

FORTRAN 语言及其应用

FORTRAN

语言及其应用

杨则正 刘晓青 李 集 编译

机
械

12

2/1

社

机械工业出版社

FORTRAN语言及其应用

杨则正 刘晓青 李 集 编译

李月景 邹定国 审校



机械工业出版社

本书通过 29 个典型实例介绍 FORTRAN 语言的内容、结构和应用。
实例讨论工程技术中的数值计算, 内容涉及数值积分、方程组求解、
矩阵、傅里叶级数和拉普拉斯变换, 其中包括放射性物质蜕变计算, 管内
温度场分布求解, 示波器图象分析, 质谱仪中离子径迹图绘制等有实际参
考价值的实例。实例 12 专门讨论数值计算中必然出现的误差问题及其解
决方法。此外, 本书安排了在国内外许多 FORTRAN 语言的书中很少提
及、甚至认为 FORTRAN 只适合于科技计算因而干脆不予讨论的非数值
型问题应用实例, 包括文献处理, 报表设计, 统计分析等。最后讨论程序
的优化问题。本书可供从事工程技术、管理和教学工作者以及大专院校学
生参考。

FORTRAN 语言及其应用

杨则正 刘晓青 李 集 编译

李月景 邹定国 审校

责任编辑: 邱锦来
封面设计: 王 伦

JS398/01

机械工业出版社出版(北京阜成门外百万庄南里一号)

(北京市书刊出版业营业许可证出字第 117 号)

机械工业出版社印刷厂印刷

新华书店北京发行所发行·新华书店经售

开本 $787 \times 1092 \frac{1}{4}$ · 印张 $15 \frac{1}{4}$ · 字数 370 千字

1987 年 9 月北京第一版 · 1987 年 9 月北京第一次印刷

印数 0,001—5,500 · 定价: 3.50 元

统一书号: 15033 · 6963

前 言

不了解计算机语言，自然也就无从着手搞程序设计。但是，程序设计的核心不是语言，而是就具体研究对象构造相应的算法（或者说形成算法思想），以及组织适当的数据结构。因此，单纯讲述程序语言的各种严格的繁琐规定，不仅使读者感到枯燥乏味，而且，也难以收到应有的效果。另一方面，结合实例讲述程序语言要考虑到：例子过于简单，对读者帮助不大。反之，过分强调所谓专业，例子太专、太深、太窄或太偏，势必使一本介绍程序语言的书失去其应用的普遍性。

Daniel D. MacCracken 所著《FORTRAN 及其工程应用》（《FORTRAN with Engineering Application》）一书着眼于工程技术问题分析，突出算法思想描述，通过 29 个典型实例讲述了 FORTRAN 语言的全部内容。我们选择其中有价值的内容材料，补充了其他实例，特别是非数值型问题方面应用的实例，而保持由浅入深的章节结构。

FORTRAN 是目前国外最为流行的高级语言之一。在国内，它的应用也日趋广泛。绝大部分介绍 FORTRAN 的书，都强调 FORTRAN 是“适合于科技计算的语言”。实际上，就象拉货的卡车可以载人，载人的客车也可以拉货一样，FORTRAN 也完全可以用来处理非数值型问题。本书将以适当的篇幅讨论 FORTRAN 在非数值型问题方面的应用。目前，数字计算机在管理方面的应用的比例不断扩大，管理方面的大部分问题，例如科技文献的处理，属非数值型应用范畴，而不是数值计算。

本书可作为学习 FORTRAN 的辅助教材，实例和习题有助于您深入了解 FORTRAN 的内容和掌握 FORTRAN 程序设计技术。本书也是自学 FORTRAN 的参考书。我们希望它对想学会应用 FORTRAN 解决问题的广大读者有所裨益。

编写过程中，参加翻译的有刘晓青同志（实例 1、2、7、9、10、19、20）和李集同志（实例 12、22、23、24、29）。笔者与吴亢同志进行了许多有益的讨论，李月景和邹定国同志不仅对技术内容认真审校，而且，在文字表述方面，也作了仔细推敲。在此，对上述同志以及为促成本书与读者见面而付出辛勤劳动的所有同志表示衷心感谢。由于水平所限，书中错误在所难免，欢迎读者批评指正。

杨则正 1985.12

几点说明

1. 除少数例子, 因机器容量限制和/或没有具备相应软件而未能上机调试外, 本书实例程序均在 HP3000 计算机上通过。

2. 本书材料是这样安排的: 讲述一个语句, 相应举一程序予以说明。一般, 实例程序中用到的语句都在前面出现过; 但为使程序保持其完整性, 保留了某些程序中已经用到、但本书中要在后面才讲到的那些语句, 凡遇到这种情况, 在实例程序之前, 先举一个由已学过的语句组成的简单程序例子, 以便使初学者不致发生困难。

3. FORTRAN 程序是由各种不同的语句组成的。因此, 同一个程序, 往往既可以用来说明这个语句, 也可以说明那个语句。本书选择程序实例力求有针对性。例如, 实例 6 (文献分类程序) 是配合讲述逻辑 IF 语句的, 这个程序中连续使用了 23 个逻辑 IF 语句, 如此等等。

4. 本书中所说的“程序片断”是指为集中说明某一个语句, 而从程序中“截”下来的部分。“程序片断”与“程序段”是不同的。程序段在标准 FORTRAN 中是有明确的定义的 (一个主程序是一个程序段, 一个子程序也是一个程序段)。

5. 为节省篇幅, 个别程序没有展示其全部语句, 但不影响读者对程序的完整结构的了解。读者只需凭借展示的语句完全可以依葫芦画瓢将删节的语句添上去。例如, 实例程序 6 和 8 就是这样的。

6. 基于同样的理由, 并考虑到排版工作量, 原程序中一般都不加文字注释行, 只有最后几个程序中使用了注释行, 并相应附上中文译文。

7. 本书程序实例中的变量名不限于 6 个字符, HP 机允许使用 15 个字符组成的变量名。

8. 框图中循环体表示, 采用的是文献〔4〕的画法, 穿孔卡的简化表示法是按文献〔4〕和〔5〕的办法画的。

9. 本书实例程序是按标准 FORTRAN 66 陈述的, 而在讲述语句时, 对标准 FORTRAN 77 的一些新规定作了介绍。

10. (西德) 柏林工业大学 (TUB) 将文献〔1〕全文译成德文, 编、译过程中, 我们参照了德文译本 (文献〔2〕), 将英制一律换算为法定单位制; 此外, 译本对原书中若干错排、误排以及不确切之处直接作了更正, 不再一一单独说明, 只有译本未发现的个别出错处, 本书中以“注”的形式作了说明。

11. 本书大部分练习题及其答案均取材于文献〔1〕, 为便于读者利用, 书末附有答案 (限于篇幅, 语句较多的个别例子未列出)。

目 录

实例 1	导线的电阻 程序基本知识	1
实例 2	导线截面积和电阻计算 变量的值 整型常数和实型常数	5
实例 3	绳子张力计算 算术运算符的优先级	7
实例 4	放射性物质的衰变 FORTRAN 函数	11
实例 5	主题词自动编号 GO TO 语句 程序的运行	14
实例 6	文献分类程序 逻辑 IF 语句 文件概念	20
实例 7	RLC 振荡电路计算 框图 算术 IF 语句	27
实例 8	杨辉三角形在文献工作中的应用 计算 GO TO 语句 算法思想	32
实例 9	电阻表 输入和输出格式	44
实例 10	牛顿近似算法 程序错误检查	52
实例 11	RLC 串联电路 复数基本运算	62
实例 12	误差 双精度数	68
实例 13	回路电流判断 逻辑运算	81
实例 14	生成前 1000 个质数 数组 下标变量	86
实例 15	立塔绷索计算 DO 语句	98
实例 16	线性方程组的高斯解法 带下标的 DO 语句	105
实例 17	文献磁带读取 格式数组	114
实例 18	大小写字母转换 DATA 语句	118
实例 19	大型联立方程组解法 磁带操作	121
实例 20	利用数值积分求贝塞尔函数 语句函数	125
实例 21	数值表通用编排格式设计 子程序 COMMON 语句和 EQUIVALENCE 语句 可变维数	132
实例 22	任意函数的数值积分 EXTERNAL 语句	147
实例 23	函数的零值 ENTRY 语句 非标准返主语句	151
实例 24	工件日产量统计计算	157
实例 25	矩阵子程序包 矩阵求逆 特征值和特征向量	162
实例 26	管内温度分布拉普拉斯方程解法	172
实例 27	阴极射线管示波器图象的傅里叶分析	188
实例 28	单极质谱仪中离子的径迹 微分方程数值解脱机绘制成图	195
实例 29	求解两个多项式商的拉普拉斯逆变换	202
	关于程序优化问题的讨论	225
	习题解答	229
	参考文献	237

实例1 导线的电阻 程序基本知识

我们来考察一个程序例子，作为学习程序设计的开始。假定，有一根铜导线，已知其长度和横截面积，我们要计算它的电阻，所用的公式为：

$$R = \rho \frac{L}{A}$$

式中 R ——电阻；

L ——长度；

A ——面积。本程序中单位用 mm^2 ；

ρ ——电阻率。标准铜退火在 20°C 时的电阻率， $\rho = 0.01754\Omega \cdot \text{mm}^2/\text{m}$ 。

我们要写一程序，这个程序首先读取记录有铜导线长度和横截面积数值的数据卡，接着计算导线的电阻，最后打印长度、横截面积和电阻值。

程序1是完成这项工作的 FORTRAN 程序，它写在 FORTRAN 程序纸上。我们仔细研究一下这个程序，先看看 FORTRAN 语言的成分是什么和怎样构造一个程序。有这么一点背景知识，读者就可以自己做程序设计的练习了。

程序中头三行第一列都有字母“C”，它们称为注释行。注释行用来给人提供阅读程序的信息，对计算机毫无作用。要大力鼓励读者按自己的要求写注释，这样，可以对复杂程序的各个部分予以标识并使之目的明确。第一个注释行表示“程序1”；第二个注释行表示“计算已知长度和截面积的导线的电阻”；而第三个注释行仅仅用来保证行间间隙，使程序显得清晰、易读。

注意，这里我们一下子就遇到两个问题：一是使计算机程序便于阅读；二是要确切地告诉计算机我们想让它干什么。前者，即对人来说的可读性问题，通过在程序中使用适当的注释行，选用有意义的变量名，以及妥善安排书写格式、留出一定空行等项办法来解决[⊖]；至于叫计算机准确地执行任务的问题，则通过遵循有关语句书写的各种规定来保证。

以 REAL 开头的那行是程序的第一条语句。我们看到，第7列是语句的起始位置（有两个语句前面分别标有1和2，这是语句标号）。在这个程序里，REAL 语句的作用是“通知”计算机，以 L 、 A 和 R 命名的三个变量均为实型数。这意味着它们可以在 10^{-40} 到 10^{40} 的大致范围内取一正值、负值或零。本程序中所用的变量名与公式中出现的名称是一样的，但以后我们将会看到，并非总是如此。我们以后还会学到，除实型变量外，还有整型、双精度型、复型和逻辑型四种类型的变量。

下一条语句，即 READ 语句，要求计算机读入一张数据卡。READ 语句的作用是：把语句中的变量 L 和 A 的值从一张穿孔卡读入计算机。括号里的“5”是读卡机的代号，“1”是 READ 语句所属的 FORMAT 语句的标号。

FORMAT 语句描述数据值在卡片上的格式或安排：第一个数据值穿在卡片1~10列的任何位置，必须穿小数点；如果是负数，在1~10列还要穿一个负号。卡片上的第二个数据

[⊖] 所谓空行指的是像程序1中第3个注释行那样的空行。纯粹的空白行是不允许使用的

来仔细讨论这条重要语句。

赋值语句的左边必须永远仅有一个变量名，作为这个语句执行结果的值被赋予这个变量。赋值号“=”告诉计算机，这条语句是赋值语句。现在我们看到一个与计算公式十分相当的表达式。在 FORTRAN 语言中，用星号“*”表示乘，用斜线“/”表示除。在以后的例子中，我们还会看到，加、减是用普通的加、减号“+”、“-”来表示的，而指数运算则用两个星号“**”来表示。

我们再强调一遍算术赋值语句的意义：利用表达式中已获值的变量的值（不管是什么值），对右边的表达式进行计算，新计算出来的表达式的值再赋给左边的变量。

WRITE 语句的作用是取出数据和打印结果。6 是输出设备代号，2 是 WRITE 语句所属的 FORMAT 语句的标号，这跟 READ 语句情况很相似。待打印的三个变量的数值，按照我们所要求的在打印纸上的排列顺序被打印出来。

这个 WRITE 语句的 FORMAT 语句，跟上一个 FORMAT 语句有些不同，它规定打印机正常走纸（一行），一行最多允许打印 10 个数，每个数字在小数点后可有五位数，而且，要求按照“指数形式”，即一个数乘以 10 的幂的形式打印出来。

STOP 语句“通知”计算机停止程序的执行。

END 语句后面不再有任何语句，它是程序的结束语句。在这个例子中，乍看起来，STOP 和 END 似乎相互重复，但实际上并非如此。END 语句必须是最后一个语句，STOP 语句则不然，除放在 END 语句之前外，它还可以多次出现在程序的任何位置。有时候并不需要 STOP 语句。在以后的例子中，我们将看到它的使用的各种不同情况。

当这个程序连同图 1-1 的数据卡一起提交给计算机去执行时，便打印出输出结果。我们看到，打印出来的数字的表达形式，跟它们在数据卡上的输入形式有些不一样。这里，打印的输出数字是用指数形式来表示的。例如，我们要把第一个数 $1.00000 \text{ E} + 03$ 读作 1.00000×10^3 ，第二个数 $4.90500 \text{ E} - 2$ 等于 4.90500×10^{-2} 。

程序 1 的输出打印结果：

1.00000 E + 03 4.90500 E - 02 3.57594 E + 02

这种打印数字的形式允许我们写程序时不必事先知道结果可能有多大或多小。以后还会学到，如果我们愿意，也可以将结果按穿孔卡片上的形式输出。

我们把上面讲的内容归纳如下：

1. 注释行：用于对程序作标识和解释。
2. REAL：规定变量为实型变量。
3. READ：例如，READ(5,1) L, A——将相应数据卡上的数据，按照语句标号为 1 的 FORMAT 语句所规定的格式，读入计算机，并分别在 L 和 A 的名下存储起来。
4. FORMAT 语句永远与 READ 语句和 WRITE 语句连用，规定数据读入和输出的格式。
5. 算术赋值语句：为计算出结果，需要用一条或若干条这样的语句。算术赋值语句由左边的一个变量名、赋值号和右边表达式组成。凡出现在赋值号右边表达式中的变量，必须是已经被赋值的变量（通过 READ 语句被赋值，或通过执行前面的另一赋值语句被赋值，或者通过我们以后要学到的 DATA 语句被赋值）。
6. WRITE：例如，WRITE(6,2) L, A, R——将变量 L, A 和 R 的值，按标号为 2 的

FORMAT 语句所规定的格式, 依次打印出来。

7. STOP: 可执行语句, 中止程序执行; 程序中可以有、也可以有一个或多个 STOP 语句, 它可以出现在任何位置。

8. END: 程序结束语句, 必须有一个而且只能有一个 END 语句。

用户只可能在一台具体的而不是抽象的计算机上算题。因此, 可能遇到文中所未讲到的另外情况。为此, 我们作以下两点说明:

(一) 程序书写格式

以上提到的程序书写格式, 是一种标准格式。如果按标准格式编写程序, 凡配有 FORTRAN 语言的计算机都能“承认”或接受。但是, 某些计算机系统, 例如德律风根公司曾生产的 TR440 系统和惠普公司生产的 HP/3000 系统, 除标准格式外, 还提供各自的所谓“自由格式”。当然, 自由格式绝对不是意味着用户可以随意“自由”选取 FORTRAN 程序的格式, 而是向用户提供一种更为方便的简单格式, 从而有助于提高程序设计的效率。例如, 按 TR440 系统自由格式规定, 一条语句以“!”开始, 如果有语句标号, 则将“!”插在标号和语句关键字中间。这样, 敲一个“!”键总比先敲六个空白键然后再“正式”敲入语句来得方便。

(二) 关于输入和输出设备

用户上机算题, 要频繁地与输入、输出设备直接打交道。穿孔卡片不是唯一的数据载体, 输入设备也不限于读卡机, 除打印机外, 还有其他各种输出设备。目前, 应用最为广泛的输入设备是 CRT 显示器。如果采用读卡机输入程序, 那末, 用户送入计算机的程序就是一叠依次排列的穿好孔的卡片; 如果通过显示器输入程序, 则是一行行的屏幕上的语句。显示器操作方便、容易修改。一般情况下, 用户在调试程序时, 可直接在屏幕上看到结果。显示器同时又是输出设备, 通常只有当结果准确无误时, 方通过打印机打印出来。

练 习

下面每个练习题, 要求写出完整的程序, 并说明数据卡看上去像什么样子。除几张控制卡外, 程序要完整。

1. 一张穿孔卡记录有三角形底边 B 和这个底边上的高 H 的值, 利用下面的公式计算三角形的面积,

$$A = BH / 2$$

按照 B 、 H 和 A 的顺序打印出它们的数值。记住, 在程序中写常数 2 时要有小数点。数据卡上取 $B = 10.0$, $H = 16.5$ 。

2. 一张穿孔卡依次含有电阻和电压值, 用欧姆定律计算电流,

$$i = \frac{V}{R}$$

i = 电流;

V = 电压;

R = 电阻。

按电压、电阻和电流的顺序, 打印出它们的数值。数据卡上取 $R = 4700 \Omega$, $V = 8.37 \text{ V}$ 。

3. 利用下面的算式, 计算介电质平行板电容器的电容,

$$C = \epsilon_0 \frac{A}{S}$$

式中 C —— 电容;

A —— 面积;

S —— 距离;

ϵ_0 —— 真空介电常数, $\epsilon_0 = 8.855 \times 10^{-6} \mu\text{F}/\text{m}$

设 $A = 0.43 \text{m}^2$, $S = 0.0005 \text{m}$.

实例2 导线截面积和电阻计算 变量的值 整型常数和实型常数

本实例程序在第一个程序基础上稍有扩充, 这样, 在着重学习一两点新知识的同时, 还可以巩固我们在程序结构方面的现有知识。

这次, 依然从 20°C 温度下退火铜标准导线开始讨论, 但已知圆形截面导线的直径, 没有直接给出横截面积。要求程序不仅算出导线电阻, 还要算出它的横截面积。我们通过以下两个公式对这个问题作数学描述:

$$A = \frac{\pi}{4} d^2$$

$$R = \rho \frac{L}{A}$$

式中 A —— 面积;

d —— 直径;

L —— 长度;

R —— 电阻;

ρ —— 电阻率。本程序中 $\rho = 0.01754 \Omega \cdot \text{mm}^2/\text{m}$ 。

相应的程序见程序 2。

我们看到, 这个程序与上一个程序有相似之处。REAL 语句含有附加变量 D。顺便提一下, REAL 语句中变量名的顺序无意义。READ 语句和 FORMAT 语句是我们已熟悉的。

计算横截面积的赋值语句有两点新内容: 一是用两个星号表示指数运算符, 它“命令”计算机对 d 的值进行二次幂的自乘运算; 二是在实例中第一次引进了整型常数——只允许取整数值、书写时不带小数点的常数。

FORTRAN 程序中, 语句是按书写顺序执行的, 除非我们用 GO TO 语句和 IF 语句 (这两个语句在后面介绍) 改变程序执行顺序。因此, 一旦由一赋值语句给一变量赋了值, 我们马上就可以在后续语句里直接引用该变量。在计算电阻的语句中, 我们就用到了这一点: 此语句中的 A 通过上一个赋值语句获值。

这里, 我们还看到, 程序里的实型常数也可以写成指数形式, 正像我们看到的输出结果的书写格式一样。如果指数是正数而不是负数, 则加号可省略。其他一些以指数形式出现的实型常数及相应的非指数形式的写法举例如下:

12.08E5

1208000.

1.E7	10000000.0
1.E-5	0.00001
.75311E-4	0.000075311

```

C      PROGRAM 2
C
      REAL L, A, D, R
      READ(5,1) L, D
1     FORMAT(8F10.0)
      A = 0.78539816 * D * * 2
      R = 1.754E-2 * L/A
      WRITE(6,2) L, D, A, R
2     FORMAT(1H,1P10E13.5)
      STOP
      END

```

程序 2 计算已知长度和直径的圆形截面导线的电阻

现在让我们总结一下实型常数和整型常数的特点。一个整型常数必须是整数：零或正或负，一个整数由前置符号（“+”或“-”，“+”号可省略）和一连串 0~9 十进制数码组成，例如：0，+101，108，-10000等都是整型数，整型数永远不带小数点。另一方面，实型常数可以是带整指数或带小数的数，例如，2.233E+03或3.3等。实型数可以是零或是 10^{-40} 到 10^{+40} 的大致范围内的任何正数或负数。写实型常数必须总带小数点。写很大或很小的数时，可以采用指数形式，但在E之前的部分仍要写上小数点。E之后的指数部分必须不带小数点。

程序其他部分没有什么新内容。数据卡如图 2-1 所示。

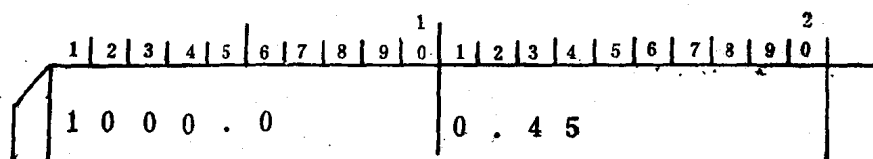


图2-1 程序 2 的数据卡形式 (示意图)

程序 2 的输出结果:

1.00000E 03 4.50000E-01 1.59043E-01 1.10285E+02

练 习

1. 将下列数写成 FORTRAN 实型数,

256 2.56 -43,000 10^{12} 0.000000492 -10 -10^{-10}

2. 下列所有的数都不允许用作实型常数,为什么?

87,654.3 +987 $9.2E+87$ $7E-9$

3. 下列各例中,是不是每对实型常数都表示同一个数?

a. 16.9 +16.9

- b. 23000. 2.3E4
 c. 0.000007 .7E-5
 d. 1.0 1.
 e. .906E5 +906.0E+2

4. 一张数据卡依次含有力、质量和时间的值。写一程序计算加速度和位移；然后将时间、力、质量、加速度、位移依次打印出来。计算公式如下（假定，初始条件是：在零时刻质点处于静止状态）：

$$a = \frac{F}{m}$$

$$x = \frac{1}{2}at^2$$

式中 F ——力；
 m ——质量；
 a ——加速度；
 t ——时间；
 x ——从时间 $t = 0$ 时刻算起质点所移动的距离。

5. 一张穿孔卡上含有单位为 Hz 的频率值和电感值。按以下公式，设计一程序计算角频率和阻抗，

$$\omega = 2\pi f$$

$$X = 10^{-3}\omega L$$

式中 ω ——角频率；
 f ——频率；
 X ——阻抗；
 L ——电感。

实例3 绳子张力计算 算术运算符的优先级

图3-1表示：两个不同质量的物体，通过一根绳子（不计绳重）两端系住，绳子跨过滑轮（不计绳子与滑轮之间的摩擦力，不计滑轮质量）。绳子张力计算公式是：

$$T = \frac{2m_1m_2}{m_1 + m_2}g$$

式中 m_1, m_2 ——质量；
 g ——重力加速度；
 T ——绳子张力。

数据卡只记录两个物体的质量， g 作为常数处理，并直接写入程序。要求计算绳子的张力，将张力和两个物体的质量一起打印出来。

程序3是在我们已熟悉的知识的基础上写的，只有两点是新内容：一是变量的命名，二是算术赋值语句右边表达式中的括号。

为了区分两个物体的质量，我们分别用M1和M2表示。这里

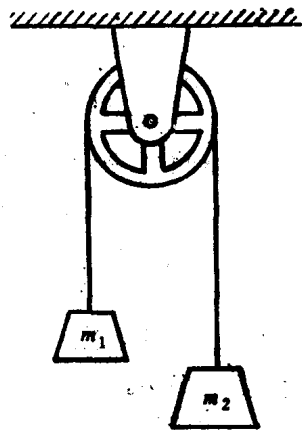


图3-1 张力计算的物系示意图

涉及到变量命名问题。FORTRAN 规定，变量名由 1~6 个字符组成，这里的字符仅限于 26 个大写英文字母和 10 个十进制数码（0~9），此外，首字符必须是字母。这样，可组合成各种各样的变量名。例如，X45，Z，VECTOR，FOURTH，ROOT3，KLMNOP，A12345，ALPHA，ZCUBED，TORQUE，FARAD 等都是合法的变量名。注意，根据上述命名规则，象以下这样一些名称是非法的，不能作为变量名：

ZSQUARED (字符数目超过 6)
 12AT7 (首字符不是字母)
 A + B (使用了字母和 0~9 十个数码之外的专用字符)
 X3.4 (使用了字母和数码外的字符)

```

C      PROGRAM 3
C
      REAL M1, M2, T
      READ(5,1)M1, M2
1     FORMAT(8F10.0)
      T = 19.613 * M1 * M2 / (M1 + M2)
      WRITE(6,2)M1, M2, T
2     FORMAT(1H, 1P10E13.5)
      STOP
      END
  
```

程序 3 绳子张力计算

在遵守上述命名规则前提下，用户可以自由选取变量名。但是，最好选用便于记忆的、能表征变量表示的对象的意义的变量名，例如，用 MASS 表示质量。当然，这全都由程序设计者本人选定。本程序中用 M1 和 M2 表示两个不同物体的质量，这只是许许多多可能中的一种可能，也可以用 MONE 和 MTWO，或用 MFIRST 和 MSECND 等等来表示。

在本书中，我们一般是在类型说明语句（即 REAL, INTEGER, COMPLEX, LOGICAL 或 DOUBLE PRECISION）中给变量命名的。在这种情况下，变量首字母不表示什么意义。例如，程序 3 的第一个语句，即类型说明语句

```
REAL M1, M2, T
```

既给定了名称分别为 M1, M2 和 T 的三个变量，也规定了它们的类型——都是实型，这里，首字母 M 和 T 与变量类型无关。但是，应该注意，这种做法并非是强制性的。如果一个变量不曾在任何类型语句中说明过，系统将按照下述规则自动规定变量为实型或整型：如果变量首字母是 I、J、K、L、M 或 N 六个字母中任何一个字母，则为整型变量，否则为实型变量。我们倾向于不要一开始就依赖这条规则。因为，本意取实型数的变量名，由于疏忽而取了整型数的变量名，这就会出问题。例如，很容易“习惯地”用 I 表示电流，用 M 表示质量。这样就会出错。

算术表达式中的括号，用来表示程序员的意图，与普通代数的算式很相似。在本例的算术赋值语句的右边表达式中，如果没有括号，前三项之积就会被 M1 除，得商加 M2。因为，除法运算先于加法运算。在这点上，FORTRAN 算符与大家熟悉的算符一致。

换言之，通过算术表达式中的括号来规定我们希望的、不同于无括号时的运算优先级的运算顺序。所谓无括号时的运算优先级，指在没有括号的情况下，先进行指数运算，再进行乘、除运算，最后进行加、减运算。

例如， $(A + B) * C$ 意味着先加后乘；没有括号，即 $A + B * C$ 则表示先乘后加。 $A / (B + C)$ 要求先做加法；没有括号，则 A 被 B 除，然后，所得商再与 C 相加。

考察表达式 $(A + B) / (C - D)$ ，注意有无括号的以下四种不同的写法及其分别表示的意义：

$$(A + B) / (C - D)$$

$$\frac{A + B}{C - D}$$

$$A + B / (C - D)$$

$$A + \frac{B}{C - D}$$

$$(A + B) / C - D$$

$$\frac{A + B}{C} - D$$

$$A + B / C - D$$

$$A + \frac{B}{C} - D$$

用数学上的写法描述的某些表达式，一般被认为有二义性，但在 FORTRAN 中是允许使用的，当然，使用时要小心。例如， $A / B * C$ 究竟表示 $\frac{A}{BC}$ 还是 $\frac{A}{B}C$ 呢？正确的答案是后者。因为在没有括号的情况下，从左到右对表达式进行计算。如果遇到这一类麻烦的问题，可遵循这样一条原则：有疑问就加括号！

在讨论了算术赋值语句中表达式的几个例子之后，可以给它正式下一个定义，以便能普遍使用这个概念。FORTRAN 语言中，算术表达式是对于数进行计算的规则。在很多情况下，一个表达式就由一个常数或一个变量组成。例如，在 $X = 12.0$ 这样一个语句中，“右边表达式”就是指常数 12.0，以后还会经常见到这一类例子。到目前为止，我们所遇到的表达式，都可以由一个、两个或多个常数和/或变量通过 $+$ 、 $-$ 、 $*$ 、 $/$ 、 $**$ 组合而成。

表 3-1

数学式子	允许的表达式	错误的表达式
ab	$A * B$	AB (没有运算符)
$a(-b)$	$A * (-B)$ 或 $-A * B$	$A * -B$ (两个算符连写)
$a + 2$	$A + 2.0$	$A + 2$ (整型实型混用)
$-(a + b)$	$-(A + B)$ 或 $-A - B$	$-A + B$ 或 $-+A + B$
a^{i+2}	$A ** (I + 2)$	$A ** I + 2$ ($= a^i + 2$ ，且整型实型混用)
$a^{b+2}c$	$A ** (B + 2.0) * C$	$A ** B + 2.0 * C$ ($= a^b + 2 * c$)
$\frac{ab}{cd}$	$A * B / (C * D)$ 或 $A / C * B / D$	$A * B / C * D$ ($= \frac{ab}{c}d$)
$\left(\frac{a+b}{c}\right)^{2.5}$	$((A + B) / C) ** 2.5$	$(A + B) / C ** 2.5$ ($= \frac{a+b}{c^{2.5}}$)
$a[x + b(x + c)]$	$A * (X + B * (X + C))$	$A(X + B(X + C))$ (丢失算符)
$\frac{a}{1 + \frac{b}{(2.7 + c)}}$	$A / (1.0 + B / (2.7 + C))$	$A / (1.0 + B / 2.7 + C)$

FORTRAN 语言要求, 一个表达式中, 不允许实型数和整型数混用[⊖], 只有一点是例外: 一个实型数允许进行整数次幂的自乘运算。因此, 象 $X**2$ 这样的表达式是允许的, 而 $X+2$ 之类的表达式则是不合法的。确立这条原则的原因在于: 在大多数计算机上, 实型量和整型量在机器内部的存储和运算方式是完全不同的。

程序 3 的输出结果:

2.00000E+00 1.00000E+00 1.30753E+01

表 3.1 分别列出了一些正确的表达式和一些错误的表达式, 以加深我们对表达式的理解。

练 习

1. 下列变量名中哪些是正确的, 哪些不符合变量命名规则? 在没有类型说明语句 (REAL 语句等) 的情况下, 哪些是实型变量, 哪些是整型变量?

G, GAMMA, GAMM423, I, IJK, IJK*,
J79-14, LARGE, R(2)16, BT07TH, 2N173,
6CA7, CDC160, DELTA, KAPPA, EPSILON,
A1.4, A1P4, FORTRAN, ALGOL

2. 下面给出几个数学式子和相应的 FORTRAN 表达式, 这些 FORTRAN 表达式都有错。请指错并写出正确的表达式:

- | | |
|--------------------------------------|------------------------------|
| a. $\frac{x+2}{y+4}$ | $X+2.0/Y+4.0$ |
| b. $\left(\frac{x+a+x}{2z}\right)^2$ | $(X+A+3.1415927)/(2.0*Z)**2$ |
| c. $\left(\frac{x}{y}\right)^{r-1}$ | $(X/Y)**(R-1)$ |
| d. $a+bx+cx^2+dx^3$ | $A+X*(B+X(C+DX))$ |

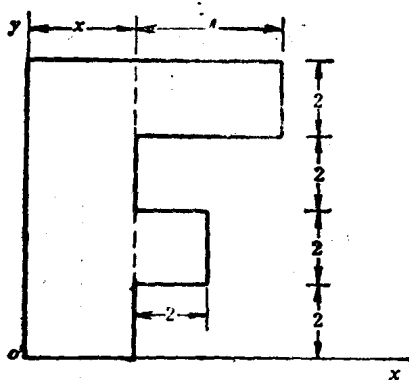


图 3-2 计算质心的示意图 (练习 5)

3. 写出下列数学式子对应的 FORTRAN 表达式:

- | | | |
|--------------------|---|--|
| a. $x+y^3$ | c. $a+\frac{b}{c+d}$ | e. $1+x+\frac{x^2}{2!}+\frac{x^3}{3!}$ |
| b. $a+\frac{b}{c}$ | d. $\left(\frac{a+b}{c+d}\right)^2+x^2$ | f. $\left(\frac{x}{y}\right)^{g-1}$ |

[⊖] 有少数 FORTRAN 系统允许使用“混合型”表达式。

4. 图 3-4 是大写英文字母 F，我们假想它是从一块金属板上剪下来的。这里，尺寸 x 是唯一的变量。这个图形的质心的坐标通过下式确定：

$$x = \frac{\sum_{i=1}^3 m_i x_i}{\sum_{i=1}^3 m_i}$$

式中， m_i 是组成 F 形金属板的三个矩形块中第 i 块的质量， x_i 是这个矩形块的质心与 y 轴的距离。约去上式中分子、分母的质量单位，则可用面积代替质量。例如，垂直置放的矩形块的质量（面积）等于 $8x$ ，它的质心与 y 轴的距离为 $x/2$ 。以尽可能简化的形式写出 x 坐标的公式，再写一程序，这个程序读取 x ，然后打印 x 和 $xbar$ ($xbar$ 是 F 型金属块的质心的 x 坐标)。

实例 4 放射性物质的衰变 FORTRAN 函数

服从单一型放射性衰变规律的物质，其放射（粒子的）速度由下式求出，

$$R = R_0 e^{-\lambda t}$$

式中 R ——放射性物质在 t 时刻的质量；

R_0 ——放射性物质在 0 时刻的质量；

t ——时间；

T ——半衰期；

λ ——衰变常数， $\lambda = 0.693/T$ 。

要写一程序，读入 R_0 、 t 和 T 的值，计算 λ 和 R ，最后将这五个值打印出来。

要做到这一点，我们必须学会指数运算，这正是 FORTRAN 数学函数提供的功能。在写程序之前，先扼要介绍这方面的知识。

FORTRAN 语言中有几种常用数学函数，如平方根，绝对值，正弦，余弦，反正切，对数和本实例中需要的指数等。可供用户调用的函数确切有多少，往往依所用 FORTRAN 文本和具体的计算机而异。但是，所有的 FORTRAN 系统都具备上面提到的函数。

各种函数都有自己的名称。表 4-1 列出了下面要用到的几种函数。

表 4-1 几种标准 FORTRAN 函数

数 学 函 数	FORTRAN 名称	数 学 函 数	FORTRAN 名称
指数	EXP	双曲正切	TANH
自然对数	ALOG	平方根	SQRT
常用对数 (10 为底的对数)	ALOG10	反正切 (用 rad 表示)	ATAN
正弦 (角度用 rad 表示)	SIN	绝对值	ABS
余弦 (角度用 rad 表示)	COS		

调用 FORTRAN 函数很简单：写出函数名称，后面跟着一对圆括号，括号内写表达式。FORTRAN 编译程序首先求出表达式的值，然后，再计算出相应的函数值。例如，在本实例中，要想求出我们所需要的指数，所做的一切只是在一算术赋值语句的适当位置写入