

國防工業出版社

FELIX C-256 计算机 FORTRAN语言

薛学勤 李淑扬 李隆江 编

国防工业出版社

内 容 简 介

本书共分两部分。第一部分是 FORTRAN IV 语言, 在这部分中, 比较详尽地介绍了标准 FORTRAN 语言及其某些扩充的功能, 并配有较多的例题, 便于自学。第二部分介绍 FELIX C-256 计算机的作业控制语言 (JCL) 的使用, 磁盘、磁带文件的使用, 程序的分节与覆盖, 程序库的应用等, 并给出了各种典型作业的组织方式, 供使用者参考。

本书供科技工作者, 程序人员在 FELIX C-256 机上算题时使用。也可作为高等学校学生学习 FORTRAN 语言的参考书。

FELIX C-256 计算机 FORTRAN 语言

薛学勤 李淑扬 李隆江 编

国防工业出版社出版

新华书店北京发行所发行 各地新华书店经售

机械工业出版社印刷厂印装

787×1092¹/₁₆ 印张14¹/₄ 328千字

1980年7月第一版 1980年7月第一次印刷 印数: 00,001—11,900册

统一书号: 15034·1922 定价: 1.50元

前 言

为了满足广大读者学习 FORTRAN 语言的需要, 并且为了供某些读者在 FELIX C-256 及同类计算机上进行程序设计时参考, 我们在推广 FORTRAN 语言讲稿的基础上, 并参照我们在北京航空学院计算中心进行程序设计和用户咨询工作中的经验编写了这本书。

本书分为两部分。第一部分讲述 FORTRAN 语言, 内容包括国际标准 FORTRAN (*Draft ISO recommendation ISO/TC 97/SC 5*, 1965 年 10 月) 和 FELIX 机某些扩充的功能。为便于读者理解, 对每一个语句都做了较详细的解释, 并有数量较多的例题。其中大部分例题都给出了相应的框图。第二部分介绍 FELIX 计算机的作业控制语言 (简称 JCL) 的应用, 内容包括主要控制卡片的格式和用法, 磁盘和磁带文件的应用, 程序库, 程序的分节和覆盖, 作业组织举例及计算机的概况介绍。想学习 FORTRAN 语言而并不准备在 FELIX 机上做题的读者可以只读第一部分, 第二部分主要是为在此类计算机 (包括 FELIX C-512, IRIS 60 等) 上做题的读者编写的, 也可供那些想了解操作系统功能的读者参考。附录 (FELIX C-256 概况介绍) 供阅读各章时查阅, 学习重点放在第二部分的读者也可以先粗读一下附录再去读其他各章。

本书第一章至第六章由薛学勤编写, 第七、八、十二章由李淑扬编写, 第九、十、十一、十三章和附录由李隆江编写。陈望梅同志审校了部分初稿。由于我们水平有限, 时间仓促, 所以本书一定有不少缺点和错误, 敬请读者批评指正。

目 录

第一部分 FORTRAN IV 语言	1	4.1 计算转移语句	38
第一章 FORTRAN 语言概述	1	4.2 赋值转移语句	40
§ 1 FORTRAN 程序举例	1	第五章 数组、循环	42
§ 2 FORTRAN 程序的结构	4	§ 1 数组与维数语句	42
§ 3 FORTRAN 程序的书写格式	6	1.1 数组	42
3.1 FORTRAN 的字符集	6	1.2 维数说明语句	43
3.2 FORTRAN 程序的书写规则	7	1.3 数组元素的存贮顺序	45
§ 4 语句标号、标识符和标准函数	10	1.4 程序举例	46
4.1 语句标号	10	§ 2 循环	47
4.2 标识符	11	2.1 循环语句 (DO 语句)	48
4.3 标准函数	11	2.2 继续语句 (CONTINUE 语句)	50
第二章 常数、变量与类型说明语句	13	2.3 关于循环的嵌套和循环内的转移	51
§ 1 常数	13	§ 3 程序举例	53
1.1 整型常数	13	第六章 输入输出语句	57
1.2 实型常数	14	§ 1 输入输出概述	57
1.3 双精度型常数	15	1.1 文件上的输入输出	57
1.4 复数型常数	15	1.2 文件的 FORTRAN 号	58
1.5 逻辑型常数	16	1.3 输入输出表	59
1.6 文字型常数	16	§ 2 带格式的读写语句	61
§ 2 变量及类型说明语句	17	2.1 带格式的读语句	61
2.1 隐含规则	17	2.2 带格式的写语句	61
2.2 类型说明语句	17	§ 3 格式语句	62
§ 3 隐含说明语句	18	3.1 格式语句的一般形式	62
第三章 赋值语句	20	3.2 各类区描述符	62
§ 1 算术表达式与算术赋值语句	20	3.3 I 型描述符	63
1.1 算术表达式	20	3.4 F 型描述符	65
1.2 算术赋值语句	23	3.5 E 型描述符	69
§ 2 逻辑表达式与逻辑赋值语句	25	3.6 G 型描述符	70
2.1 关系表达式	25	3.7 D 型描述符	71
2.2 逻辑表达式	26	3.8 重复因子 r 与分隔符	71
2.3 逻辑赋值语句	27	3.9 比例因子 mP	73
§ 3 标号赋值语句	28	3.10 L 型描述符	76
第四章 控制语句	29	3.11 X 型描述符	76
§ 1 无条件转移语句 (GOTO 语句)	29	3.12 H 型描述符	77
§ 2 停语句和暂停语句	30	3.13 A 型描述符	79
2.1 停语句 (STOP 语句)	30	3.14 Z 型描述符	81
2.2 暂停语句 (PAUSE 语句)	31	3.15 T 型描述符	83
§ 3 条件语句	32	3.16 打印机纵向走纸控制	84
3.1 逻辑条件语句 (逻辑 IF 语句)	32	3.17 格式语句小结	85
3.2 算术条件语句 (算术 IF 语句)	35	3.18 格式说明与输入输出表	86
§ 4 计算转移语句和赋值转移语句	38	3.19 数组中的格式说明	87
		§ 4 无格式读、写语句	88
		4.1 无格式的读语句	89

4.2 无格式的写语句	89	§ 9 函数和子程序小结	127
§ 5 输入输出的辅助语句	89	第八章 公共语句、等价语句、数据初值语 句和数据块子程序	129
5.1 重绕语句	89	§ 1 公共语句	129
5.2 回退语句	89	§ 2 等价语句	132
5.3 文件结束语句	90	§ 3 数据初值语句	136
§ 6 带文件结束和阅读错误选择的读语句	90	§ 4 数据块子程序	139
§ 7 名字表输入输出语句	91	第二部分 作业控制语言(JCL)的应用	141
7.1 名字表语句	91	第九章 编链控制卡片	141
7.2 名字表读语句	92	§ 1 控制卡片	141
7.3 名字表写语句	97	1.1 控制卡片的一般格式	143
§ 8 直接存取文件的读、写语句	98	1.2 控制卡片格式描述中的约定	144
8.1 直接读语句	98	§ 2 编链控制卡片	145
8.2 直接写语句	99	2.1 JOB卡片	146
8.3 寻找语句	99	2.2 编译卡片	146
§ 9 与 FORTRAN I 兼容的输入输出 语句	100	2.3 连接编辑卡片	148
9.1 印刷语句	100	2.4 运行卡片	148
9.2 读语句	100	2.5 SYSRUN 卡片	149
9.3 穿孔语句	101	2.6 EOJ 卡片	150
§ 10 内部读、写语句	101	第十章 磁盘和磁带文件	151
10.1 内部写语句	101	§ 1 有关文件的一些概念	151
10.2 内部读语句	103	1.1 文件	151
第七章 函数和子程序	104	1.2 文件的介质	151
§ 1 概述	104	1.3 文件的结构	152
§ 2 语句函数	104	1.4 文件的类型	153
2.1 语句函数的应用	104	§ 2 文件的定义	153
2.2 语句函数的定义语句	105	§ 3 有关文件的控制卡片	163
2.3 语句函数的调用方法	107	3.1 外设分配卡片 ASSIGN	163
2.4 语句函数的程序举例	107	3.2 文件标识卡片 LABEL	164
§ 3 内部函数和常用外部函数	108	3.3 预标卡片 INIT	164
3.1 内部函数表和常用外部函数表	108	3.4 磁盘区租用卡片 ALLOC	165
3.2 调用方法	110	§ 4 文件的操作	166
§ 4 返回语句、调用语句和外部语句	111	4.1 系统文件的操作	166
4.1 返回语句	111	4.2 顺序使用文件的操作	167
4.2 调用语句	111	4.3 直接存取文件的操作	169
4.3 外部语句	111	4.4 通过 BUFFER IN (OUT) 使用的磁带文件的 操作	170
§ 5 函数子程序	112	§ 5 使用磁盘磁带文件的例子	170
5.1 函数子程序的定义语句	112	第十一章 程序库	173
5.2 函数子程序的调用方法	113	§ 1 程序库的类型、管理和介质	173
§ 6 子例程子程序	118	§ 2 程序库的创立	173
6.1 子例程子程序定义语句	118	§ 3 程序的入库	174
6.2 子例程子程序的调用方法	119	3.1 源程序入库	174
6.3 子例程子程序应用举例	121	3.2 BT 程序入库	176
§ 7 专用子程序	122	3.3 IMT 程序入库	176
7.1 SSWTCH 开关子程序	122	§ 4 从程序库中调用程序	177
7.2 SLITE 和 SLITET 子程序	124	4.1 源程序	177
§ 8 入口语句 ENTRY	126		

4.2 BT 程序	179	§ 4 对入库的源程序进行编译的作业	194
4.3 IMT 程序	180	§ 5 从 BT 程序库中调用程序的作业	195
§ 5 带、盘之间程序库的复制	180	§ 6 从 IMT 程序库中调用程序的作业	195
§ 6 数学库的用法	180	§ 7 编译、连接编辑和运行的作业	196
第十二章 程序的分节与覆盖	182	§ 8 程序分节的作业	197
§ 1 分节与覆盖的意义	182	§ 9 使用磁带文件的作业	198
§ 2 一般程序的结构	182	附录 FELIX C-256机概况介绍	199
§ 3 节与分节的必要性	183	§ 1 硬件	199
3.1 模块	183	1.1 主机	199
3.2 节	183	1.2 外围设备	207
§ 4 程序分节的控制卡片和命令卡片	184	§ 2 软件	208
4.1 SEG 控制卡片	184	2.1 软件系统的结构	208
4.2 SEGMENT 命令卡片	185	2.2 多道程序	212
4.3 SEGDATA 命令卡片	186	2.3 作业的批处理	213
4.4 TREE 控制卡片 (树形卡片)	187	2.4 远程处理	215
§ 5 分节程序举例	188	附表一 FORTRAN 语句表 (一)	215
第十三章 作业举例	193	附表二 FORTRAN 语句表 (二)	217
§ 1 只编译作业	193	附表三 EBCDIC 代码和卡片孔型表	218
§ 2 编译并把 BT 程序入库的作业	193	附表四 各种常用控制卡片的格式	218
§ 3 编译、连接编辑并把 IMT 程序入库的作业	194	附表五 十进制数-十六进制数的转换	220

第一部分

FORTRAN IV语言

第一章 FORTRAN语言概述

FORTRAN 语言是一种适用于科技计算的程序设计语言。FORTRAN 一名是由 英语 FORMula TRANslator (公式翻译) 两个单词的第一个音节拼写而成的。这种语言规则简单, 使用方便, 容易掌握。目前, 国内外科技工作者不仅广泛采用 FORTRAN 语言来编写程序, 而且也用它作为交流技术资料的手段之一。

用 FORTRAN 语言编写的程序称为 FORTRAN 源程序。将源程序输入后, 经过 FORTRAN 编译程序翻译成为由机器指令表示的程序, 这种程序称为目的程序。

本章主要介绍 FORTRAN 程序的概貌, 书写格式以及某些基本概念。

§1 FORTRAN程序举例

我们从阅读一个简单的 FORTRAN 程序入手, 开始 FORTRAN 程序语言的学习。对于本节所举的程序例子, 只要求读者通过阅读对 FORTRAN 程序有一个全面的感性认识, 以利于后续各章的学习。程序中所出现的各种 FORTRAN 语句的作用及其使用规则将在以后各章中分别详细介绍, 这里可暂不深究。

例 测得某三角形 $\triangle ABC$ 的三边长分别为 $a = 45.7$ 毫米, $b = 79.4$ 毫米, $c = 105.3$ 毫米, 试编写一个计算 $\triangle ABC$ 面积的 FORTRAN 程序。

由初等数学可知, 已知三角形三边的长, 求其面积 A 的公式是

$$A = \sqrt{s(s-a)(s-b)(s-c)}$$

其中 a, b, c 分别为三角形的三边长,

$$s = \frac{a+b+c}{2}$$

本例的 FORTRAN 程序如下:

```
C      EXAMPLE
C
      READ (105, 999) A, B, C
999    FORMAT (3F6.3)
      S = (A+B+C) / 2.0
      AREA = SQRT (S*(S-A)*(S-B)*(S-C))
      WRITE (108, 124) A, B, C, AREA
124    FORMAT (1H , 2HA=, F6.3, 5X, 2HB=, F6.3,
&      5X, 2HC=, F7.3, 5X, 5HAREA=, F9.3)
      STOP
      END
```


这个 FORTRAN 源程序共有十一行。现将每一行的作用分别说明如下:

第 1 行和第 2 行都是注释行。它对程序不起任何作用，是可有可无的。这里的第一个注释行只是注上“例子”一词，而第二个注释行没有任何注释内容，是一个空白行，其作用是使注释行与源程序的语句行之间有一个明显的分隔，便于阅读。

第3行 READ (105, 999) A, B, C 是输入语句。它指出在文件的 FORTRAN 号为 105 的作业文件上 (这里可暂时理解为设备号为 105 的卡片阅读机上), 按语句标号为 999 的格式语句所规定的格式控制, 读入三个数据并赋给名字为 A, B, C 的三个变量。

第4行 999 FORMAT (3 F6.3) 是格式语句。这里，它是控制输入格式用的。这语句具体规定了在一张卡片上穿三个数据，每一个数据占卡片上六列的长度，其中有三位小数，这就是格式语句括号中 3 F6.3 的具体含意。按这格式将 a 、 b 、 c 三个已知数据穿在卡片上，具体如图 1-1 所示。

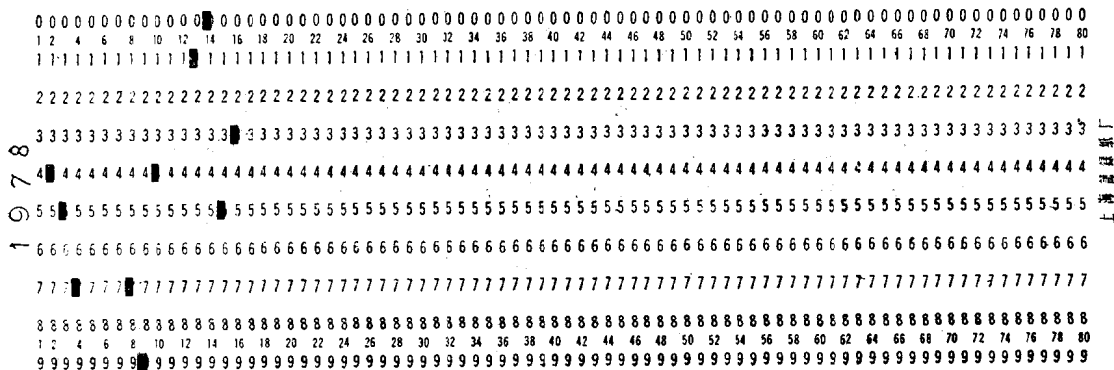


图1-1 数据卡片

第5行 $S = (A + B + C) / 2.0$ 是赋值语句。它计算表达式 $(A + B + C) / 2.0$ 的值并将计算结果赋给名字为 S 的变量。

第6行 AREA=SQRT(S*(S-A)*(S-B)*(S-C)) 也是赋值语句。它计算表达式 $\text{SQRT}(S*(S-A)*(S-B)*(S-C))$ 的值并将计算结果赋给名字为 AREA 的变量。显然, AREA 中的值就是所要计算的三角形 $\triangle ABC$ 的面积。语句中出现的 SQRT(X) 是 FORTRAN 语言中备有的标准函数, 其作用是求 X 的平方根。

第7行 WRITE (108, 124) A, B, C, AREA 是输出语句。它指出在文件的 FORTRAN 号为 108 的输出文件上 (这里可暂时理解为设备号为 108 的宽行打印机上), 按语句标号为 124 的格式语句所规定的格式控制, 输出名字为 A、B、C、AREA 四个变量的值。也就是印出初始数据和计算结果。

第 8 行和第 9 行 这两行表示了一个完整的格式语句 124 FORMAT (1H␣, 2HA=, F6.3, 5X, 2HB=, F6.3, 5X, 2HC=, F7.3, 5X, 5HAREA=, F9.3)。其中, 第 9 行是第 8 行的继续^②。这语句用来控制输出格式, 括号中的每个格式描述符的含意由图 1-2 示意说明, 其中第一个格式描述符 1H␣是控制打印机纵向走纸用的, 图中没有表示出来。

● 本例的格式语句124如果按照FORTRAN语句的书写规则可以一行写下的,但由于印刷版面的限制,这里利用继续行的办法来表示。以后各章中,凡遇到类似情况,均按这种方式处理。

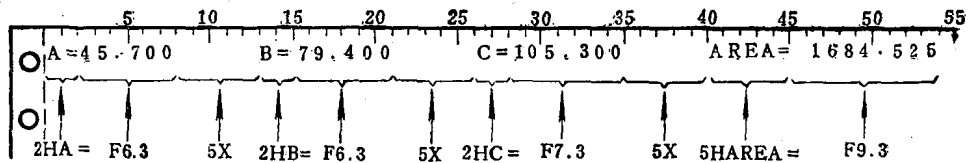


图1-2 输出的结果

这里使用符号“□”表示一个空格字符。在以后各章中，为了明确起见，凡是 FORTRAN 语句或字符串中必须指明有空格存在时，都以符号“□”表示之。

第 10 行 STOP 是停语句。它使程序停止执行并将控制转给操作系统的中央监控程序。

第 11 行 END 是结束行。它标志一个 FORTRAN 程序段的结束。

本例题的 FORTRAN 程序也可以用子程序段的形式来编写。整个程序可以由一段已知三边长计算三角形面积的子程序段和调用这子程序段的主程序段来组成。为了对 FORTRAN 程序有一个比较全面的感性认识，这里对这种形式的程序也作一简单介绍。本例题的程序具体如下：

主程序段

```
C      EXAMPLE
C
      CALL TP (45.7, 79.4, 105.3, SQ)
      WRITE (108, 25) SQ
25  FORMAT (1H , 3HSQ=, F9.3)
      STOP
      END
```

子程序段

```
SUBROUTINE TP (A, B, C, AREA)
  S = (A + B + C) / 2.0
  AREA = SQRT (S * (S - A) * (S - B) * (S - C))
  RETURN
END
```

子程序段中，第一个语句

```
SUBROUTINE TP (A, B, C, AREA)
```

叫子程序定义语句。每个子程序段的第一个语句必须是子程序定义语句，其作用是定义子程序的具体名称和确定形式参数的个数、类型以及排列次序。就本例来说，它定义子程序的名称是 TP，圆括号中的四个变量名 A、B、C、AREA 是形式参数，分别形式上表示三角形的三边长和其面积。子程序 TP 中，这四个变量没有具体确定的数值，只是在形式上描述了这四个量之间的运算关系。当其它程序段调用这子程序时，分别用有确定数值的参数去代替形式参数。这确定的参数称实在参数。只要改变实在参数的值，就可以计算出对应的结果。这样，一个子程序段可以多次调用，从而使程序编写的工作变得更为灵活。语句 RETURN 称返回语句。其作用是使子程序在执行结束后，返回到调用程序段中去。

一个子程序段必须要有其它的程序段来调用它。调用程序段可以是主程序段或其它子程序段。本例的调用程序段是一主程序段。这里，语句

```
CALL TP (45.7, 79.4, 105.3, SQ)
```

就是调用语句。其作用是调用名字为 TP 的子程序，括号中所指出的常数和变量名 SQ 就是与子程序段 TP 的形式参数在个数、类型及次序上都一一对应的实在参数。即 $A = 45.7$, $B = 79.4$, $C = 105.3$, 计算得的结果存入名为 SQ 的变量之中。当主程序段执行这语句后，就转去执行子程序段 TP。执行子程序 TP 后，计算出具体三角形面积并返回到主程序段调用语句之后的输出语句 `WRITE (108, 25) SQ`。程序中其它语句的作用前面已有说明，这里不再赘述。本程序执行时，具体路线如图 1-3 所示。

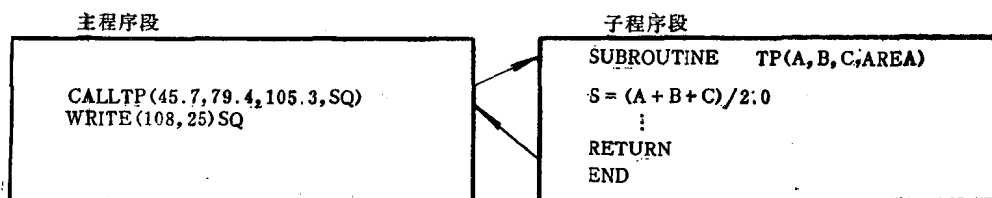


图 1-3

由上述例子可以看到，FORTRAN 语言是一种分段结构的、比较直观的程序设计语言。

§ 2 FORTRAN 程序的结构

一个 FORTRAN 程序是由一个或几个程序段所构成。分段结构是 FORTRAN 语言的主要特点。程序段分主程序段、子例程子程序段、函数子程序段和数据块子程序段四种。

程序段就是一个以结束行 END 终止的 FORTRAN 语句序列。凡是以子程序定义语句开头的程序段称为子程序段。除子程序定义语句以外的其它 FORTRAN 语句开头的程序段称为主程序段。一个 FORTRAN 程序只能有一个主程序段，可以有一个或几个子程序段，也可以没有子程序段。程序总是从主程序段开始执行的。如上节的例子中，前一种形式表示的程序就只有一个主程序段，而后一种形式表示的程序是由一个主程序段和一个子程序段所构成。

组成一个 FORTRAN 程序的各个程序段之间是相互独立的。也就是说，每个程序段是一个独立的编译单位，程序段中所使用的语句标号和给变量、数组命名用的标识符只有在本程序段内有意义。所以，虽然在不同程序段中使用了相同的标识符，但它所标识的量一般来说是不相同的。

程序分段结构的优点是：

(1) 可以使程序的编写和调试工作平行进行。例如，对于一个大型的程序，可以按问题的具体内容或处理方法分成若干个相对独立的部分，分别由几个程序人员同时分头编写与调试，最后，进行联合调试。由于各程序段是相互独立的，除了对程序段之间参数传递的方式要作必要的协调外，其余在程序段中所用到的变量标识符、语句标号等都可以由程序人员自己确定，互相不受牵制。这样，可以加快程序的编制和调试的进度。

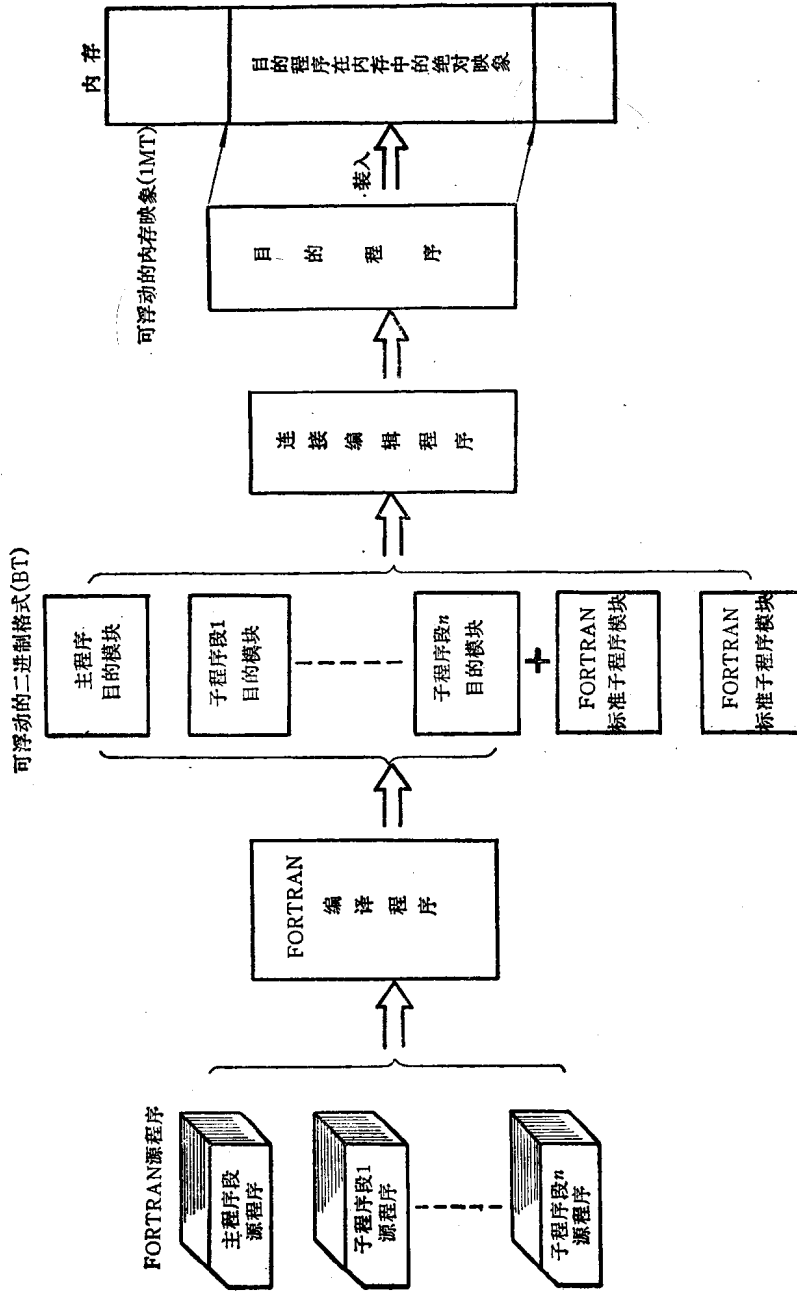


图1-4 源程序编译过程示意图

(2) 对于一些常用的计算方法或专业性的数据处理方法都可以预先编成子程序段, 保存在程序库中供用户选用, 从而可以避免大量的重复性的工作, 节省人力物力, 提高工作效率。

那么, 一个具有若干个互相独立的程序段所组成的 FORTRAN 源程序是怎样编译成一个完整的可执行的目的程序的呢? 对这个问题, 下面作一概括性的说明。在 FELIX C-256 机上, 一个 FORTRAN 源程序分两步翻译成一个完整的可执行的目的程序。首先, 由 FORTRAN 编译程序将每个程序段编译成独立的机器指令表示的目的程序模块, 称为可浮动的二进制目的模块, 简称 BT 格式目的模块。然后, 再由连接编辑程序将这些编译好的 BT 格式模块, 加上根据程序的需要, 从标准子程序库中选择的标准子程序的 BT 格式模块连接编辑成一个完整的可执行的目的程序。这个目的程序称为可浮动的内存映象, 简称 IMT 格式的目的程序。这个 IMT 格式的目的程序经装入程序 (也称加载程序) 装入 (或叫加载) 到内存的某一分区之后, 就可以启动运行了。整个上述过程如图 1-4 所示。

构成一个程序段的 FORTRAN 语句分为可执行语句和不可执行语句两类。

可执行语句说明程序的某种操作。每个可执行语句在目的程序中都对应若干条机器指令。可执行语句有下列三种:

(1) 赋值语句, 它给变量或数组元素赋值。

(2) 控制语句, 它控制程序执行的顺序。如转移、调用子程序、返回、停止等语句都是控制语句。

(3) 输入输出语句。

不可执行语句只给编译程序提供一些信息, 诸如数据的类型, 存贮的分布, 输入输出的格式等等。这类语句的作用是为可执行语句的执行作好准备。不可执行语句有下列五种:

(1) 组织语句, 它说明变量的类型, 存贮的分布等。

(2) 数据初值语句。

(3) 输入输出格式语句。

(4) 子例程子程序定义语句。

(5) 函数子程序定义语句。

上述各类语句将在以后各章中详细介绍。

§3 FORTRAN 程序的书写格式

在前两节中, 通过具体的程序举例, 对一个完整的 FORTRAN 程序及其结构作了概括的介绍。从本节开始, 将具体介绍 FORTRAN 语言。这里, 首先介绍 FORTRAN 语句的书写规则。

3.1 FORTRAN 的字符集

众所周知, 任意一个英语句子中所使用的字符总不外乎是 26 个字母和一些规定的标点符号。我们称这 26 字母和规定的标点符号为英语的基本字符。同样, FORTRAN 虽然是一种简单的程序设计语言, 但作为语言它也规定了自己所使用的基本字符。FORTRAN

的基本字符有 49 个。具体是：

26 个大写英文字母 A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W, X, Y, Z

10 个数字 1, 2, 3, 4, 5, 6, 7, 8, 9, 0

13 个专用字符，它们是：

= 等号

+ 加号

- 减号

× 星号

空格

/ 斜杠

, 逗号

. 小数点

(左括号

) 右括号

\$ 美元符号

& 商业和符号

▼ 单引号

FORTRAN 编译程序允许在文字型常数中，或在 H 和 A 型区描述符控制下的格式数据中使用除上述字符集之外的计算机本身所特有的其它字符。但是，为了使一个 FORTRAN 程序能比较容易地从一台计算机移植到另一台计算机上运行，最好严格按 FORTRAN 字符集规定的字符使用。

3.2 FORTRAN 程序的书写规则

因为 FORTRAN 语言原来是适合于用卡片输入的语言，所以 FORTRAN 程序在书写格式上的一个显著特点是分行书写。

一张卡片有 80 列，自左至右顺序编上列号，称第 1 列，第 2 列，……，第 80 列，每一列穿一个字符。卡片及每个字符的孔型如图 1-5 所示。

FORTRAN 程序纸上的一行相应于一张卡片。所以，其每一行也相应分为 80 列，自左至右顺次称为第 1 列，第 2 列，……，第 80 列，每一列书写一个 FORTRAN 字符。FORTRAN 程序纸的格式如图 1-6 所示。

FORTRAN 编译程序只处理第 1 列～第 72 列中的字符，对第 73 列～第 80 列中的字符不予处理（即不产生任何目的程序）。以下所说的 FORTRAN 行具体就是指第 1 列～第 72 列的内容。

一般来说，一个 FORTRAN 语句写在程序纸的一行上就足够了。但是，如果一个语句很长而在一行内写不下时，那么允许接在下一行内继续写。这样，这个语句的第一行叫做起始行，以下接着写的行叫做继续行。FORTRAN 允许一个语句最多可以有十九个继续行。起始行和继续行通称为语句行（或称 FORTRAN 行）。书写一个 FORTRAN 语句时，总是从一个起始行写起，然后书写后面可能出现的继续行。语句行的书写规则



图1-5 卡片

如下:

(1) 起始行的书写规则

起始行分三个区填写

第1~5列 填写语句标号, 叫标号区。

第6列 填写0字符或是空格字符, 标志这一行是起始行。

第7~72列 填写具体的 FORTRAN 语句内容。

(2) 继续行的书写规则

继续行只能跟在起始行或前一个继续行之后, 它的书写规则是:

第1~5列 一般为五个空格字符。

第6列 填写一个除0字符或空格字符以外的任意其它字符, 标志这一行是继续行。

第7~72列 填写 FORTRAN 语句的后续内容。可见, 判别一个行是不是继续行的标志是看第6列中的内容。如果第6列中含有除0和空格字符以外的其它字符, 那么该行就是继续行。这就要提醒我们注意, 在写 FORTRAN 语句或检查穿孔卡片时, 一定要特别注意第6列中的内容, 否则, 由于错列将会造成意想不到的语法错误。

例如 下列各行都是一行写完一个 FORTRAN 语句的行,

```

READ (105, 999)  A, B, C
AREA=SQRT (S*(S-A)*(S-B)*(S-C))
105  A=SIN(X)+1.5/(PL**2-DEL)

```

而下列三行表示了带有两个继续行的一个 FORTRAN 语句。

```

999  FORMAT (1H , 2HA=, F6.3, 5X,
1      2HB=, F6.3, 5X, 2HC=, F7.3, 5X,
2      5HAREA=, F9.3)

```

书写语句行时, 应注意下列几点:

(1) 如果一个语句需要加标号时, 那么语句标号必须写在起始行的标号区中;

(2) 除逻辑条件语句外, 任何两个 FORTRAN 语句不能写在同一行中;

(3) 在一个 FORTRAN 语句中, 除文字型常数中的空格字符外, 一般的空格字符是没有意义的。如下列三种不同表示的语句行是等价的。

```
7 6 4  A=SIN (X) / (X+2. 0)
7 6 4      A      =   SIN      (X)      / (      X+   2. 0      )
7 6 4  A      =S I N ( X ) /                      (X+2. 0)
```

除语句行外, 从 § 1 的程序举例中可以看到, 一个程序段还必须有一个结束行, 并可能有若干个注释行。它们的书写规则分别叙述于下:

结束行, 它不是 FORTRAN 语句, 不产生任何目的程序。它只是给编译程序一个信息, 表示一个程序段的结束。应强调指出: 每个程序段必须有一个结束行。结束行的书写规则是:

第 1~6 列 空格字符。

第 7~72 列 依次书写 E、N、D 三个字符。E、N、D 三个字符的每一个字符的前后都可以有任意个空格字符。下面列举的几种结束行的写法都是允许的, 但通常采用第一种写法。

```
END
```

```
      E      N      D
E              N              D
```

注释行, 它对程序的动作不产生任何影响, 其作用仅仅是对程序作必要的注释说明。所以, 在程序段中, 注释行是可有可无的, 根据程序人员的需要而定。注释行可以插在程序段的任意位置上, 但不允许插在起始行与继续行或继续行与继续行之间。注释行之后允许跟任意个注释行。注释行的书写规则是:

第 1 列 填写一个字符 C, 表示这是注释行。

第 2~72 列 使用 FORTRAN 字符填写注释的内容。但在第 2 列中不允许使用字符 “/” (斜杠), 这是 FELIX C-256 机 FORTRAN 的一点特殊规定。

下面列举的各行都是注释行。

```
C      THE  MAIN  PROGRAM
C
C      THIS IS A COMMENT LINE
```

§ 4 语句标号, 标识符和标准函数

语句标号和标识符是 FORTRAN 中最常用的两个基本概念, 虽然简单, 但较重要。

4.1 语句标号

在一个程序段中, 为了转向或引用某个语句, 可以给它一个编号以标识这一语句。这个编号称语句标号。如程序举例中语句

```
999  FORMAT (3F6. 3)
```

中的号码 999 就是语句标号。

语句标号是 1~5 位的无符号十进制整数, 即 1~99999 范围内的任一个号码。标号左边的零无意义。显然, 只有当要引用这一语句时, 才有必要加语句标号。没有标号的语句, 在标号区中是五个空格, 称无标号语句。因为语句标号仅仅是一个起标识作用的号码, 所