B40 传送到 Microsoft 汇编程序,以指定汇编 T1. ASM 源文件时,文件缓冲器存储容量为 40k。

1.4 FORTRAN 5.0 源程序的运行

对 FORTRAN 5.0 源程序编译和连接后产生的文件扩展名为 .EXE 的可执行文件在操作系统状态下均可运行。如果该文件在当前目录中,键入文件名(可以省略文件扩展名 .EXE)和回车键后,即可运行此源程序。如果该文件不在当前目录中,还需要指出该文件的路径。

1.5 FORTRAN 5.0 程序结构

一个 FORTRAN 5.0 源程序可以由一个主程序或一个主程序和若干个子程序组成(通常主程序写在最前面)。主程序和各个子程序均用 END 语句结束,而它们各自开始的第一条语句标识不同的程序单位。

1.5.1 程序单位

FORTRAN 程序单位可分为主程序和子程序。主程序单位通常称为主程序段(块),子程序单位称为子程序段(块)或辅程序段(块)。

主程序段的第一条语句可以是 PROGRAM 语句,或非 PROGRAM、FUNCTION、SUBROUTINE、BLOCK DATA 语句,以表示主程序单位由此开始。主程序是 FORTRAN 5.0 源程序运行时首先要执行的程序单位,从它的第一条可执行语句开始执行。在主程序中可以多次调用子程序,但在子程序中绝对不能调用主程序。控制整个源程序执行路线的程序单位是主程序,所以它在源程序执行的过程中起主导作用。

子程序段的第一条语句可以是 FUNCTION 语句、SUBROUTINE 语句、BLOCK DATA 语句。它们表示的子程序单位分别是函数子程序、子例行子程序、数据块子程序。其中,函数子程序和子例行子程序不仅可以被主程序调用,也可以被另外函数子程序及子例行子程序调用。数据块子程序特殊,它不能被任何程序单位调用,仅用于源程序编译阶段给有名公用区的变量赋初值。

1.5.2 程序行

一个程序单位由若干程序行构成。程序行通常按照列区的划分书写,1~5 列为语句标号区,6 列为续行区,7~72 列为语句区,73~80 列为注解区。一般地,注解区不用,为空白。各种程序行应写在指定区域内,以区分不同的程序行。

1. 注释行

第 1 列为字母 C 或“*”号的行是注释行。注释的内容写在 2~72 列内。注释行对程序运行不产生任何影响,只是为阅读程序方便而加上的注释说明。注释行可出现在程序的任何地方,并且行数可根据需要而定。

2. 初始行

第 6 列(续行区)为空白或零,1~5 列(语句标号区)可有或无语句标号,7~72 列(语句区)内书写语句内容的行是初始行。该行指一个 FORTRAN 语句仅有的一行或 FORTRAN 语句的第一行。

3. 续行

第 6 列(续行区)上书写零或空格以外的字符,1~5 列(语句标号区)为空白,7~72 列(语句区)内书写语句内容的行是续行。该行表示本行 7~72 列的语句内容是上一行的延续。一般地,一个 FORTRAN 语句可以有 19 个续行。

4. 结束行

只在 7~72 列(语句区)内书写 END 语句的行是结束行。它标识一个程序段的结束。一个程序单位只能有一个结束行,且必须是最后一行。

5. 元命令行

第 1 列上写有美元符号“\$”的行是元命令行。元命令由 FORTRAN 5.0 提供。它控制编译程序的操作。例如,使用元命令 \$FREEFORM,编译程序可处理按自由格式编写的源程序。其形式如下:

```
$FREEFORM
源 { .....
程 { :
序 { .....
```

但必须注意,编写的源程序要符合下面规定:

①注释行的第 1 列不能用字母 C 或“*”号,只能用双引号或感叹号表示注释行;

②初始行允许从任一列开始;

③语句前的语句标号与语句之间可以不用空格分开;

④1~72 列内行末的最后一个字符是“-”(负号)号,表示下一行为该行的续行,续行可从任意一列开始。

FORTRAN 5.0 元命令的使用详见第 7 章。

1.5.3 语句标号和语句

1. 语句标号

当初始行上语句要被其它语句引用时,那么该行 1~5 列(语句标号区)上应有语句标号。语句标号用 1~99999 中的无符号整数表示。语句标号作为查找语句的标志,无大小之意义。同一个程序单位的语句标号不能相同,不同程序单位的语句标号可以相同。

2. 语句

FORTRAN 语句所在的行是初始行、续行、结束行。语句是 FORTRAN 各程序单位必不可少的主要组成部分。所以,一般地讲,FORTRAN 源程序是由语句组成的。

FORTRAN 语句分为可执行语句和非执行语句两大类。可执行语句在程序运行时使计算机产生相应的操作;非执行语句在程序编译时为编译程序提供某些必要的信息。

(1)可执行语句

可执行语句按功能可归纳为以下三种类型语句。

1)赋值语句

用于把一个表达式的值赋给一个变量。赋值语句包括算术赋值语句、逻辑赋值语句、字符赋值语句和语句标号赋值语句。程序的某些运算可以通过前三个语句实现,因而这几个语句比较常用,尤其是算术赋值语句。

语 句 名 称	形 式
算术赋值	变量=算术表达式
逻辑赋值	变量=逻辑表达式
字符赋值	变量=字符表达式
语句标号赋值	ASSIGN 语句标号 TO 整型变量 (其中语句标号是可执行语句的标号)

2)输入输出语句

主要用来传输内存与外部介质之间的数据。人与计算机的信息交往要借助下面的输入输出语句。

语 句 名 称	形 式
READ	READ(输入设备号,格式语句标号)输入量表 READ(*,格式语句标号)输入量表 READ(*,*)输入量表 READ *,输入量表 READ(输入设备号,'(格式描述符)')输入量表 (这里输入设备号还可以是星号) READ(输入设备号,整型变量)输入量表 READ([UNIT=]文件号,[FMT=]格式说明符)[, { REC=记录号 END=可执行语句标号 }][,IOSTA=整型 变量(或整型数组元素)][,ERR=可执行语句标号][输入量表] (其中,格式说明符为格式语句标号或'(格式描述符)')
WRITE	WRITE(输出设备号,格式语句标号)输出量表 WRITE(*,格式语句标号)输出量表 WRITE(*,*)输出量表 WRITE(输出设备号,格式语句标号) (这里输出设备号还可以是星号) WRITE(输出设备号,'(格式描述符)')输出量表 (这里输出设备号还可以是星号) WRITE(输出设备号,整型变量)输出量表 (其中输出设备号可以是星号) WRITE([UNIT=]文件号,[FMT=]格式说明符)[,REC=记录号][,IOSTAT=整型变量(或整 型数组元素)][,ERR=可执行语句标号]输出量表 (其中,格式说明符为格式语句标号或'(格式描述符)')
PRINT	PRINT 格式语句标号,输出量表 PRINT '(格式描述符)',输出量表 PRINT *,输出量表
OPEN	OPEN([UNIT=]文件号[,IOSTAT=整型变量(或整型数组元素)][,ERR=可执行语句标号][, FILE=字符表达式 1][,STATUS=字符表达式 2][,ACCESS=字符表达式 3][,FORM=字符表 达式 4][,RECL=整型表达式][,BLANK=字符表达式 5][,MODE=字符表达式 6]) (其中,字符表达式 1 用以表示文件的名字。字符表达式 2 表示文件状态,其值可以是 OLD、NEW、 SCRATCH 或 UNKNOWN。字符表达式 3 表示文件存取方式,其值可以是 SEQUENTIAL 或 DI- RECT。字符表达式 4 表示文件格式,其值可以是 FORMATTED 或 UNFORMATTED。字符表达式 5,其值为 NULL 或 ZERO。字符表达式 6 的值为 READ 或 WRITE 或 READ WRITE
CLOSE	CLOSE([UNIT=]文件号[,IOSTAT=整型表达式][,ERR=可执行语句标号][,STATUS=字符 表达式]) (这里,字符表达式值可以是 KEEP 或 DELETE)

语 句 名 称	形 式
INQUIRE	INQUIRE({ [UNIT=]文件号 } [, IOSTAT=整型变量(或整型数组元素)] [, ERR=可执行语句标号] [, EXIST=逻辑变量(或逻辑数组元素)] [, OPENED=逻辑变量(或逻辑数组元素)] [, NUMBER=整型变量(或数组元素)] [, NAMD=逻辑变量(或数组元素)] [, NAME=字符表达式2] [, ACCESS=字符型变量1(或数组元素)] [, SEQUENTIAL=字符型变量2(或数组元素)] [, DIRECT=字符型变量3(或数组元素)] [, FORM=字符型变量4(或数组元素)] [, FORMATTED=字符型变量5(或数组元素)] [, UNFORMATTED=字符型变量6(或数组元素)] [, RECL=整型变量(或数组元素)] [, NEXTREC=整型变量(或数组元素)] [, BLANK=字符型变量7(或数组元素)]) (其中:字符变量1表示文件存取方式,其值为SEQUENTIAL或DIRECT。字符型变量2的值可以是YES或NO。字符型变量3的值可以是NO或UNKNOWN。字符型变量4表示输入输出格式,其值可以是FORMATTED或UNFORMATTED,或空格。字符型变量5的值为YES或NO。字符型变量6的值可以是YES或NO,或UNKNOWN。字符型表达式7的值为NULL,或ZERO,或空格)
BACKSPACE	BACKSPACE([UNIT=]文件号[, IOSTAT=整型表达式] [, ERR=可执行语句标号])
ENDFILE	ENDFILE 文件号 ENDFILE([UNIT=]文件名[, IOSTAT=整型表达式] [, ERR=可执行语句标号])
REWIND	REWIND 文件号 REWIND([UNIT=]文件号[, IOSTAT=整型表达式] [, ERR=可执行语句标号])
LOCKING	LOCKING([UNIT=]文件号[, ERR=可执行语句标号] [, IOSTAT=整型变量(或数组元素或结构元素)] [, LOCKMODE=字符表达式] [, REC=整型表达式] [, RECORDS=记录个数]) (其中,字符表达式值可以是NBLCK, LOCK, NBLCK, RLCK, UNLCK之一)

上述语句中的文件号也称为设备号。

3) 控制语句

控制语句用来控制 FORTRAN 语句的执行顺序及指定操作,有 GOTO 语句(包括无条件 GOTO 语句、计算 GOTO 语句和赋值 GOTO 语句)、IF 语句(包括逻辑 IF 语句、ELSE IF 语句、ELSE 语句和 END IF 语句)、CASE 语句等。具体各语句如下。

语 句 名 称	形 式
无条件 GOTO	GOTO 可执行语句标号
计算 GOTO	GOTO(S ₁ [, S ₂ ...]) [, 整型表达式] (其中, S ₁ , S ₂ , ... 表示不同可执行语句标号)
赋值 GOTO	GOTO 整型变量 [,](S ₁ [, S ₂ ...]) (其中, S ₁ , S ₂ , ... 同上语句)
逻辑 IF	IF(逻辑表达式)可执行语句
算术 IF	IF(算术表达式)S ₁ , S ₂ , S ₃ (其中 S _i (i=1, 2, 3) 表示可执行语句标号)
块 IF	IF(逻辑表达式)THEN
ELSE IF	ELSE IF(逻辑表达式)THEN
ELSE	ELSE
END IF	END IF
CASE	CASE(常量)
SELECT CASE	SELECT CASE(表达式)
END SELECT	END SELECT
DO	DO [语句标号 [,]] 循环变量 = 表达式 1, 表达式 2 [, 表达式 3]
CONTINUE	CONTINUE
DO WHILE	DO [语句标号 [,]] WHILE(逻辑表达式)