

Microsoft FORTRAN 5.0

程序员参考手册

(共三册)

许 晓 编译

- 强大的图形函数包
- 混合语言编程
- 支持源级调试(CodeView)
- 支持 OS/2 性能
- 更丰富的控制结构
- 更好的可移植性

Microsoft FORTRAN 5.0 必备资料:

《Microsoft FORTRAN 5.0 程序员参考手册》

《Microsoft FORTRAN 5.0 用户手册》

《Microsoft FORTRAN 5.0 调试工具 CodeView 使用指南》

2

中国科学院希望高级电脑技术公司

一九九一年八月

版权所有
不许翻印
违者必究

■ 北京市新闻出版局

准印证号 3314 — 90314

■ 订购单位：北京 8721信箱资料部

■ 电 话： 2562329

■ 电 传： 01—2561057

■ 电 挂： 0755

■ 地 址：海淀影剧院北侧

■ 乘 车： 320、332、302路海淀黄庄下车

■ 办公地点：公司大楼 101房间

前 言

Microsoft FORTRAN 5.0 是运行在各种微型计算机上的最新 FORTRAN 版本,它吸取了其它语言的优点,克服了一般 FORTRAN 语言数据结构种类少、语句的格式呆板、控制结构不灵活、编译器的灵活性差和不支持图形等缺点,在与已经普及的 Microsoft FORTRAN 语言兼容的基础上增进和增加了若干重要的功能,如:

- 第一次提供了 FORTRAN 图形库;库中包括了丰富的绘图函数和字体输出函数。
- 支持所有 IBMSAA 扩展,包含许多 VAX 扩展。SAA 和 VAX 上的 FORTRAN 应用程序可以方便地移植到 MS-DOS 计算机。
- 支持 OS/2 系统,包括动态连接库与多流执行。程序可以在 DOS 和 OS/2 下运行,具有很好的灵活性。
- 允许对源程序进行条件编译。
- 提供了丰富的接口,可以方便地进行混合语言编程。
- 提供了某些新的数据结构,如复合数据类型(结构与记录)。
- 提供了若干新的控制结构,如 SELECT CASE 分支语句和 DO WHILE 循环语句等。
- 实用程序 CodeView 提供 FORTRAN 源程序的源级调试。
- 此外,还增加了许多新的函数与过程,如 EPSILON 和 HUGE 等。

Microsoft FORTRAN 5.0 的参考资料包括《Microsoft FORTRAN 5.0 程序员参考手册》、《Microsoft FORTRAN 5.0 用户手册》和《Microsoft FORTRAN 5.0 调试工具 CodeView 使用指南》。

《Microsoft FORTRAN 5.0 程序员参考手册》介绍 FORTRAN 语言的基本元素、程序结构和输入/输出系统,详细讲述了每条语句的用法,说明了内部函数和附加过程;此外,还详细解释了语言的每个元命令及编译与连接选项的用法。

《Microsoft FORTRAN 5.0 高级程序设计》讨论 FORTRAN 程序设计的高级技术,包括浮点运算、内存模式、OS/2 下的编程、混合语言编程、语言之间的数据传递、记录结构和程序优化等,该书着重介绍 Microsoft FORTRAN 5.0 图形库,包括每个绘图函数、字体函数的使用方法,并以示例相辅。

《Microsoft FORTRAN 5.0 调试工具 CodeView 使用指南》介绍调试 FORTRAN 源程序的工具 CodeView 的命令和使用方法。利用 CodeView 能监视和修改 FORTRAN 的变量,在源程序行上设置断点,单步或多步跟踪源程序的执行等等。CodeView 能极大地提高 FORTRAN 的编程效率。

编者

目 录

前言

第一章 引言	1
§ 1.1 关于本手册	1
§ 1.2 文档约定	2
第一部分 语言参考手册	3
第二章 FORTRAN 的基本元素	4
§ 2.1 字符	4
§ 2.2 名字	4
§ 2.2.1 全局名和局部名	5
§ 2.2.2 未说明的名字	6
§ 2.3 数据类型	6
§ 2.3.1 整型数	7
§ 2.3.2 单精度 IEEE 实型数	8
§ 2.3.3 双精度 IEEE 实型数	9
§ 2.3.4 复型数据	9
§ 2.3.5 逻辑数据类型	9
§ 2.3.6 字符数据类型	10
§ 2.3.6.1 C 串	11
§ 2.3.6.2 字符子串	12
§ 2.4 记录	13
§ 2.5 数组	15
§ 2.6 属性	17
§ 2.6.1 ALLOCATABLE 属性	17
§ 2.6.2 ALIAS 属性	18
§ 2.6.3 C 属性	18
§ 2.6.4 EXTERN 属性	19
§ 2.6.5 FAR 属性	19
§ 2.6.6 HUGE 属性	19
§ 2.6.7 LOADDS 属性	19
§ 2.6.8 NEAR 属性	20
§ 2.6.9 PASCAL 属性	20
§ 2.6.10 REFERENCE	20
§ 2.6.11 VALUE 属性	20
§ 2.6.12 VARYING 属性	21
§ 2.7 表达式	21

§ 2. 7. 1	算术表达式	22
§ 2.7.1.1	整数的除法	23
§ 2.7.1.2	算术操作数的类型变换	23
§ 2. 7. 2	字符表达式	25
§ 2. 7. 3	关系表达式	25
§ 2. 7. 4	逻辑表达式	26
§ 2. 7. 5	数组表达式	28
§ 2. 7. 6	操作符的优先级	29
第三章	程序结构	30
§ 3. 1	程序行	30
§ 2. 2	语句标号	31
§ 3. 3	自由格式源代码	31
§ 3. 4	语句与元命令的次序	32
§ 3. 5	参量	33
§ 3. 6	程序单元	35
§ 3. 7	主程序	36
§ 3. 8	子程序	36
§ 3. 9	块数据子程序	37
§ 3. 10	函数	37
§ 3. 10. 1	外部函数	37
§ 3. 10. 2	语句函数	38
第四章	输入/输出系统	39
§ 4. 1	FORTRAN 的 I/O 系统	39
§ 4. 2	I/O 语句	39
§ 4. 2. 1	文件名(FILE=)	41
§ 4. 3. 2	设备(UNIT=)	42
§ 4. 2. 3	文件的访问方法(ACCESS=)	43
§ 4. 2. 3. 1	顺序文件访问	43
§ 4. 2. 3. 2	直接文件访问	43
§ 4. 2. 4	输入/输出缓冲区尺寸(BLOCKSIZE=)	44
§ 4. 2. 5	编辑表	44
§ 4. 2. 6	错误和文件尾的处理(IOSTAT=,ERR=,END=)	45
§ 4. 2. 7	格式指定符(FMT=)	47
§ 4. 2. 7. 1	FORMAT 语句标号	47
§ 4. 2. 7. 2	整型变量名	48
§ 4. 2. 7. 3	字符表达式	48
§ 4. 2. 7. 4	字符变量	48
§ 4. 2. 7. 5	星号(*)	48
§ 4. 2. 7. 6	字符或非字符数组名	48
§ 4. 2. 7. 7	字符数组元素	49

§ 4. 2. 8 名字表说明符(NUL=)	49
§ 4. 2. 9 文件结构(FORM=)	49
§ 4. 2. 10 输入/输出表	50
§ 4. 2. 11 文件共享(MODE=, SHARE=)	51
§ 4. 2. 12 记录数(REC=)	52
§ 4. 3 选择文件类型	52
§ 4. 4 文件位置	54
§ 4. 5 内部文件	54
§ 4. 6 回车控制	55
§ 4. 7 格式 I/O	56
§ 4. 7. 1 不可重复的编辑描述符	56
§ 4. 7. 1. 1 省字号编辑	57
§ 4. 7. 1. 2 字符编辑(H)	57
§ 4. 7. 1. 3 位置编辑: Tab, Tab Left, Tab Right(T, TL, TR)	57
§ 4. 7. 1. 4 位置编辑(X)	58
§ 4. 7. 1. 5 可选的正号编辑(SP, SS, S)	58
§ 4. 7. 1. 6 斜杠编辑	59
§ 4. 7. 1. 7 反斜杠编辑(\)	59
§ 4. 7. 1. 8 终止格式控制(;)	59
§ 4. 7. 1. 9 比例因子编辑(P)	60
§ 4. 7. 1. 10 空格的解释 (NB, BZ)	61
§ 4. 7. 2 可重复编辑描述符	61
§ 4. 7. 2. 1 整数编辑(I)	62
§ 4. 7. 2. 2 十六进制编辑(Z)	62
§ 4. 7. 2. 3 没有指数的实数编辑(F)	63
§ 4. 7. 2. 4 带指数的实数编辑(E)	64
§ 4. 7. 2. 5 对大范围值的实数编辑(G)	65
§ 4. 7. 2. 6 双精度实数编辑(D)	66
§ 4. 7. 2. 7 逻辑编辑(L)	67
§ 4. 7. 2. 8 字符编辑(A)	67
§ 4. 7. 3 格式和 I/O 表之间的交互作用	67
§ 4. 8 表 I/O	69
§ 4. 8. 1 表式输入	69
§ 4. 8. 2 表式输出	71
§ 4. 9 名字表式 I/O	72
第五章 语句	76
§ 5. 1 语句的种类	76
§ 5. 2 语句目录	78
ALLOCATE 语句	79
ASSIGN 语句(标号赋值)	80

赋值语句计算型	81
AUTOMATIC 语句	83
BACKSPACE 语句	84
BLOCK DATA 语句	84
CALL 语句	85
CASE 语句	88
CHARACTER 语句	89
CLOSE 语句	90
COMMON 语句	91
COMPLEX 语句	92
CONTINUE 语句	93
CYCLE 语句	93
DATA 语句	94
DEALLOCATE 语句	96
DIMENSION 语句	96
DO 语句	99
DO WHILE 语句	101
DOUBLE COMPLEX 语句	102
DOUBLE PRECISION 语句	103
ELSE 语句	104
ELSEIF 语句	104
END 语句	105
END DO 语句	106
END IF 语句	106
ENDFILE 语句	106
ENTRY 语句	107
EQUIVALENCE 语句	108
EXIT 语句	110
EXTERNAL 语句	111
FORMAT 语句	112
FUNCTION 语句	112
GOTO 语句(赋值型 GOTO)	114
GOTO 语句(计算型 GOTO)	115
GOTO 语句(无条件 GOTO)	116
IF 语句(算术型 IF)	116
IF 语句(逻辑型 IF)	117
IF THEN ELSE(块型 IF)	118
IMPLICIT 语句	119
INCLUDE 语句	120
INQUIRE 语句	122

INTEGER 语句	125
INTERFACE TO 语句	126
INTRINSIC 语句	127
LOCKING 语句	127
MAP ... END MAP 语句	129
NAMelist 语句	131
OPEN 语句	133
PARAMETER 语句	136
PAUSE 语句	137
PRINT 语句	138
PROGRAM 语句	138
READ 语句	139
REAL 语句	141
RECORD 语句	141
RETURN 语句	142
REWIND 语句	143
SAVE 语句	144
SELECT CASE. . . END SELECT 语句	145
语句函数	146
STOP 语句	147
STRUCTURE. . . END STRUCTURE	148
SUBROUTINE 语句	149
类型语句	150
UNION. . . END UNION 语句	150
WRITE 语句	151
第六章 内部函数和附加过程	154
§ 6. 1 常用的内部函数	154
§ 6. 1. 1 数据类型转换函数	155
§ 6. 1. 2 数据类型信息函数	157
§ 6. 1. 3 截断与舍入	158
§ 6. 1. 4 绝对值函数符号传送函数:	159
§ 6. 1. 5 求余函数	160
§ 6. 1. 6 正差函数:	161
§ 6. 1. 7 求最大值和最小值函数	161
§ 6. 1. 8 双精度积函数	162
§ 6. 1. 9 复数操作函数	163
§ 6. 1. 10 平方根函数:	164
§ 6. 1. 11 指数函数与对数函数:	165
§ 6. 1. 12 三角函数	166
§ 6. 1. 13 字符函数	167

§ 6. 1. 14	文件尾函数	169
§ 6. 1. 15	地址函数	169
§ 6. 1. 16	位操作函数	170
§ 6. 2	字典序函数列表	172
§ 6. 3	附加过程	176
§ 6. 3. 1	时间和日期过程	176
§ 6. 3. 2	运行错误的处理过程	178
§ 6. 3. 3	命令行参数过程	178
§ 6. 3. 4	随机数过程	179
第二部分 编译与连接		180
第七章 元命令		181
§ 7.1	条件编译无命令的使用	182
§ 7.2	元命令目录	185
\$DEBUG	和 \$NODEBUG	185
\$DECLARE	和 \$NODECLARE	186
\$DEFINE		186
\$DO66		187
\$ELSE		188
\$ELSEIF		188
\$ENDIF		189
\$FLOATCALL	和 \$NOFLOATCALL	189
\$FREEFORM	和 \$NOFREEFORM	190
\$IF		191
\$INCLUDE		192
\$LARGE	与 \$NOTLARGE	193
\$LINESIZE		194
\$LIST	与 \$NOLIST	194
\$MESSAGE		195
\$PACK		195
\$PAGE		196
\$PAGESIZE		196
\$STRICT	与 \$NOTSTRICT	197
\$SUBTITLE		198
\$TITLE		198
\$TRUNCATE	与 \$NOTTRUNCATE	199
第八章 FL 命令		200
§ 8. 1	FORTRAN 文件的最基本的编译连接及运行	200
§ 8. 1. 1	用 FL 来编译和连接	200
§ 8. 1. 2	使用 FL 选项	201

§ 8. 1. 3	FL 的环境变量	202
§ 8. 1. 4	说明进行下一遍编译	202
§ 8. 1. 5	终止 FL	202
§ 8. 1. 6	使用 FL 命令(示例)	202
§ 8. 1. 7	运行 FORTRAN 程序	203
§ 8. 2	用 FL 选项(/HELP)来获得帮助	204
§ 8. 3	浮点选项(/FP)	204
§ 8. 4	内存模式选项(/A, /M)	205
§ 8. 5	OS/2 库选项(/LP, /LI, /LC)	206
§ 8. 6	数据阈值选项(/GT)	208
§ 8. 7	段的命名与组织(/ND, /NM, /NT)	208
§ 8. 8	约束程序文件的建立(/F)	209
§ 8. 9	指定 FORTRAN 语言选项(/4Y/4N)	209
§ 8. 9. 1	可选择语言特性的控制(/4Ys, /4Yi, /4Yv)	210
§ 8. 9. 2	源文件语法的控制(/4Yf, /4Nf, /4Yt, /4Nt, /4Y6, /4N6)	211
§ 8. 9. 2. 1	调试的选项	212
§ 8. 9. 2. 2	声明选项	212
§ 8. 9. 3	自动变量	213
§ 8. 9. 4	设置缺省整型数长度(/4I2, /4I4)	213
§ 8. 9. 5	条件编译(4cc, /D)	213
§ 8. 10	对源文件的指定(/Tf, /Ta)	214
§ 8. 11	只编译而不连接(/C)	215
§ 8. 12	为目标文件命名(/Fo)	215
§ 8. 13	可执行文件的命名(/Fe)	216
§ 8. 14	创建列表文件(/F)	216
§ 8. 15	特殊文件名	218
§ 8. 16	行的尺寸(/S1)及页的尺寸(/Sp)	218
§ 8. 17	标题(/St)与子标题(/Ss)	219
§ 8. 18	列表文件的格式	220
§ 8. 19	包含文件的搜索(/I, /X)	225
§ 8. 20	警告及出错信息的处理	227
§ 8. 20. 1	对出错信息的理解	227
§ 8. 20. 2	警告级选项(/W)选项:	228
§ 8. 21	语法错误(/Zs)	228
§ 8. 22	为调试做准备(/Zi, /Od, /Zd0)	229
§ 8. 23	80186、80188、80286 或 80386 的使用(/G0, /G1, /G2)	230
§ 8. 24	优化(/O 和 Zp)	230
§ 8. 25	堆栈探针的启动与禁止(/Ge, /Gs)	232
§ 8. 26	自动库选择的禁止(/Z1)	233
§ 8. 27	堆栈大小的设置(/F)	234

§ 8. 28	外部名长度的限制(/H)	234
§ 8. 29	对目标文件作标号(/V)	234
§ 8. 30	与库文件的连接	235
§ 8. 31	覆盖的建立	235
§ 8. 32	只连接不编译的 FL 命令	235
§ 8. 33	指定汇编选项/MA	236
附录 A	ASCII 码表	237
附录 B	与以前的版本的差别	239
附录 C	Microsoft FORTRAN 5.0 的局限	256
附录 D	错误信息	260

第一章 引言

Microsoft FORTRAN 5.0 在已经普及的 Microsoft FORTRAN 4.0 语言的基础上进行了许多新的改进,增加了一些重要的新特征。5.0 版与系统应用程序结构(SAA)FORTRAN 和 VAX 扩展完全兼容,同时还提供了某些新的程序结构,如复合数据类型(结构)和 SELECT CASE 分支语句等。5.0 的编译指令提供了许多高级特征,允许对程序代码进行条件编译。FL 编译与连接指令提供了若干个新的选项,并对已有选择项进行了改进。

Microsoft FORTRAN 的新版本支持增强型 OS/2 系统,包括动态连接库和多执行流。程序可以在 DOS、OS/2 或两个操作系统下运行,具有很好的灵活性。另外,Microsoft FORTRAN

5.0 第一次有了自己的图形库,利用它可以设计出丰富多彩的 FORTRAN 图形应用程序。本章介绍了《Microsoft FORTRAN 5.0 程序员参考手册》和书中所用的文档约定等信息。

关于内存模式的讨论、FORTRAN 程序调用非 FORTRAN 子程序和函数的方法、OS/2 下的程序设计,以及图形库的用法,请参见《Microsoft FORTRAN 5.0 用户手册》一书。关于窗口型 CodeView 调试器的用法,请参阅《Microsoft FORTRAN 源程序调试工具 CodeView 使用指南》。

§ 1.1 关于本手册

《Microsoft FORTRAN 5.0 程序员参考手册》介绍了由 Microsoft FORTRAN 优化编译程序 5.0 版实现的 FORTRAN 语言,本手册是 FORTRAN 5.0 程序的必备文档。

文档中的术语“OS/2”是指 OS/2 系统——Microsoft OS/2X 和 IBM OS/2;术语“DOS”既是指 MS——DOS 也是指 IBM PC DOS。

Microsoft FORTRAN 符合 ANSI X3.9-1987 标准的程序语言 FORTRAN77。

注:Microsoft FORTRAN 语言包含了 ANST 标准语言全集的扩展。

第二章讨论了 FORTRAN 语言的基本元素;第二章描述了 FORTRAN 程序的结构;第四章给出 FORTRAN 输入/输出(I/O)系统的详细信息;第五章详细描述了所有 FORTRAN 语句;第六章解释了所有 FORTRAN 内部函数;第七章包括了 Microsoft FORTRAN 的元命令;第八章讨论了 FL 命令。下表列出了各个专题的相应章节。

题目

章节

字符、名字、数据类型、属性、表达式
源程序中的行格式、子程序、函数与
参量;FORTRAN 程序结构
FORTRAN 中的输入和输出
FORTRAN 语句列表
FORTRAN 内部函数列表
编译指令与元命令列表
FL 命令

第二章 FORTRAN 的元素
第三章 程序结构
第四章 输入/输出系统
第五章 语句
第六章 内部函数
第七章 元命令
第八章 FL 命令

§ 1.2 文档约定

本手册中使用下列符号约定(在大多数情况下,空格在 FORTRAN 中是不重要的)。

约定的例子

约定的说明

OUT, TXT, COPY,
LINK 和 FL
[[任选项]]
{选择1选择2}

大写字母表示文件名和DOS命令,也可用在命令行选择项中(除非应用程序仅接受小写字母)。

双重方括号内的项是任选的。

花括号和竖杆可表示在二个或二个以上的项之间作选择。

必须选择这些项之一,除非所有项同时都包含在双重方括号之内。

重复元素...

跟在一项后的三个点表示可以输入具有相同形式的更多项。

三个竖点表示例子中的部分内容被有意省略了。

示例:

下例说明了定义 EXTERNAL 语句语法行中的符号约定:

```
EXTERNAL name[[[attrs]]][,name[[[attrs]]]] ...
```

此语法行说明:当使用 EXTERNAL 语句时,必须首先在 EXTERNAL 后面输入要指定的 name,然后可任选地输入一个左括号(),接着输入指定的属性(attrs),然后再输入一个右括号();如果想要指定更多的名字,任选地紧跟着输入属性(attrs),则必须输入一个逗号,接下来跟一个名字,任选地跟一个左括号,后接属性,再接一个右括号。由于 [,name, [[attrs]] 之后跟三点(); 所以可以输入多个项(一个逗号,跟一个名字,可选地在括号中紧跟一些属性)。

第一部分 语言参考手册

本部分介绍 Microsoft FORTRAN 语言 5.0 版, 包括其 ANSI 标准特征和特殊的 Microsoft 扩展。前四章介绍 Microsoft FORTRAN 语言, 其中第二章讲论 FORTRAN 语言的基本元素, 第三章概述 FORTRAN 程序结构, 第四章讨论 FORTRAN 输入/输出系统。其余的两章提供了应用这些特征所需的信息。第五章按字母顺序列出了每条语句的参考信息, 包括语法与用法; 第六章介绍了有关内部函数和附加过程的信息。

本部分包含以下各章:

第二章 FORTRAN 的基本元素

第三章 程序结构

第四章 输入/输出系统

第五章 语句

第六章 内部函数和附加过程

第二章 FORTRAN 的基本元素

本章集中介绍了 FORTRAN 程序的基本元素：特殊字符、FORTRAN 标识符的命名与作用域、数据类型及其使用规则。本章还讲述了基本算术与逻辑函数。

§ 2.1 字符

FORTRAN 源文件可包含 ASCII 字符集中任何可打印的字符。当字符常量和字符变量进行逻辑比较时，FORTRAN 字符集的排序序列是 ASCII 序列。附录 A 中列出的 ASCII 字符集包括如下内容：

- 52 个大小写字母字符(A~Z 和 a~z)

除了字符常量和 Hollerith 域外，Microsoft FORTRAN 编译器将上下文中的所有小写字母解释成大写字母。但在字符常量和 Hollerith 域中，大小写是有区别的。例如，语句 WRITE(*, *) 和 write(*, *) 是相同的，而字符常量 'ijk' 和 'Ijk' 却不同。

在字符常量中，对大小写的敏感性有个例外。第五章中列出的作为 FORTRAN 语句一部份的字符常量是不分大小写的，除非指定了 \$ STRICT 元命令，请参阅第七章“元命令”。

例如，在使用 CLOSE 语句时，可以输入一个字符常量来说明是保留还是删除文件。其语法为 [[,STATUS=status]], status 可接受的值为“KEEP”和“DELETE”。只要未设置 \$ STRICT 元命令，则将 STATUS 置为“KEEP”等价于将 STATUS 置为“Keep”或“KeEp”。

- 10 个数字(从 0 到 9)，数字可以用在用户定义的名字中，但不能做为首字符。
- ASCII 字符集中的其他可打印字符、空格和 TAB 字符。在 Microsoft FORTRAN 中，美元符号(\$)和下划线(_)可以用在用户定义的名字中，但下划线不能是第一个字符。

Microsoft FORTRAN 源程序中的空格一般没有什么意义，但是，使用空格可使程序易于阅读。例外的情况如下：

- 在字符常量或 Hollerith 域中，空格是有意义的；
- 第六列的空格或 0 表示初始行参见 3.1 节。

制表字符的解释取决于制表字符所在的列：

列	说明
---	----

1~5	源程序行中跟在制表字符后的字符被解释为第七列中的字符。
6~72	除在字符和 Hollerith 常量(见 4.7.1.2 节)内之外，制表字符解释为单空格。字符或 Hollerith 常量中的制表字符仍解释为制表字符。

§ 2.2 名字

所有的变量、数组、函数、程序和子程序都用名字来标识。名字由字母数字字符组成，而且必须遵循下列规则：

- 名字的第一字符必须是字母；其余字符必须是字母或数字。Microsoft FORTRAN 允许将符号\$用作名字的 IMPLICIT 排序序列中 Z 之后的字母字符。
- 忽略空格。变量名 low volt ge 和 lowvoltage 对编译器来说是相同的。
- 名字可以长达 31 个字符；不是所有的字符都是重要的。仅仅前六个字母数字字符是重要的，其余字符均被忽略。(此限制不适用于 Microsoft FORTRAN，除非使用了\$STRIC 或\$STRUNCATE 宏命令。)空格字符 不计算在内。例如，在指定了\$ TRUNCATE 或\$ STRICT 时，名字 delicate 和 deli cat e 都被解释为 delica。
- 操作系统或连接程序可能对名字长度会有其他限制。与其它语言不同，FORTRAN 的关键字不是保留字。编辑程序通过上下文识别关键字，例如，程序可有名为 IF, READ 或 GOTO 的数组，但这些名字会使程序较难阅读和理解。为了提高可读性和方便查错，程序员应该避免使用与 FORTRAN 语句相似的名字。

当一个记号错误使得编译程序把一个 FORTRAN 关键字解释为一个变量名部分时将产生另一类错误。考虑下列二条语句：

```
DO 5 INC =1,20
```

```
DO 5 INC = 1. 20
```

第一条语句是一个 DO 循环的开始；第二条语句把 1. 20 赋给名为 DO5INC 的变量。两条语句之间唯一的不同之处是第一条包含一个逗号而第二条包含一个句号。编译程序不能发现这类错误。DO 循环的终止行(标号为 5 的行)与程序中的其他行没有什么不同，因而编译程序无法得知一条 DO 语句被丢失。

由于编译程序保留下列三个预定义名字，故它们不能用作程序中的名字：

1. _main，这是主程序的外部名字。(在一定的条件下允许使用“main”，但建议不要使用。详细信息请见 3. 7 节)。
2. COMMQQ，这是空的公共块名。
3. BLKDQQ，这是块数据子程序的缺省名。

此外，以二个下划线字符()开头或以 QQ 终止的名字(_main 或 MAINQQ)均为编译程序保留。如果你需要使用以二个下划线字符开头或以 QQ 结尾的名字，请使用 ALLAS 属性，以给变量提供一个可接受的外部名。

§ 2. 2. 1 全局名和局部名

名字类型：全局名和局部名。

全局名在给定程序的任何地方都可识别，所以它们仅需在程序的任何地方作一次全局定义。所有的子程序、函数、公共块和程序名都是全局的。例如，若程序中使用了名为 Sort 的子程序，则不能在该程序中将一个函数命名为 Sort。

但是，只要不在不同的程序单元内引用全局名 Sort 名字 sort，就可以将名字 Sort 用作一个局部名。例如，只要说明变量 Sort 的子程序不调用函数 Sort，含有函数 Sort 的程序就可以包含该子程序。

公共块名是全局名的一个特例，它也可用作同一程序的一个局部名，因为公共块名总是包含在斜杠之间，因而可与其它名字区分开。例如，如果程序含有名为 /distance / 的公共块，该程序中的一个数组也可以命名为 distance(数组具有局部名)。

局部名仅在一个程序单元里有意义。在同一程序的另一个程序单元中，相同的名字可

再次使用，以引用同一程序对象，或不同对象。

所有的变量、数组、参量和语句函数都有局部名。

语句函数的参量是局部名的一个特例。这些参量仅在语句函数的语句中有意义。但是，如果在语句函数的语句之外使用参量名，那么子程序中的局部变量必须与语句函数的同名参量有相同的数据类型。详细信息请参阅 5.2 节内容。

§ 2.2.2 未说明的名字

如果一个名字没有明确地定义，那么编译程序将根据上下文对该名字进行分类。如果 IMPLICIT NONE 已经说明过，当任何没有明确类型的变量被首次使用时，就产生一个出错信息。如果已经说明了 \$DECLARE 元命令，那么在首次使用 DATA、PARAMETER 或赋值语句中未赋值的任何变量时，就会产生警告信息。下表解释未说明的名字的分类方法：

名字	分类
作为一个变量或出现在函数引用中	变量或函数返回值的类型由名字的第一个字母确定。缺省时，以字母 I、J、K、L、M 或 N (大写或小写) 开头的变量名为 INTEGER (整整) 型。以任何其它字母或美元符号开头的变量名为 REAL (实数) 型。IMPLICIT 语句可改变第一个字母字符 (包括美元符号) 与类型之间的对应关系。
作为 CALL 语句的目标	编译程序假定该名字为子程序名。子程序并不通过其名字来返回值，所以不存在类型问题。

§ 2.3 数据类型

本节描述了 FORTRAN 的六个基本数据类型。它们是：

- 1. 整型 (INTEGER, INTEGER * 1, INTEGER * 2 和 INTEGER * 4);
- 2. 实型 (REAL, DOUBLE PRECISION, REAL * 4 和 REAL * 8);
- 3. 复数型 (COMPLEX, COMPLEX * 8, DOUBLE COMPLEX 和 COMPLEX * 16);
- 4. 逻辑型 (LOGICAL, LOGICAL * 1, LOGICAL * 2 和 LOGICAL * 4);
- 5. 字符型 (CHARACTER [* n], 此外 $1 \leq n \leq 32,767$);
- 6. 记录型 (STRUCTURE 类型定义的变量)。

变量、数组、符号常量或函数等的数据类型可在说明语句中说明。若未说明数据的类型，编译程序就根据名字的第一个字母来确定其数据类型 (正如 2.2.2 节所述)。类型语句也可以说明数组变量的维数信息，还可用于变量和数组的初始化 (在 Microsoft FORTRAN 中)。类型语句的详细描述请见第五章“语句”。

以下各小节分别描述各个数据类型。内存要求见表 2.1。

表 2.1 内存要求

类型	字节	注释
----	----	----